

**DISTRIBUTION-FREE STATISTICAL PROCESS CONTROL AND
BAYESIAN FEASIBILITY DETERMINATION**

A Dissertation
Presented to
The Academic Faculty

By

Di Liu

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology

December 2022

Copyright © Di Liu 2022

**DISTRIBUTION-FREE STATISTICAL PROCESS CONTROL AND
BAYESIAN FEASIBILITY DETERMINATION**

Thesis committee:

Dr. Seong-Hee Kim, Advisor
School of Industrial and Systems
Engineering
Georgia Institute of Technology

Dr. Nagi Gebraeel
School of Industrial and Systems
Engineering
Georgia Institute of Technology

Dr. Yao Xie, Advisor
School of Industrial and Systems
Engineering
Georgia Institute of Technology

Dr. Thomas Kurfess
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Kamran Paynabar
School of Industrial and Systems
Engineering
Georgia Institute of Technology

Date approved: August 16, 2022

That person who helps others simply because it should or must be done, and because it is the right thing to do, is indeed without a doubt, a real superhero.

Stan Lee

To my wife, and our son.

ACKNOWLEDGMENTS

The doctoral study in the past five years is a precious journey in my life. Spending half of the program during the pandemic makes it even more unforgettable, and I feel fortunate and grateful to have been through it. At the end of this journey, I want to express my thanks to all the people who have supported me along the way.

First and foremost, I would like to extend my sincerest gratitude to my advisors, Dr. Seong-Hee Kim and Dr. Yao Xie, for their support and trust during these years. Their guidance helped me in all the time of research, especially during the pandemic. Thanks to their extraordinary patience, we were able to continue inspiring communications and making progress on research even during the difficult time when we worked remotely. It has been a wonderful journey working with them, from which I learned how to be a researcher.

I would like to thank my thesis committee members: Dr. Kamran Paynabar, Dr. Nagi Gebraeel and Dr. Thomas Kurfess for their kindness in evaluating my thesis and their valuable feedback. The advice they provided on my research definitely broadens my horizon and perfects this dissertation.

I would also like to acknowledge LG Electronics for their financial support to enable this fantastic adventure.

Finally, I owe my gratitude to my families. I am grateful to my parents, Hong Yin and Yanlin Liu, who encouraged me to start my PhD program and explore my intellectual limit. I would like to express my special thanks to my wife, Xiangyi Yan, for her endless support during my doctoral study. I could not have been through the tough moments and finished my PhD program without her.

TABLE OF CONTENTS

Acknowledgments	v
LIST OF TABLES	ix
LIST OF FIGURES	x
SUMMARY	xii
I INTRODUCTION	1
1.1 Multivariate Time-series Monitoring Literature Review	4
1.2 Image Monitoring Literature Review	8
1.3 Feasibility Determination with Bayesian Method Literature Review	12
II DISTRIBUTION-FREE MULTIVARIATE TIME-SERIES MONITORING	14
2.1 Notation and Problem	14
2.2 Distribution-free Multivariate Time-Series Monitoring Procedure	16
2.3 Implementation Issues	19
2.3.1 User-specified Parameters	19
2.3.2 Parameter Estimation	19
2.4 Numerical Results	21
2.4.1 In-Control Performance	24
2.4.2 Out-of-Control Performance	26

2.5	Application: Wafer Etching Process	32
2.5.1	In-control Process	33
2.5.2	Out-of-control Process	35
2.6	Conclusions and Implications	37
III	DISTRIBUTION-FREE IMAGE MONITORING WHEN THE	
	TARGET PATTERN IS A RANK-ONE MATRIX	39
3.1	Problem and Background	39
3.1.1	Problem Definition	40
3.1.2	Motivating Problem	41
3.2	Distribution-free Image Monitoring Procedure	42
3.2.1	Monitoring Statistics	43
3.2.2	Theoretical Analysis	45
3.2.3	Computation Analysis	49
3.3	Numerical Experiments	50
3.3.1	Simulated Data	50
3.3.2	Verifying Theoretical Results for Dependent Entries	55
3.3.3	Baseline Procedures	56
3.3.4	Experiment Results	57
3.4	Battery Coating Process	61
3.5	Conclusion	62
IV	FEASIBILITY DETERMINATION WITH GAUSSIAN PRO-	
	CESS FOR MULTIPLE STOCHASTIC CONSTRAINTS	64
4.1	Problem Formulation	65
4.1.1	Problem and Notation	65
4.1.2	Feasibility Determination for One Constraint	66
4.2	Feasibility Determination for Multiple Constraints	68

4.2.1	VOI of Multiple Constraints	68
4.2.2	Accelerated Computation of VOI	69
4.3	Convergence Analysis	70
4.4	Numerical Results	72
4.4.1	One-Dimensional Example	73
4.4.2	Two-Dimensional Example	76
4.5	Conclusion	78
V	CONCLUSION	79
	Appendices	81
	Appendix A: Appendix for Chapter 2	81
	References	85

LIST OF TABLES

1	Control limit (H) and actual ARL_0 (followed by standard error in the parenthesis) for monitoring different settings of simulated processes with target $ARL_0 = 550$	26
2	ARL_1 /EDD (followed by standard error in the parenthesis) for sparse shifts of VAR(1) processes	27
3	ARL_1 /EDD (followed by standard error in the parenthesis) for uniform shifts of VAR(1) processes	28
4	ARL_1 /EDD (followed by standard error in the parenthesis) for bidirectional shifts (the first variable shifted mean value with positive amount, the second and third variables shifted mean values with negative amount) of VAR(1) processes	29
5	ARL_1 /EDD (followed by standard error in the parenthesis) for sparse and uniform shifts of EVAR(1) processes	30
6	Control limit (H) and actual ARL_0 for various settings of simulated processes with target $ARL_0 = 1000$ (standard deviation in parentheses)	58
7	ARL_1 for out-of-control Type 1 simulated processes (standard deviation in parentheses)	59
8	ARL_1 for various out-of-control Type 2 simulated processes with normal noises (standard deviation in parenthesis)	60
9	ARL_1 for various out-of-control Type 2 simulated processes with non-normal noises (standard deviation in parenthesis)	60
10	Average values of batch size m from 1000 runs	84

LIST OF FIGURES

1	LED pump performing wafer etching.	1
2	An equipment performing battery coating (source: www.youtube.com/watch?v=q9HbHZXEEDs&ab_channel=LGChem)	2
3	Three characteristics of wafer etching process. (a) cross-correlation matrix of measurements at five locations; (b) auto-correlation of X_{5n} ; (c) histogram of X_{5n}	33
4	Individual components of 182 in-control observations (top) and their average values (bottom)	34
5	VAR-Shewhart, VAR-CUSUM, G-MCUSUM and DFMM procedures when the monitored process is in-control	34
6	Individual components of 721 observations (top) and their average values (bottom)	36
7	VAR-Shewhart, VAR-CUSUM and G-MCUSUM procedures when the monitored process is out-of-control	36
8	The DFMM procedure before adjustment (top) and after adjustment (bottom)	37
9	Heatmaps of thickness measurements of the electrode slurry on two different portions of a foil	42
10	Characteristics of battery coating data	42
11	Setup phase for the Distribution-free Image Monitoring Procedure	44
12	Algorithmic statement of the Distribution-free Image Monitoring Procedure	45

13	Three shifts of Δ for Type 1 simulated processes when $d = 1$	53
14	Five shift patterns of Δ for Type 2 simulated processes when $d = 1$	55
15	The DFIM procedure on training data from August/8 12:00 to August/9 0:00	61
16	The DFIM procedure on out-of-control data from August/9 0:00 to August/10 13:00	62
17	The mean functions and feasible region for Example 1.	74
18	PCD vs. n for verifying approximated VOI using an example with a one-dimensional input space and two constraints.	74
19	PCD vs. n for comparing fitting strategies using an example with a one-dimensional input space and two constraints.	75
20	The mean functions and feasible region for Example 2.	76
21	PCD vs. n for verifying approximated VOI using an example with a two-dimensional input space and two constraints.	77
22	PCD vs. n for comparing fitting strategies using an example with a two-dimensional input space and two constraints.	77
23	Overlapping v.s. non-overlapping batches with size m	81

SUMMARY

This thesis consists of three main parts. The first two parts focus on multivariate time-series monitoring that commonly arises in the quality control problems, and the third part considers the feasibility determination via simulation which has a broad range of applications including manufacturing process control.

In Chapter 2, we consider the problem of detecting a shift in the mean of a multivariate time-series process with a general marginal distribution and a general cross- and auto-correlation structure. We propose a distribution-free monitoring procedure that does not need model fitting nor trial-and-error calibration. Control limit of the procedure can be determined analytically, which allows efficient implementation and easy generalization. The main idea is to convert each observation vector into a one-dimensional T^2 quantity that captures cross-correlation. The T^2 quantities form a univariate auto-correlated process, and CUSUM statistics are constructed on the T^2 quantities. Then using the fact that the CUSUM statistics on the auto-correlated process behave as a reflected Brownian motion asymptotically under some conditions, the control limits of the CUSUM procedure are analytically determined by setting the first-passage time of the Brownian motion equal to a target in-control average run length. We compare the performance of our procedure with three baseline procedures on simulated data with various cross- and auto-correlation and real data from a wafer etching process. The proposed procedure delivers actual in-control average run length close to the target and shows comparable or better performance in detecting a shift in mean compared to baseline procedures.

In Chapter 3, we consider an image monitoring problem for a manufacturing process where a series of 2-dimensional images are converted into a series of random matrices and the mean of these random matrices is expected to be a matrix with rank one. Existing image-based monitoring procedures usually assume each compo-

ment in a monitored image process has normally distributed observations, and the observations are independent or following a specific correlation structure. However, a real-world case such as a battery coating process often violates these assumptions. We propose a distribution-free image monitoring procedure to detect a shift in the mean matrix of monitored images. Two monitoring statistics are calculated based on the singular value decomposition technique, and the two statistics are composited into a two-variate vector. Then the two-variate vectors are monitored by the procedure introduced in Chapter 2. The effectiveness of the proposed procedure, measured by average run lengths, is demonstrated using various simulated data and a real-data example from a battery coating process.

In Chapter 4, we consider the problem of finding a set of feasible inputs in the presence of constraints on multiple performance measures when the constraints are stochastic in that the performance measures can only be evaluated via noisy observations. When similar inputs are more likely to have similar performance measures and when each observation is expensive, a Gaussian process (GP) can be employed to model the performance measures. [1] utilizes a GP for feasibility determination with a single stochastic constraint. We extend the Bayesian procedure due to [1] to multiple constraints. The decision on which input to test next to obtain a new observation is based on a value-of-information function but the calculation of the function can take long, hindering the efficient implementation of the Bayesian feasibility check procedure. To accelerate the computation of the proposed Bayesian procedure, we propose another version with an approximation for the value-of-information function that is quick and easy to calculate. We prove the convergence of our proposed procedures and demonstrate the effectiveness of the procedures incorporating both independent and multi-task GPs.

CHAPTER I

INTRODUCTION

Statistical process monitoring has important applications in various industries especially for quality control in manufacturing. Quality control requires efficient monitoring methods that raise an alarm when an abnormality is detected to ensure the manufacturing process operates as intended. Many production lines use sensors and collect data related to the status of equipment (e.g., pressure levels or temperatures) or quality measures. For example, in a wafer etching process, each wafer's depths are measured at five locations as quality measures.



Figure 1: LED pump performing wafer etching.

Battery coating process is another example, where the thickness of a electrode slurry layer is measured within a designated two-dimensional area on the coated foil. Figure 2 shows an equipment performing slurry coating. The two-dimensional

measurements in this type of applications can be treated as image data and monitored by image monitoring procedures.

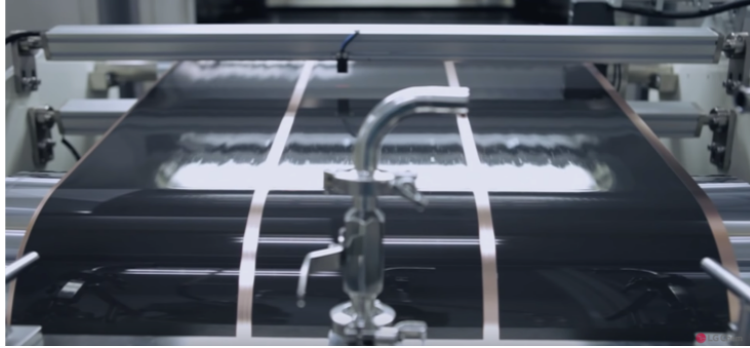


Figure 2: An equipment performing battery coating (source: www.youtube.com/watch?v=q9HbHZXEEDs&ab_channel=LGChem)

As the sensor technology evolves, data acquisition is becoming more efficient and multi-sensor systems such as advanced production process can generate large amount of data streams. This provides opportunities for developing data-driven quality control methodologies, but also makes related statistical process monitoring problem challenging. The challenges of monitoring multivariate data collected consecutively in time are trifold. First, the measurements collected from each product are multivariate data, and they tend to be spatially correlated, such as the depth measurements from different locations of a wafer. Secondly, when items are produced in a short time interval (e.g., every one minute), a temporal correlation exists among measurements. Finally, the measurements rarely follow a common time-series model such as a vector autoregressive (VAR) model, but may have a more general marginal distribution and correlation structure. One focus of this thesis is developing distribution-free monitoring procedures to address these challenges. Specifically, we propose the following two monitoring procedures:

- We propose a monitoring procedure, DFMM, to effectively monitor a general multivariate time-series process. Our procedure (1) captures cross-correlation by converting each multivariate observation into a one-dimensional T^2 statis-

tic and autocorrelation by using, so called, the variance parameter which provides the asymptotic behavior of the sample mean of autocorrelated data; (2) is distribution-free as the monitoring statistic behaves like a reflected Brownian motion asymptotically under some conditions; and (3) determines control limits analytically as ARL_0 can be approximated by the first-passage time of the reflected Brownian motion. There are the main contributions of the paper. Although the proposed methodology is motivated by the common challenge in quality control of manufacturing, it is generalizable for monitoring multivariate time-series processes from other applications.

- We propose a distribution-free procedure, DFIM, to monitor an image process with cross- and auto-correlation when the target image is a matrix of rank one. The proposed procedure does not assume any specific correlation structures nor a specific marginal distribution. It represents the target pattern by a matrix of rank one and the observed images by random matrices. We apply the SVD technique to the target pattern and to the random matrices constructing two different statistics. More specifically, we obtain the pair of singular vectors corresponding to the unique singular value from the SVD of the target pattern of rank one, and multiply the singular vectors with random matrices representing images to get the first statistic. When the process is in control, this statistic is expected to have mean equal to the singular value of the target pattern. We obtain the second statistic by performing SVD on each residual matrix obtained by subtracting the target pattern from each random matrix and taking the largest singular value of the residual matrix. When the process is in control, the residual matrix is expected to be a random matrix with mean zero; otherwise it has the non-zero mean and its singular value would have a different mean value from its in-control case. As these two statistics detect different types of shifts, we combine them into a two-variate statistic vector, and monitor a series of the

statistic vectors using the DFMM procedure introduced earlier.

Manufacturing processes often have various inputs that can be adjusted to achieve different production outcomes. When a process is identified to be out-of-control, a natural follow-up question is how to tune the related inputs to obtain desired quality measures. This can be modeled as a feasibility determination problem which is another focus of this thesis.

Feasibility determination aims to find a set of feasible inputs in the presence of constraints on some performance measures. The constraints are stochastic in that the performance measures can only be evaluated via noisy observations. Feasibility determination with stochastic constraints arises in many industrial applications where similar inputs are more likely to have similar performance measures and noisy observations are expensive. We propose a Bayesian procedure that employs Gaussian process (GP) to model the performance measures. The procedure chooses input to test next to obtain a new observation based on a value-of-information function to achieve a balanced trade-off between exploration and exploitation.

1.1 Multivariate Time-series Monitoring Literature Review

In a typical statistical process control (SPC) procedure, a monitoring statistic is constructed from multivariate observations, and a shift is detected when the monitoring statistic exceeds a control limit that is designed to produce a target in-control average run length (ARL_0). In this paper, ARL_0 stands for the average run length until an alarm is raised when the monitored process is in control. Most existing procedures for multivariate observations are designed under the assumptions of normal observations and no correlation across measurements (i.e., cross-correlation) or in time (i.e., auto-correlation). However, multivariate observations often exhibit non-normality and both cross- and auto-correlation. These properties complicate the development of a procedure that delivers a targeted ARL_0 , and deteriorate the detection of anomalies

when such procedures are applied to a multivariate time-series process. For example, [2] and [3] show that the false alarm rate increases if a procedure developed for uncorrelated observations is applied to auto-correlated observations.

Due to the increasing importance of monitoring a time-series process with correlation, procedures are developed to tackle this challenge. [4] monitors a multivariate time-series process with both cross- and auto-correlation by fitting a vector autoregressive (VAR) process to the observations. They then construct T^2 statistics of residuals and apply Hotelling's T^2 procedure. If the VAR process fits well, then the residuals become independent multivariate normal observations, and thus the proposed scheme should work effectively. When residuals are not normally distributed, it raises false alarms at an unintended rate. [5] proposes a VAR-based monitoring procedure that applies a nonparametric scheme to the residuals calculated from a fitted VAR process. Their proposed procedure is shown to outperform a VAR-based T^2 and exponentially weighted moving average (EWMA) procedures for a tool condition monitoring scenario of titanium alloy milling. [6] proposes the Z-chart to handle multivariate time-series correlation by assuming the observations can be modeled as a VAR process, and the monitoring statistic is constructed directly based on original observations instead of residuals of the fitted VAR process. [7] introduces a group of multivariate CUSUM procedures that fit a VAR process to the observations and construct statistics either based on the observations or the residuals. In recent work, [8] proposes a new monitoring statistic for detecting a change-point in a spatio-temporal process in the presence of cross- and auto-correlation. Still, they assume that observations follow a VAR or vector autoregressive moving-average (VARMA) process.

The procedures reviewed so far are model-based procedures and usually assume that the process to be monitored has a normal marginal distribution. If the monitored process indeed follows an assumed model such as a VAR process, the decision-maker can avoid model fitting but focus on parameter estimation of the assumed model. In

addition, as the monitoring statistic is constructed from i.i.d. data, the control limits can be determined analytically by standard SPC techniques. These are the properties that we pursue, but the model-based procedures have serious limitations. Although these model-based procedures are useful and effective for the problems considered in their specific studies, their performances highly depend on goodness-of-fit to an assumed model, and they may perform poorly when the underlying assumption does not hold. One may claim that a general multivariate time-series model can be fitted in that case. However, the topic itself is a vast research field in the simulation community, and only a few limited techniques are available. For example, [9] proposes a vector autoregressive to anything (VARTA) technique that fits data to a multivariate time-series model with general marginal distributions and general cross- and auto-correlation structures. The VARTA technique tends to fail when the number of observations is small, or the cross- or auto-correlations are out of a specific range. There are additional works on the topic, including [10] and [11], but it takes time to find an appropriate model using these techniques and they do not scale well as the number of components or the dimensionality increases. Even when a reasonably good multivariate time-series model is found, control limits need to be determined by time-consuming trial-and-error simulation to achieve a target ARL_0 . In a production site with many facilities, the number of processes that need to be monitored can be large, and it would be inconvenient to use a control chart that needs time-consuming model fitting and trial-and-error calibration of control limits for a given target ARL_0 . Thus, it is desirable to have a control chart that produces a short detection delay with a targeted ARL_0 (to control false alarm rates) while not requiring any model fitting or trial-and-error simulation for control limit calibration.

There have been efforts to develop distribution-free monitoring procedures which do not rely on whether a specific model fits well observations from a monitored process. [12] and [13] propose distribution-free CUSUM procedures for monitoring gen-

eral univariate auto-correlated processes. Using the fact that the CUSUM statistics constructed from a univariate auto-correlated process behave like a reflected Brownian motion asymptotically, their procedures determine control limits analytically for a given target ARL_0 . For multivariate observations, [14] develops distribution-free versions of multivariate CUSUM procedures that do not require the multivariate normality assumption. However, their procedures do not handle auto-correlation in the monitored process and require trial-and-error simulation to set control limits.

There are procedures designed for spatio-temporal processes which could be considered as multivariate time-series processes but in a higher dimension. [15] proposes distribution-free multivariate CUSUM procedures based on the generalized likelihood ratio test principle to detect a shift of a spatial cluster in a spatio-temporal process. [16] considers a similar problem as in [15] and proposes a CUSUM procedure with control limit determined analytically under the normal marginal assumption. Their methodology is extended to handle general marginal distribution by [17]. [18] proposes a series of reduced-dimension monitoring procedures to handle cross-correlation existing in the spatio-temporal surveillance problem. Although these procedures claim to consider spatio-temporal processes, they only handle spatial correlation but assume no auto-correlation between observations. The critical difference between our study and these reviewed work is that we consider the problem of monitoring a process with both cross- and auto-correlation in observations. [19] proposes another distribution-free CUSUM procedure for monitoring multivariate processes without auto-correlation, based on log-linear analysis of the observations. This procedure is extended to handle auto-correlation in the latest work of [20] by exploiting a decorrelation algorithm. However, it still requires trial-and-error simulation to set control limits which can be time-consuming and inconvenient if a decision-maker wants to apply the monitoring procedure to many processes.

1.2 Image Monitoring Literature Review

One way to monitor image data is to convert them into long one-dimensional vectors and use profile monitoring. A profile refers a functional relationship between explanatory variables and quality attributes at the variables. As discussed in [21], profile monitoring is naturally extended to image monitoring since an image can be described as a profile between location variables within the image and quality attributes at the locations. In the example of a battery coating process, locations on the foil and thickness measurements at the locations can be treated as a profile, and a target profile is simply a vector of constants over the designated locations. Many profile monitoring procedures are developed based on standard statistical process control (SPC) procedures to test the null hypothesis that the mean of long one-dimensional vectors is equal to the target profile. [22] and [23] provide expository discussions about applying SPC to profile monitoring. A majority of profile monitoring procedures are designed for applications where the profile can be well represented by a parametric regression model, either linear or nonlinear. A group of profile monitoring procedures based on fitting the profile data with linear regression models are proposed by [24], [25], [26] and [27]. Studies concerned with polynomial and other nonlinear profile monitoring include [28], [29], and [30]. Instead of fitting a regression model, some studies consider data reduction techniques such as principle component analysis (PCA) and independent component analysis (ICA) to help construct control charts for profile monitoring. [31] conducts a comparison study of profile monitoring procedures based on regression and those based on PCA.

The aforementioned procedures are demonstrated to be effective in their specific applications, but they are designed for situations where no cross-correlation (i.e., component-wise correlation) exists among measurements in a profile, and no auto-correlation (i.e., time-wise correlation) exists among successive profiles. In addition,

measurements are assumed to be normally distributed. Monitoring a process with cross-correlation with a procedure that ignores cross-correlation could easily result in errors. For example, computing T^2 statistics without correctly addressing the covariance between components will generate wrong results. It is well known that auto-correlation increases the false alarm rate if a procedure developed for independent measurements over time is applied to auto-correlated measurements as discussed in [2] and [3]. [32] shows that the non-normality of measurements also increases the false alarm rate of profile monitoring procedures designed for normally distributed measurements.

In order to handle possible cross-correlation, [33] combines PCA with linear regression to remove cross-correlation among measurements in a profile. [34] and [35] develop wavelet-based distribution-free profile monitoring procedures. Their procedures are successfully applied to a lumber manufacturing process where each profile contains 2048 thickness measurements of a sawed board. They are capable to address general cross-correlation structure and general marginal distribution of measurements, but they still assume no auto-correlation among profiles. In order to handle possible auto-correlation, [36] proposes a T^2 chart to monitor a change in estimated parameters after fitting a regression model to profiles. Specifically, the cross-correlation among measurements is handled by constructing the U statistic of the estimated parameters. [37] proposes a multivariate EWMA chart to monitor the change in the estimated regression parameters, where cross-correlation among measurements is also taken into account when constructing the charting statistic. These two procedures both need the assumption that the monitored process follows a first-order auto-regression model with a normal marginal distribution. The assumptions on the cross-correlation structure and the marginal distributions limit their use to broader applications.

Instead of transforming image data into profiles, which are one-dimensional vec-

tors, there have been efforts to develop image-based monitoring procedures by exploiting the characteristics of two-dimensional (2D) image data. [21] summarizes a number of image-based monitoring procedures in different categories. Most image-based monitoring procedures can handle cross-correlation among measurements in random matrices but do not consider auto-correlation among matrices. [38] takes image data and characterizes them by a quantile-quantile (Q-Q) plot. Then they use a profile monitoring procedure to monitor the Q-Q plot. They assume independent images, assuming no auto-correlation over time, and the measurements in the image follow a normal distribution. [39] proposes a monitoring framework that represents each original image in a form of random matrices, applies singular value decomposition (SVD) to the random matrix, and reconstructs the image using a subset of singular values and corresponding singular vectors to obtain a matrix with a lower rank than the original matrix. After reconstruction, the repetitive textured parts that are not of monitoring interest can be removed and the anomalies that are required to be detected can be preserved. Their procedures still do not consider possible auto-correlation among images. [40] considers a monitoring problem of images with a smooth background, and proposes a smooth-sparse decomposition algorithm to capture an anomaly which is assumed to be sparse. [41] proposes to model images by tensors. They utilize a low-rank tensor decomposition for dimension reduction and feature extraction, and monitor the extracted features using T^2 and Q control charts. Their methodology does not require measurements in the image data (represented by random matrices) to follow normal distributions, and is able to handle both cross- and auto-correlation in the image data. However, their procedure requires the estimation of an empirical distribution of monitoring statistics in order to determine the control limit. Some recent works, such as [42] and [43], utilize wavelet transformation to extract features from 2D image data before applying SPC methodologies. [42] proposes to use a generalized likelihood ratio (GLR) control chart to monitor features

extracted by wavelet transformation. They assume the measurements in the image data are independent and follow normal distributions. [43] extends the methodology of [42] by integrating it with a regression-based parametric model which is designed to handle potential temporal or spatial correlation in the data, but they still assume that data follow normal distributions. [44] considers image monitoring applications with a specific target pattern such as uniformity, and assumes only one defect exists in an image. They propose to divide an image into a set of smaller regions and monitor the vector of average intensities computed from the regions using a GLR monitoring procedure. Besides detecting a defect, their procedure can also provide diagnostic information such as where in the image a defect happened. The framework of [44] is extended by [45] to detect multiple defect clusters in an image. [46] considers a similar way to divide the image into regions and monitor the means of regions based on the one-way analysis of variance (ANOVA), then applies a statistical test to identify the location of the defect. [47] proposes an EWMA procedure for similar image monitoring problems as those considered in [44] and [45], and integrates a region growing algorithm in their monitoring framework to provide diagnostic information about how large the defect cluster is. However, all these procedures reviewed so far assume no auto-correlation among images over time. Some spatiotemporal monitoring charts in environment and public health surveillance, such as those discussed by [15], [16] and [17], can also be applied in image monitoring. One key difference between applications of spatiotemporal monitoring and image monitoring such as the motivating problem of this paper is that the latter usually requires fast computation of monitoring statistics to achieve real-time detection. On the other hand, spatiotemporal monitoring has a longer time interval between data acquisition and thus allows monitoring statistics that are expensive to compute. In addition, spatial information often plays a bigger role in spatiotemporal monitoring which by its nature concerns the detection of spatial changes, while image monitoring is more focused on the quick

detection of abnormality in a timely manner.

More and more image-based monitoring applications demonstrate non-normality and high cross- and auto-correlation in images, which violate the assumptions commonly employed in many existing monitoring procedures in the literature. We demonstrate this point using data from a battery coating process in Section 3.1.2. Among the aforementioned procedures, [41] can handle general cross- and auto-correlation structures and non-normality, but the determination of its control limits is time-consuming as it requires the estimation of an empirical distribution of monitoring statistics. This is another common limitation of procedures discussed so far. When a chart requires either the distribution estimation of monitoring statistics or trial-and-error simulation for the determination of control limits, it is not only time-consuming but also makes it impractical to implement the chart when a limited amount of in-control data is available. In this paper, we propose a distribution-free procedure to monitor an image process with cross- and auto-correlation when the target image is a matrix of rank one. The proposed procedure does not assume any specific correlation structures nor a specific marginal distribution. In addition, the control limit of our procedure can be analytically determined for a pre-specified in-control performance measured by the in-control average run length (ARL_0).

1.3 Feasibility Determination with Bayesian Method Literature Review

For an optimization problem with expensive noisy observations, the Bayesian approach with a Gaussian process (GP) has been adopted broadly to identify an input with the maximal or minimal performance measure. For example, see [48], [49], [50], and [51]. In Bayesian optimization, a GP is assigned as a surrogate model for the performance measure function over inputs. The decision on which input to test next to obtain a new observation is made based on the so-called acquisition function to achieve a balanced trade-off between exploration and exploitation. The Bayesian

approach has received attention in recent studies of feasibility determination. [52] designs sequential sampling schemes to allocate a finite budget to determine which of several inputs have a single performance measure exceeding a known threshold. [53] proposes the multi-step-look-ahead procedure for feasibility determination with a single constraint where a Markov process models the underlying performance measure function over all inputs. [54] proposes a new reward function called the normal reward function for feasibility determination using the Bayesian approach. [1] proposes a procedure incorporating a GP for solving feasibility determination with a single constraint. However, all reviewed works consider a single stochastic constraint.

Bayesian feasibility determination is similar to a Bayesian binary classification problem. [55] provides detailed discussions and reviews methods for Bayesian classification problem. However, the existing works more focus on developing a classifier for a given number of samples and do not consider which input to sample next to improve the classification.

CHAPTER II
DISTRIBUTION-FREE MULTIVARIATE TIME-SERIES
MONITORING

Process monitoring aims to detect an anomaly and raise an alarm as quickly as possible to ensure a monitored process operates as intended. Many production systems collect observations related to quality measures of each product either manually or by sensors, and the observations often form a multivariate time-series process that can be monitored by a statistical method. For example, in a wafer etching process, depth measurements are observed at different locations of each etched wafer and a series of observations, each consisting of several depth measurements, is multivariate time-series.

For effective monitoring for a general multivariate time-series process, it is desirable to have a distribution-free monitoring procedure that can handle general marginal distributions and general cross- and auto-correlation structures. Also, to avoid heavy modeling and time-consuming trial-and-error simulation for the calibration of control limits, we desire an analytical method that determines control limits for a given target false alarm rate. In this chapter, we propose a monitoring procedure that achieves these desirable properties.

2.1 Notation and Problem

Let $\mathbf{X}_n = (X_{1,n}, X_{2,n}, \dots, X_{p,n})^\top$ represent the $p \times 1$ vector consisting of the p components from the n th observation. The process to be monitored is a p -dimensional discrete-time multivariate process $\{\mathbf{X}_n : n = 1, 2, \dots\}$ following a general marginal distribution F with mean vector $E[\mathbf{X}_n] = \boldsymbol{\mu} \in \mathbb{R}^p$ and positive definite lag- ℓ covari-

ance matrix $\mathbf{\Lambda}_\ell = \text{Cov}(\mathbf{X}_n, \mathbf{X}_{n+\ell}) = \text{E}[(\mathbf{X}_n - \boldsymbol{\mu})(\mathbf{X}_{n+\ell} - \boldsymbol{\mu})^\top]$ for $\ell = 0, \pm 1, \pm 2, \dots$. We denote the marginal covariance matrix $\mathbf{\Lambda}_0 = \text{Cov}(\mathbf{X}_n, \mathbf{X}_n)$ as $\boldsymbol{\Sigma}$.

Note that we do not assume a multivariate normal marginal distribution but assume a general stationary distribution F for \mathbf{X}_n when the process is in control. Further, $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$ represent the in-control mean vector and an out-of-control mean vector, respectively. We assume the marginal distribution family F and lag- ℓ covariance matrices $\mathbf{\Lambda}_\ell$ for $\ell = 0, \pm 1, \pm 2, \dots$ do not change but the mean vector $\boldsymbol{\mu}_0$ is shifted to $\boldsymbol{\mu}_1$ when a monitored process is out of control. Covariance matrices $\mathbf{\Lambda}_\ell$ for $\ell = 0, \pm 1, \pm 2, \dots$ are assumed to be positive definite. Our goal is to develop a monitoring procedure that raises an alarm as soon as possible after a shift in the mean occurs.

For an observation \mathbf{X}_n , let Y_n denote the T^2 statistic with respect to the in-control mean:

$$Y_n = (\mathbf{X}_n - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1} (\mathbf{X}_n - \boldsymbol{\mu}_0). \quad (1)$$

Then we can define a univariate process $\{Y_n : n = 1, 2, \dots\}$ from $\{\mathbf{X}_n : n = 1, 2, \dots\}$ and our problem becomes detecting a mean shift in the univariate process $\{Y_n : n = 1, 2, \dots\}$. Let ν_0 and ν_1 represent the in-control and out-of-control mean value of Y_n . We denote the marginal standard deviation and *variance parameter* of $\{Y_n : n = 1, 2, \dots\}$ as σ_Y and Ω_Y^2 . Note the *variance parameter* is defined as:

$$\Omega_Y^2 \equiv \lim_{n \rightarrow \infty} n \cdot \text{Var} \left(\frac{\sum_{i=1}^n Y_i}{n} \right),$$

which provides the asymptotic behavior of the sample mean of size n as n goes to infinity. The variance parameter is known to be a useful measure for the variability of the sample mean when auto-correlation exists in the process. Moreover, the univariate process $\{Y_n : n = 1, 2, \dots\}$ is assumed to satisfy the following Functional Central Limit Theorem (FCLT):

Assumption 1. *When the process $\{Y_n : n = 1, 2, \dots\}$ is in control, there exist finite*

real constants ν_0 and $\Omega_Y^2 > 0$ such that as $n \rightarrow \infty$, the sequence of random functions $\{\mathcal{C}_n(\cdot) : n = 1, 2, \dots\}$ converges in distribution to standard Brownian motion $\mathcal{W}(\cdot)$ in the Skorohod space $D[0, 1]$ where the standardized CUSUM, $\mathcal{C}_n(t)$ of $\{Y_n : n = 1, 2, \dots\}$ is defined as

$$\mathcal{C}_n(t) \equiv \frac{\sum_{i=1}^{\lfloor nt \rfloor} Y_i - nt\nu_0}{\Omega_Y \sqrt{n}} \quad \text{for } t \in [0, 1]. \quad (2)$$

Formally,

$$\mathcal{C}_n(\cdot) \xrightarrow[n \rightarrow \infty]{D} \mathcal{W}(\cdot)$$

where $\xrightarrow[n \rightarrow \infty]{D}$ represents convergence in distribution as $n \rightarrow \infty$.

One can refer to [56] for a more detailed discussion about when the FCLT holds. Chapter 4.4 of [57] states that from a practical perspective, it is usually reasonable to act as if the FLCT is valid as long as Ω_Y^2 is finite. Many statistics are formulated to estimate the variance parameter in the literature of steady-state simulation. See [58] for some examples.

2.2 *Distribution-free Multivariate Time-Series Monitoring Procedure*

This section formally formulates our distribution-free multivariate time-series monitoring (DFMM) procedure. The main idea is to convert the p -dimensional process $\{\mathbf{X}_n : n = 1, 2, \dots\}$ into a univariate process $\{Y_n : n = 1, 2, \dots\}$ by calculating the T^2 statistic according to (1). Then we focus on monitoring any mean shift in the univariate process $\{Y_n : n = 1, 2, \dots\}$ and use the distribution-free CUSUM procedure proposed by [12]. Specifically, the one-sided CUSUM statistic is computed as

$$S^+(n) = \begin{cases} 0, & \text{if } n = 0, \\ \max\{0, S^+(n-1) + (Y_n - \nu_0) - K\}, & \text{if } n = 1, 2, \dots \end{cases} \quad (3)$$

where K is a positive real-valued constant such that $K = k\sigma_Y$ for $k > 0$. For i.i.d. normal observations, $k = 0.5$ is usually used according to [59]. [12] suggests $0.01 \leq k \leq 0.1$ for a general auto-correlated process where a smaller k is preferred for large auto-correlation or highly non-normal observations. With a pre-specified control limit $H > 0$, the procedure raises an alarm at the index $T^* = \min\{n : S^+(n) \geq H \text{ and } n = 1, 2, \dots\}$.

Theorem 2.2.1. *If $\{Y_n : n = 1, 2, \dots\}$ satisfies Assumption 1 where $Y_n = (\mathbf{X}_n - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{X}_n - \boldsymbol{\mu}_0)$ for in-control process $\{\mathbf{X}_n : n = 1, 2, \dots\}$, then*

$$S^+(n) \stackrel{\text{D}}{\approx} \mathcal{B}(n) - \inf\{\mathcal{B}(u) : 0 \leq u \leq n\}$$

where $\stackrel{\text{D}}{\approx}$ denotes approximate (asymptotically exact) equality in distribution and $\mathcal{B}(\cdot)$ denotes Brownian motion with drift defined as $\mathcal{B}(t) = d_Y t + \Omega_Y \mathcal{W}(t)$ for $d_Y = -K$ and $t \in [0, \infty)$.

The proof is similar to the one given in [12]. This theorem implies that when $\{S^+(n) : n = 1, 2, \dots\}$ is in control, (a) it behaves approximately like a reflected Brownian motion with drift $-K$ and (b) the expected value of T^* (i.e., ARL_0) for a reasonably large H can be approximated by the expected first-passage time of the reflected Brownian motion. Instead of approximating ARL_0 for a given H , [12] searches the value of H for a user-specified target ARL_0 from the following nonparametric approximation:

$$\frac{\Omega_Y^2}{2K^2} \left\{ \exp \left[\frac{2K(H + 1.166\Omega_Y)}{\Omega_Y^2} \right] - 1 - \frac{2K(H + 1.166\Omega_Y)}{\Omega_Y^2} \right\} \approx \text{target } \text{ARL}_0. \quad (4)$$

The left-hand side of equation (4) is the expected first-passage time of a reflected Brownian motion with drift $-K$ through H . Therefore, H determined by equation (4) is expected to deliver actual ARL_0 close to the target when the monitored process is in control and the target ARL_0 is sufficiently large.

The full algorithmic statement of DFMM is given in Algorithm 1.

<p>Input: values of $\boldsymbol{\mu}_0, \boldsymbol{\Sigma}, \nu_0, \sigma_Y, \Omega_Y^2$, constant k between 0.01 and 0.1, and target ARL_0</p> <ol style="list-style-type: none"> 1 Set $K = k\sigma_Y$, and calculate H by solving equation (4) 2 Set $S^+(0) = 0$ and $n = 0$ 3 do 4 $n \leftarrow n + 1$ 5 Obtain \mathbf{X}_n 6 Calculate $Y_n \leftarrow (\mathbf{X}_n - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{X}_n - \boldsymbol{\mu}_0)$ 7 Calculate $S^+(n) = \max\{0, S^+(n-1) + (Y_n - \nu_0) - K\}$ 8 while $S^+(n) < H$ 9 Raise an out-of-control alarm
--

Algorithm 1: Distribution-free Multivariate Time-series Monitoring (DFMM) Procedure

Lemma 2.2.2. *If $\boldsymbol{\mu}_0$ is shifted by a non-zero vector $\boldsymbol{\delta}$, then $E[Y_n]$ is also shifted by a positive quantity for a positive definite covariance matrix $\boldsymbol{\Sigma}$.*

Proof. Suppose that the monitored process $\{\mathbf{X}_n : n = 1, 2, \dots\}$ is in control with $E[\mathbf{X}_n] = \boldsymbol{\mu}_0$. The in-control mean of Y_n is

$$\begin{aligned} \nu_0 = E[Y_n] &= E[(\mathbf{X}_n - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{X}_n - \boldsymbol{\mu}_0)] = E[\text{tr}(\boldsymbol{\Sigma}^{-1}(\mathbf{X}_n - \boldsymbol{\mu}_0)(\mathbf{X}_n - \boldsymbol{\mu}_0)^\top)] \\ &= \text{tr}(\boldsymbol{\Sigma}^{-1}E[(\mathbf{X}_n - \boldsymbol{\mu}_0)(\mathbf{X}_n - \boldsymbol{\mu}_0)^\top]) = \text{tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}) = \text{tr}(\mathbf{I}_p) = p, \end{aligned}$$

where \mathbf{I}_p is a $p \times p$ identity matrix. The out-of-control mean of $\{Y_n : n = 1, 2, \dots\}$ for $\{\mathbf{X}_n : n = 1, 2, \dots\}$ with $E[\mathbf{X}_n] = \boldsymbol{\mu}_1 = \boldsymbol{\mu}_0 + \boldsymbol{\delta}$ is

$$\begin{aligned} \nu_1 = E[Y_n] &= E[(\mathbf{X}_n - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{X}_n - \boldsymbol{\mu}_0)] = E[\text{tr}(\boldsymbol{\Sigma}^{-1}(\mathbf{X}_n - \boldsymbol{\mu}_0)(\mathbf{X}_n - \boldsymbol{\mu}_0)^\top)] \\ &= \text{tr}(\boldsymbol{\Sigma}^{-1}E[\mathbf{X}_n\mathbf{X}_n^\top - \boldsymbol{\mu}_0\mathbf{X}_n^\top - \mathbf{X}_n\boldsymbol{\mu}_0^\top + \boldsymbol{\mu}_0\boldsymbol{\mu}_0^\top]) \\ &= \text{tr}(\boldsymbol{\Sigma}^{-1}(\boldsymbol{\Sigma} + \boldsymbol{\mu}_1\boldsymbol{\mu}_1^\top - \boldsymbol{\mu}_0\boldsymbol{\mu}_1^\top - \boldsymbol{\mu}_1\boldsymbol{\mu}_0^\top + \boldsymbol{\mu}_0\boldsymbol{\mu}_0^\top)) \\ &= \text{tr}(\mathbf{I}_p + \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)^\top) \\ &= p + (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0) = p + \boldsymbol{\delta}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}. \end{aligned}$$

The above results show that whenever the mean of $\{\mathbf{X}_n : n = 1, 2, \dots\}$ is shifted by $\boldsymbol{\delta}$, there is a corresponding shift of $\boldsymbol{\delta}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}$ reflected in the mean of $\{Y_n : n = 1, 2, \dots\}$. Since $\boldsymbol{\Sigma}$ is assumed to be a positive definite matrix, $\boldsymbol{\delta}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta} > 0$ holds for any

non-zero vector δ . Therefore, the non-zero shift reflected in the univariate process $\{Y_n : n = 1, 2, \dots\}$ is always positive. \square

Lemma 2.2.2 shows that it is suitable to utilize a one-sided univariate monitoring procedure that is capable of detecting a positive shift such as the DFMM procedure.

2.3 Implementation Issues

The DFMM procedure requires several user-specified parameters such as target ARL_0 and constant k , and the estimation of parameters $\boldsymbol{\mu}_0$, $\boldsymbol{\Sigma}$, ν_0 , σ_Y^2 and Ω_Y^2 . In this section, we discuss how to choose or estimate these parameters.

2.3.1 User-specified Parameters

The selection of the target ARL_0 depends on the application at hand. Since ARL is measured as the number of observations, the frequency of collecting observations should be considered when choosing an appropriate ARL_0 in practice. For example, when an observation is recorded every four hours, a target $ARL_0 = 550$ would imply approximately one false alarm every three months. For reference value $K = k\sigma_Y$, [12] recommends $k = 0.1$ when the monitored process is a general auto-correlated process with small to medium lag-one correlation. If the auto-correlation is high or a monitored process exhibits severe non-normality, k smaller than 0.1 but greater than or equal to 0.01 is recommended.

2.3.2 Parameter Estimation

Suppose that we have in-control training observations $\{\mathbf{X}_n^0 : n = 1, 2, \dots, N\}$. Parameters $\boldsymbol{\mu}_0$ and $\boldsymbol{\Sigma}$ are estimated by sample mean vector and covariance matrix of the in-control observations as follows:

$$\boldsymbol{\mu}_0 \leftarrow \bar{\mathbf{X}} \equiv \frac{1}{N} \sum_{n=1}^N \mathbf{X}_n^0 \quad \text{and} \quad \boldsymbol{\Sigma} \leftarrow \hat{\boldsymbol{\Sigma}} \equiv \frac{1}{N-1} \sum_{n=1}^N (\mathbf{X}_n^0 - \bar{\mathbf{X}})(\mathbf{X}_n^0 - \bar{\mathbf{X}})^\top.$$

With these estimated in-control mean vector and covariance matrix, Y_n^0 's which represent T^2 statistics for $n = 1, 2, \dots, N$ are calculated as

$$Y_n^0 \leftarrow (\mathbf{X}_n^0 - \bar{\mathbf{X}})^\top \hat{\boldsymbol{\Sigma}}^{-1} (\mathbf{X}_n^0 - \bar{\mathbf{X}}).$$

Then we estimate ν_0 and σ_Y^2 by sample mean and sample variance of $Y_1^0, Y_2^0, \dots, Y_N^0$:

$$\nu_0 \leftarrow \bar{Y} \equiv \frac{1}{N} \sum_{n=1}^N Y_n^0, \quad \sigma_Y^2 \leftarrow S_Y^2 \equiv \frac{1}{N-1} \sum_{n=1}^N (Y_n^0 - \bar{Y})^2.$$

The estimation of the variance parameter Ω_Y^2 is a little more complicated than the previous parameters. We estimate Ω_Y^2 by the CvM estimator due to [58] using $Y_1^0, Y_2^0, \dots, Y_N^0$. Appendix 1.1 shows how to calculate the overlapping CvM estimator for a given batch size m and Appendix 1.2 gives algorithms to determine the batch size m . Note that the batch size m is used only for the variance parameter estimation from the training dataset. The DFMM procedure monitors the raw observations $\{Y_n : n = 1, 2, \dots\}$ directly rather than batch means, thus there is no detection delay caused by batching.

The appropriate selection of m can be obtained using a batch size determination algorithm proposed by [13]. The batch size determination algorithm finds a batch size m good for a so-called non-overlapping area estimator, one of the standardized-time-series (STS) estimators for the variance parameter. [58] shows that if a batch size m works well for a non-overlapping STS estimator, then the same batch size also works well for an overlapping CvM estimator. Therefore, we can find a batch size m that is good for a non-overlapping STS estimator and use the batch size for the overlapping CvM estimator in the DFMM procedure. Ideally, batch size m needs to be large enough to ensure that STS-weighted-area statistics, denoted by Z_i in Algorithm 3 of the Appendix, are approximately independent and normally distributed. [13]

combines independence and normality tests to find such m . When the training data is small, no m in a possible range may pass the statistical tests. In that case, [13] recommends the largest m that ensures at least 20 non-overlapping batches. If the decision-maker does not want to run the batch size determination algorithm, the largest m with at least 20 non-overlapping batches can be used as a quick heuristic. A formal algorithmic statement of the statistical tests for the selection of m from a training dataset is given in Algorithm 3 of Appendix 1.2.

2.4 Numerical Results

In this section, we compare the performance of the DFMM procedure with the following three procedures:

- VAR-based Shewhart (VAR-Shewhart): [4] presents a VAR-Shewhart procedure that fits a VAR process to observations and constructs T^2 statistics of the residuals of the fitted VAR process. If the underlying process indeed follows a VAR process, then the residuals are independent and normally distributed, and their T^2 statistics are i.i.d. chi-squared distributed. A Shewhart procedure is then used to monitor the i.i.d. T^2 statistics.
- VAR-based CUSUM (VAR-CUSUM): [7] presents a VAR-CUSUM procedure. This procedure also calculates T^2 statistics of residuals of a fitted VAR process. Then CUSUM statistics are constructed on top of the T^2 statistics following equation (3), and the control limits are calibrated carefully by trial-and-error simulation for a given target ARL_0 using the fitted VAR process.
- General multivariate CUSUM (G-MCUSUM): [20] presents the G-MCUSUM procedure. This procedure does not rely on fitting a specific model or distribution to the observations. It implements a decorrelation algorithm and a non-parametric procedure to construct CUSUM statistics introduced by [60]. The

control limits are determined by trial-and-error simulation for a given target ARL_0 .

These three procedures are chosen because they consider both cross- and auto-correlation in a multivariate time-series process. VAR-Shewhart and VAR-CUSUM are model-based procedures, while G-MCUSUM is distribution-free. The DFMM procedure is also distribution-free, but it is different from G-MCUSUM in a sense that it does not require trial-and-error simulation to determine the control limit H for a given target ARL_0 . There are two versions of G-MCUSUM: the first version estimates parameters only once from the training data while the second version updates parameters during monitoring as long as a new observation does not raise an alarm. [61] discusses that updating parameters during monitoring has little benefit when they are already accurate but adds computational burdens and estimation errors if new observations are in fact out of control. In this paper, we use the first version without parameter update because we use large training data (1000 in-control sample paths, each with 10,000 observations) and want to avoid introducing estimation errors.

The three procedures and the DFMM procedure are tested on two simulated multivariate time-series processes, namely VAR(1) and exponential VAR(1) processes that are defined as follows:

- VAR(1) process: a p -dimensional VAR(1) process $\{\mathbf{Z}_n\}$ is defined by

$$(\mathbf{Z}_n - \boldsymbol{\mu}) = \boldsymbol{\Psi}(\mathbf{Z}_{n-1} - \boldsymbol{\mu}) + \boldsymbol{\epsilon}_n, \quad n = 1, 2, \dots, \quad (5)$$

where \mathbf{Z}_n 's are p -dimensional vectors with $E[\mathbf{Z}_n] = \boldsymbol{\mu}$, $\boldsymbol{\Psi}$ is a $p \times p$ coefficient matrix, and $\boldsymbol{\epsilon}_n$ are i.i.d. multivariate normal vectors with mean being a zero vector and covariance matrix $\boldsymbol{\Sigma}_\epsilon$. In our experiment, we set the in-control mean vector $\boldsymbol{\mu}_0$ to be a zero vector. $\boldsymbol{\Psi}$ is set to a diagonal matrix where all diagonal entries equal ϕ . The process covariance matrix $\boldsymbol{\Sigma}$ is chosen to be a symmetric tridiagonal matrix with diagonal entries being one and subdiagonal entries being

ρ . By changing the values of ϕ and ρ , the auto- and cross-correlation in a simulated VAR(1) process can be varied. Note that the performances of VAR-Shewhart and VAR-CUSUM heavily rely on the goodness-of fit of observations to a VAR time-series model. For monitoring a simulated VAR(1) process, we use true parameters of the VAR(1) model for all monitoring procedures.

- Exponential VAR(1) process: an exponential VAR(1) process, which we denote by an EVAR(1) process, is constructed by applying the vector-autoregressive-to-everything (VARTA) framework [9]. Specifically, a p -dimensional base VAR(1) process $\{\mathbf{Z}'_n : n = 1, 2, \dots\}$ is generated according to (5), having marginal means being zeros, diagonal matrix Ψ' with diagonal entries ϕ' , and symmetric tridiagonal matrix Σ' with diagonal entries 1 and subdiagonal entries ρ' . Then define

$$X_{j,n} = F_j^{-1}[\Phi(Z'_{j,n})], j = 1, 2, \dots, p \quad (6)$$

where $Z'_{j,n}$ is the j th component of vector \mathbf{Z}'_n , $\Phi(\cdot)$ is the cumulative distribution function (cdf) of the standard normal distribution, F_j^{-1} is the inverse cdf of the non-normal random variable $X_{j,n}$. By the transformation defined by (6), a multivariate process $\{\mathbf{X}_n : n = 1, 2, \dots\}$ becomes a multivariate time-series process with non-normal marginal distributions. In our experiment, we consider F_j to be the cdf of an exponential random variable with mean as 1. Since the correlation structure Ψ and Σ of process $\{\mathbf{X}_n : n = 1, 2, \dots\}$ is determined by parameters ϕ' and ρ' used in the base process $\{\mathbf{Z}'_n : n = 1, 2, \dots\}$, different values of ϕ' and ρ' are considered to reflect different correlation settings in the process $\{\mathbf{X}_n : n = 1, 2, \dots\}$. For the implementation of the VAR-Shewhart and VAR-CUSUM procedures, we fit a VAR process to the (non-normal) observations \mathbf{X}_n and estimate model parameters of the fitted VAR process. Note that this is what a decision-maker would do when applying these two model-based proce-

dures. We found that the fitted VAR process to \mathbf{X}_n from an EVAR(1) process is a VAR(1) process with the in-control mean vector being a vector of all ones and Ψ and Σ being close to Ψ' and Σ' of the base process $\{\mathbf{Z}'_n : n = 1, 2, \dots\}$, respectively.

The dimension p of all simulated processes is set to five. Auto-correlation parameter ϕ and ϕ' are varied in $\{0.3, 0.5, 0.7\}$, and cross-correlation parameters ρ and ρ' are varied in $\{0.1, 0.7\}$. The target ARL_0 is set to 550 for all cases. For the VAR-CUSUM and DFMM procedures, k is set to $k = 0.05$.

2.4.1 In-Control Performance

In this subsection, we compare procedures based on their behaviors when a monitored process is in control. When observations are in-control, a monitoring procedure's desired performance is to raise a false alarm at a target frequency level measured by ARL_0 . As the VAR-Shewhart and VAR-CUSUM procedures assume that a monitored process is a VAR process, we first test VAR(1) processes. Then to see what happens to actual ARL_0 when the assumption no longer holds, we test EVAR(1) processes.

For each simulated process setting, we pre-generate 1000 in-control sample paths, each with 10,000 observations, as training data for estimating parameters, fitting VAR process and/or calibrating control limits.

For the VAR-Shewhart procedure, the value of H is determined solely based on the chi-squared distribution of residuals under the assumption that the monitored process follows a VAR process. Under this assumption, T^2 statistics of residuals always follow a chi-squared distribution with degrees of freedom p . Thus, the value of H is determined analytically from the chi-squared distribution with target $ARL_0 = 550$. This H is used for both VAR(1) and EVAR(1) processes to get actual ARL_0 . To get actual ARL_0 , residuals are calculated directly from the true VAR(1) model. However, for the EVAR(1) process, a VAR(1) process is fitted first, and then residuals are

calculated from the fitted VAR(1) process.

For the VAR-CUSUM procedure, H is determined by trial-and-error simulation. For a VAR(1) process, we generate observations from the true VAR(1) process and determine H by trial-and-error simulation. For an EVAR(1) process, we first fit a VAR process to the observations generated from the EVAR(1) process and then determine H by trial-and-error simulation using *the fitted VAR process*. We found that the fitted VAR process has parameters ϕ and ρ close to ϕ' and ρ' of the true underlying EVAR(1) process, respectively. Thus, we get similar values of H from the trial-and-error simulation for both VAR(1) and EVAR(1) processes.

For the G-MCUSUM procedure, there is no need to fit any parametric model, but H is determined by trial-and-error simulation directly using observations \mathbf{X}_n from the pre-generated 1000 runs. The DFMM procedure does not require fitting, either. We only estimate parameters $\boldsymbol{\mu}_0, \boldsymbol{\Sigma}, \nu_0, \sigma_Y$, and Ω_Y^2 from the pre-generated 1000 runs using the batch size m reported in Table 10 in the Appendix. Then we take the average of these estimators and determine H analytically by solving equation (4).

We test whether the control limit H determined by the above approaches generates actual ARL_0 close to the target 550 using another 1000 sample paths independent of the training data. Table 1 shows the values of control limits, estimated ARL_0 , and standard errors of the four monitoring procedures under each setting of processes. For the VAR-Shewhart and VAR-CUSUM procedures, we observe that the procedures have actual ARL_0 close to the target 550 when the underlying monitored process indeed follows a VAR process. However, for EVAR(1) processes where the VAR assumption does not hold, both procedures show significantly smaller ARL_0 than the target, raising too many false alarms. Interestingly, control limits of G-MCUSUM calibrated from the 1000 training sample paths result in much smaller ARL_0 than the target 550 on the 1000 new sample paths, and we performed additional calibration by trial-and-error simulation using the new 1000 sample paths (so 2000 sample paths

in total). Control limits with asterisks in the table are those re-calibrated. This phenomenon is due to the large variability of run lengths for the G-MCUSUM procedure. For example, when the VAR(1) process with $(\phi, \rho) = (0.3, 0.1)$ is tested with 1000 runs, the standard error of estimated ARL_0 of G-MCUSUM is about ten times bigger than those of the other three procedures. In other words, to determine H at the same level of the standard error of ARL_0 , the G-MCUSUM procedure would require 100 times more runs compared to the other three procedures. The control limits of the DFMM procedure, which are determined analytically by equation (4), generate ARL_0 close to the target except when the auto-correlation is high (i.e., $\phi = 0.7$). In this case, actual ARL_0 is higher than the target, resulting in a lower false alarm rate.

Table 1: Control limit (H) and actual ARL_0 (followed by standard error in the parenthesis) for monitoring different settings of simulated processes with target $ARL_0 = 550$

Process	VAR-Shewhart		VAR-CUSUM		G-MCUSUM		DFMM	
	H	ARL_0	H	ARL_0	H	ARL_0	H	ARL_0
VAR(1)	(ϕ, ρ)							
	(0.3, 0.1)	19.13 566 (18.31)	50.59 561 (16.59)	35.41*	554 (107.25)	56.06	540 (15.63)	
	(0.3, 0.7)	19.13 524 (16.22)	50.50 530 (15.64)	34.10*	566 (135.14)	56.07	533 (15.86)	
	(0.5, 0.1)	19.13 557 (17.60)	51.21 548 (16.56)	33.33*	551 (131.48)	68.37	550 (15.65)	
	(0.5, 0.7)	19.13 573 (17.86)	50.85 563 (17.03)	34.10*	557 (128.19)	68.29	554 (15.87)	
	(0.7, 0.1)	19.13 572 (18.02)	51.85 578 (16.80)	34.09*	584 (141.21)	95.98	617 (17.45)	
	(0.7, 0.7)	19.13 549 (17.79)	51.31 590 (17.50)	34.04*	534 (124.04)	96.43	637 (18.01)	
EVAR(1)	(ϕ', ρ')							
	(0.3, 0.1)	19.13 29 (0.88)	52.27 120 (3.69)	36.71*	557 (105.00)	118.52	533 (16.05)	
	(0.3, 0.7)	19.13 23 (0.68)	51.73 91 (2.71)	36.27*	589 (106.72)	137.46	538 (15.38)	
	(0.5, 0.1)	19.13 31 (0.98)	51.56 104 (3.10)	35.88*	505 (92.09)	146.37	563 (16.57)	
	(0.5, 0.7)	19.13 27 (0.85)	51.73 86 (2.36)	35.47*	561 (117.20)	171.00	611 (18.55)	
	(0.7, 0.1)	19.13 35 (1.09)	51.35 91 (2.65)	37.60*	592 (104.51)	203.09	638 (18.16)	
	(0.7, 0.7)	19.13 30 (0.95)	51.68 77 (2.34)	37.00*	530 (104.12)	241.00	643 (18.28)	

2.4.2 Out-of-Control Performance

In this subsection, we present the performances of the four procedures when a monitored process is out of control. When the observations are obtained from an out-of-control process, monitoring procedures are expected to raise an alarm quickly. The detection capability is measured by ARL_1 and estimated expected detection delay (EDD). ARL_1 is estimated by the average number of observations until an alarm

Table 2: ARL_1/EDD (followed by standard error in the parenthesis) for sparse shifts of VAR(1) processes

(ϕ, ρ)	Δ	VAR-Shewhart		VAR-CUSUM		G-MCUSUM		DFMM	
		ARL_1	EDD	ARL_1	EDD	ARL_1	EDD	ARL_1	EDD
(0.3, 0.1)	1	258(8.60)	232(24.80)	105(2.37)	101(9.01)	14(0.46)	266(9.10)	63(1.16)	59(4.65)
	1.5	194(6.45)	177(20.01)	69(1.24)	61(4.27)	12(0.37)	228(6.98)	42(0.67)	36(2.35)
	2	149(4.66)	138(16.31)	54(0.86)	41(2.62)	11(0.30)	196(6.64)	32(0.46)	25(1.59)
	2.5	116(3.73)	129(13.57)	41(0.63)	33(2.06)	9(7.29)	184(5.72)	25(0.36)	20(1.16)
	3	89(2.92)	86(9.53)	35(0.49)	25(1.32)	9(0.22)	170(5.45)	21(0.28)	16(0.89)
4	65(2.08)	63(7.51)	27(0.33)	18(1.03)	8(0.19)	159(4.77)	16(0.20)	12(0.62)	
(0.3, 0.7)	1	161(4.93)	133(13.91)	102(2.18)	83(5.90)	9(0.26)	191(5.56)	62(1.23)	51(2.99)
	1.5	107(3.52)	116(11.50)	69(1.34)	60(4.66)	8(0.17)	167(4.67)	40(0.71)	34(2.43)
	2	75(2.43)	76(9.53)	51(0.84)	45(3.03)	7(0.17)	146(4.05)	30(0.50)	25(1.45)
	2.5	51(1.62)	46(5.12)	41(0.69)	32(1.99)	7(0.14)	136(3.53)	25(0.40)	21(1.31)
	3	37(1.16)	27(3.97)	34(0.54)	24(1.89)	6(0.12)	126(3.47)	21(0.32)	16(0.84)
4	22(0.71)	18(2.09)	26(0.35)	21(1.34)	6(0.10)	117(3.21)	16(0.23)	13(0.77)	
(0.5, 0.1)	1	310(10.21)	287(29.25)	165(3.85)	146(13.10)	13(0.52)	310(10.72)	76(1.44)	63(5.30)
	1.5	277(8.44)	247(26.51)	118(2.43)	96(7.94)	12(0.36)	246(8.35)	51(0.89)	39(3.06)
	2	234(7.14)	217(25.46)	86(1.74)	62(4.87)	10(0.32)	218(7.60)	40(0.64)	30(2.00)
	2.5	193(6.41)	170(17.68)	69(1.20)	55(3.97)	10(0.27)	201(6.82)	31(0.48)	25(1.59)
	3	152(4.67)	126(14.10)	59(0.99)	39(3.06)	9(0.24)	191(6.05)	26(0.39)	21(1.36)
4	113(3.48)	99(11.70)	43(0.65)	30(2.20)	8(0.20)	170(5.44)	20(0.28)	13(0.76)	
(0.5, 0.7)	1	238(7.17)	171(21.21)	159(3.68)	127(10.88)	11(0.34)	229(7.96)	72(1.44)	55(4.32)
	1.5	172(5.48)	170(21.49)	105(2.33)	85(7.16)	9(0.26)	188(5.93)	49(0.94)	41(2.92)
	2	133(4.21)	107(12.77)	83(1.61)	60(5.61)	8(0.22)	170(5.04)	38(0.68)	29(2.06)
	2.5	101(3.04)	74(8.85)	66(1.23)	46(3.76)	8(0.20)	147(4.79)	30(0.56)	25(1.84)
	3	80(2.52)	51(7.34)	56(0.96)	41(3.25)	7(0.16)	143(4.65)	26(0.44)	21(1.24)
4	50(1.59)	30(4.58)	42(0.68)	30(2.16)	7(0.13)	125(3.92)	20(0.33)	16(0.94)	
(0.7, 0.1)	1	415(13.23)	193(28.64)	288(7.69)	187(22.36)	20(0.98)	489(17.42)	110(2.27)	92(7.59)
	1.5	338(10.49)	214(31.61)	196(4.71)	154(15.45)	16(0.66)	371(13.94)	73(1.33)	64(4.53)
	2	324(10.09)	243(34.16)	167(4.19)	117(13.15)	15(0.54)	318(10.81)	56(0.97)	43(2.91)
	2.5	295(9.23)	174(23.54)	138(3.03)	77(8.22)	13(0.47)	290(8.96)	46(0.75)	37(2.61)
	3	275(8.59)	152(20.64)	112(2.38)	67(7.02)	12(0.40)	254(9.17)	38(0.63)	30(1.80)
4	220(7.04)	148(24.40)	85(1.54)	53(4.76)	11(0.36)	233(7.55)	28(0.45)	24(1.44)	
(0.7, 0.7)	1	335(10.70)	187(25.61)	262(6.67)	155(20.90)	14(0.55)	342(12.14)	100(2.27)	73(7.12)
	1.5	276(8.18)	161(22.16)	200(5.06)	128(13.60)	12(0.43)	265(10.05)	71(1.51)	60(4.29)
	2	224(6.92)	97(19.90)	154(3.57)	90(10.47)	10(0.35)	236(9.04)	53(1.04)	37(2.57)
	2.5	173(5.51)	88(14.96)	120(2.62)	77(8.34)	9(0.27)	209(7.37)	44(0.86)	37(2.94)
	3	176(5.47)	66(11.54)	102(2.06)	58(6.74)	9(0.24)	191(6.63)	37(0.75)	30(2.46)
4	129(4.15)	42(8.59)	81(1.56)	50(5.40)	8(0.20)	169(5.93)	28(0.49)	23(1.53)	

is raised when a monitored process is out of control from the first observation. To estimate EDD, we simulate sample paths where the first 2000 observations are in-control and a shift occurs at the 2001st observation. Thus, EDD is estimated by the average number of observations after the 2000th observation until an alarm is raised. Formally, let T^* be the index when a procedure raises an out-of-control alarm for the first time when monitoring a process $\{\mathbf{X}_n : n = 1, 2, \dots\}$. In addition, let τ represent

Table 3: ARL_1/EDD (followed by standard error in the parenthesis) for uniform shifts of VAR(1) processes

(ϕ, ρ)	Δ	VAR-Shewhart		VAR-CUSUM		G-MCUSUM		DFMM	
		ARL_1	EDD	ARL_1	EDD	ARL_1	EDD	ARL_1	EDD
(0.3, 0.1)	1	258(8.02)	244(24.78)	105(2.20)	94(8.40)	12(0.41)	235(8.05)	65(1.18)	50(3.97)
	1.5	184(5.60)	160(15.58)	72(1.23)	58(3.79)	10(0.30)	178(6.38)	42(0.67)	36(2.15)
	2	152(4.77)	139(13.42)	53(0.84)	40(2.59)	9(0.22)	156(5.36)	31(0.44)	25(1.50)
	2.5	112(3.45)	108(10.73)	42(0.61)	36(2.23)	8(0.20)	133(4.26)	26(0.36)	22(1.28)
	3	95(3.06)	101(11.58)	35(0.49)	25(1.46)	7(0.16)	119(3.99)	21(0.28)	16(0.86)
4	61(2.01)	53(5.67)	26(0.34)	18(0.98)	6(0.12)	101(3.52)	16(0.20)	12(0.62)	
(0.3, 0.7)	1	253(8.42)	214(22.54)	106(2.20)	82(6.44)	11(0.33)	221(8.40)	63(1.14)	46(2.85)
	1.5	177(5.71)	176(19.54)	69(1.34)	47(3.37)	9(0.23)	174(6.57)	41(0.67)	31(1.87)
	2	130(4.08)	112(11.96)	53(0.89)	39(2.40)	8(0.20)	144(4.76)	32(0.49)	26(1.32)
	2.5	107(3.46)	102(11.68)	42(0.65)	32(2.09)	7(0.17)	132(4.47)	26(0.37)	19(1.09)
	3	80(2.47)	70(6.85)	35(0.48)	27(0.85)	6(0.13)	115(3.93)	21(0.28)	18(0.85)
4	54(1.68)	49(4.90)	26(0.33)	18(1.06)	5(0.10)	95(3.31)	16(0.20)	12(0.65)	
(0.5, 0.1)	1	349(10.98)	340(38.37)	168(3.97)	144(12.28)	13(0.51)	283(10.72)	77(1.57)	61(4.46)
	1.5	266(8.64)	303(33.85)	120(2.54)	96(7.10)	10(0.34)	236(8.98)	54(0.94)	42(2.86)
	2	223(6.72)	175(18.45)	87(1.71)	68(5.50)	9(0.25)	198(6.93)	38(0.60)	30(1.92)
	2.5	183(5.99)	182(21.36)	68(1.22)	49(3.84)	8(0.22)	172(6.00)	31(0.49)	24(1.58)
	3	161(4.84)	127(15.29)	58(0.94)	40(2.82)	8(0.19)	153(5.63)	26(0.38)	20(1.22)
4	116(3.55)	84(11.39)	43(0.64)	29(2.27)	7(0.17)	132(4.52)	20(0.27)	15(0.81)	
(0.5, 0.7)	1	317(9.35)	273(22.30)	163(3.76)	153(12.36)	13(0.48)	263(11.01)	75(1.45)	72(4.81)
	1.5	252(7.57)	227(23.61)	114(2.43)	91(8.79)	10(0.31)	214(8.31)	50(0.88)	39(2.70)
	2	208(6.17)	180(21.41)	85(1.54)	61(5.41)	9(0.27)	189(6.84)	40(0.66)	31(1.93)
	2.5	169(5.64)	160(17.66)	67(1.18)	49(4.14)	9(0.25)	161(5.66)	31(0.49)	25(1.59)
	3	144(4.55)	107(12.21)	57(0.89)	40(2.97)	8(0.20)	141(5.14)	26(0.40)	20(1.17)
4	100(3.29)	66(9.05)	42(0.66)	24(1.73)	7(0.16)	120(4.22)	20(0.27)	14(0.87)	
(0.7, 0.1)	1	416(12.25)	263(33.80)	274(7.14)	192(26.06)	18(0.96)	434(14.93)	110(2.39)	88(7.43)
	1.5	368(11.63)	184(22.38)	210(5.27)	170(19.00)	16(0.70)	331(12.66)	74(1.31)	60(4.26)
	2	305(9.84)	220(30.89)	165(3.86)	116(12.42)	13(0.47)	288(10.56)	55(0.93)	44(2.86)
	2.5	288(8.98)	153(22.29)	138(3.19)	94(9.90)	12(0.43)	256(8.88)	46(0.82)	40(3.01)
	3	259(8.14)	162(25.31)	114(2.31)	64(5.64)	10(0.38)	226(8.88)	39(0.66)	32(1.98)
4	225(7.22)	110(16.14)	84(1.55)	44(4.31)	9(0.28)	189(6.21)	29(0.44)	23(1.33)	
(0.7, 0.7)	1	382(11.56)	266(36.37)	265(6.70)	215(22.74)	16(0.76)	393(15.42)	109(2.27)	87(7.18)
	1.5	337(11.21)	231(32.01)	201(5.16)	101(11.38)	14(0.60)	330(12.44)	72(1.34)	56(3.97)
	2	331(10.34)	256(36.09)	164(3.84)	111(12.38)	12(0.41)	284(10.80)	57(1.01)	46(3.63)
	2.5	275(8.56)	175(25.46)	132(2.81)	89(10.19)	12(0.41)	250(8.92)	46(0.79)	37(2.75)
	3	242(7.62)	117(21.24)	109(2.12)	70(7.08)	11(0.37)	213(8.43)	38(0.62)	28(1.85)
4	205(6.47)	102(15.45)	82(1.54)	49(5.05)	8(0.23)	186(6.75)	28(0.46)	23(1.47)	

the change point. Then

$$ARL_1 = E [T^* | \tau = 1] \quad \text{and} \quad EDD = E [T^* | \tau = 2000] - \tau.$$

The big difference between the two is the value of $S^+(\tau - 1)$, the monitoring statistics right before the change point. For ARL_1 , $S^+(\tau - 1)$ is fixed at 0; but for EDD, it is a random variable. ARL_1 measures a procedure's capability to detect a shift when the monitored process starts from the out-of-control status. However, in practice, a monitored process usually starts from the in-control status, and then a shift occurs

Table 4: ARL_1/EDD (followed by standard error in the parenthesis) for bidirectional shifts (the first variable shifted mean value with positive amount, the second and third variables shifted mean values with negative amount) of VAR(1) processes

(ϕ, ρ)	Δ	VAR-Shewhart		VAR-CUSUM		G-MCUSUM		DFMM	
		ARL_1	EDD	ARL_1	EDD	ARL_1	EDD	ARL_1	EDD
(0.3, 0.1)	1	258(7.91)	281(28.67)	102(2.23)	91(6.97)	13(0.43)	233(7.54)	62(1.15)	55(4.58)
	1.5	203(6.52)	166(22.76)	71(1.19)	61(3.69)	11(0.32)	185(5.79)	42(0.65)	35(2.22)
	2	147(4.63)	118(12.75)	53(0.84)	40(2.61)	9(0.25)	159(5.32)	31(0.43)	24(1.42)
	2.5	121(3.75)	130(12.62)	43(0.60)	33(1.86)	8(0.21)	144(4.34)	26(0.35)	20(1.10)
	3	91(2.82)	72(7.63)	35(0.48)	26(1.56)	7(0.17)	133(3.68)	21(0.30)	16(0.88)
4	66(2.18)	54(5.94)	26(0.33)	19(1.05)	6(0.14)	110(3.36)	16(0.20)	12(0.60)	
(0.3, 0.7)	1	236(7.42)	217(20.40)	110(2.24)	89(6.28)	11(0.35)	223(7.55)	63(1.16)	55(3.47)
	1.5	174(5.58)	186(22.95)	68(1.19)	51(3.76)	9(0.26)	183(6.56)	42(0.68)	32(1.89)
	2	131(4.04)	122(12.85)	53(0.86)	40(2.39)	8(0.21)	155(5.32)	32(0.49)	26(1.59)
	2.5	100(3.15)	87(9.39)	42(0.63)	32(2.07)	7(0.17)	143(4.46)	25(0.35)	20(1.19)
	3	85(2.74)	81(7.95)	35(0.49)	26(1.43)	7(0.14)	128(4.21)	22(0.30)	18(0.96)
4	54(1.80)	50(5.54)	26(0.34)	18(1.07)	6(0.12)	108(3.34)	16(0.20)	12(0.63)	
(0.5, 0.1)	1	336(11.02)	312(34.94)	172(4.08)	140(15.42)	12(0.47)	278(11.89)	76(1.44)	54(3.77)
	1.5	257(7.86)	264(25.80)	114(2.40)	83(7.60)	11(0.39)	229(8.08)	52(0.94)	37(2.58)
	2	231(7.41)	154(17.07)	87(1.61)	65(5.45)	9(0.27)	194(7.30)	39(0.63)	32(2.00)
	2.5	188(6.03)	156(15.26)	71(1.25)	52(4.04)	9(0.25)	174(5.80)	33(0.52)	25(1.63)
	3	161(5.30)	142(16.69)	60(1.06)	41(3.64)	8(0.20)	159(5.90)	26(0.37)	20(1.25)
4	119(3.89)	86(12.70)	43(0.64)	30(2.21)	7(0.16)	131(4.49)	20(0.27)	14(0.86)	
(0.5, 0.7)	1	329(10.64)	320(36.08)	168(3.75)	138(11.54)	13(0.51)	275(10.45)	77(1.45)	67(4.90)
	1.5	272(8.17)	230(21.82)	118(2.49)	81(6.23)	11(0.38)	218(7.99)	53(0.95)	42(2.94)
	2	206(6.34)	179(19.34)	91(1.78)	68(5.20)	10(0.31)	188(7.25)	41(0.69)	32(1.97)
	2.5	172(5.53)	151(16.05)	65(1.04)	46(3.54)	9(0.24)	161(5.97)	31(0.48)	26(1.68)
	3	145(4.54)	107(12.13)	55(0.89)	38(2.89)	8(0.21)	150(5.50)	26(0.39)	19(1.12)
4	112(3.52)	103(11.19)	42(0.62)	30(2.12)	7(0.15)	127(4.46)	20(0.27)	17(0.95)	
(0.7, 0.1)	1	432(13.25)	305(37.85)	267(7.37)	210(25.42)	20(0.96)	453(15.70)	110(2.42)	101(9.07)
	1.5	355(11.23)	214(29.80)	208(5.33)	161(18.03)	15(0.66)	349(12.76)	75(1.36)	56(3.29)
	2	328(10.07)	187(28.34)	159(3.63)	116(11.28)	13(0.51)	306(10.96)	57(1.02)	45(3.05)
	2.5	296(10.05)	218(31.24)	138(3.12)	77(9.55)	12(0.41)	266(7.74)	46(0.83)	38(2.55)
	3	246(7.75)	105(18.53)	113(2.30)	70(8.62)	11(0.39)	238(6.78)	38(0.61)	28(1.69)
4	219(6.89)	82(12.84)	83(1.51)	42(4.20)	10(0.31)	207(6.21)	28(0.47)	24(1.42)	
(0.7, 0.7)	1	406(13.25)	277(46.69)	276(7.14)	200(21.08)	18(0.84)	384(13.99)	106(2.12)	81(5.78)
	1.5	343(10.69)	220(28.91)	194(4.68)	132(14.81)	15(0.62)	323(12.55)	73(1.33)	59(4.25)
	2	317(10.51)	165(26.24)	155(3.54)	95(10.14)	13(0.49)	281(10.78)	56(1.00)	48(3.15)
	2.5	293(9.35)	215(33.43)	132(2.90)	88(8.86)	11(0.42)	264(9.24)	46(0.76)	36(2.56)
	3	234(7.29)	128(20.24)	110(2.33)	68(6.72)	10(0.33)	219(8.26)	38(0.64)	30(2.21)
4	196(5.99)	103(18.23)	82(1.51)	41(4.39)	9(0.24)	194(6.70)	29(0.48)	22(1.40)	

later. Thus, EDD is also practically meaningful, which measures a procedure's capability to detect a shift when a monitored process stays in control at the beginning and then goes out of control later.

As in the previous section, we test two multivariate time-series processes: VAR(1) and EVAR(1) processes with $p = 5$. For each process, three types of shifts are considered: (i) sparse shift where a mean shift occurs only at the last variable with $\delta = (0, 0, 0, 0, \delta)$; (ii) uniform shift where a mean shift occurs at all of the variables by

Table 5: ARL_1/EDD (followed by standard error in the parenthesis) for sparse and uniform shifts of EVAR(1) processes

(ϕ', ρ')	Δ	sparse shift				uniform shift				bidirectional shift			
		G-MCUSUM		DFMM		G-MCUSUM		DFMM		G-MCUSUM		DFMM	
		ARL_1	EDD	ARL_1	EDD	ARL_1	EDD	ARL_1	EDD	ARL_1	EDD	ARL_1	EDD
(0.3, 0.1)	1	8(0.18)	156(4.53)	141(3.34)	120(11.24)	8(0.18)	127(3.95)	132(3.17)	97(7.91)	9(0.24)	167(5.03)	156(3.31)	140(11.05)
	1.5	8(0.17)	143(4.22)	93(1.94)	72(5.06)	5(0.10)	88(2.80)	91(2.02)	81(6.37)	7(0.17)	133(3.87)	102(1.71)	81(5.21)
	2	8(0.17)	140(4.32)	73(1.37)	65(3.99)	4(0.06)	68(1.99)	70(1.34)	52(4.07)	6(0.13)	118(3.15)	74(1.12)	62(3.79)
	2.5	7(0.15)	138(4.07)	59(0.95)	48(3.25)	3(0.04)	56(1.72)	57(1.13)	45(3.34)	6(0.11)	106(3.18)	58(0.77)	48(2.48)
	3	7(0.15)	137(3.87)	46(0.73)	36(2.21)	3(0.03)	48(1.42)	47(0.86)	41(3.27)	5(0.09)	99(2.81)	47(0.60)	37(1.93)
4	7(0.15)	136(3.97)	35(0.46)	27(1.51)	2(0.02)	40(1.14)	35(0.54)	27(1.82)	5(0.09)	91(2.57)	35(0.39)	26(1.38)	
(0.3, 0.7)	1	7(0.17)	144(4.58)	166(3.96)	143(10.17)	6(0.12)	108(3.00)	142(3.38)	117(8.94)	7(0.17)	125(4.44)	186(4.27)	151(11.25)
	1.5	7(0.14)	132(4.24)	112(2.33)	90(6.82)	4(0.08)	79(2.39)	102(2.18)	86(6.97)	6(0.13)	101(3.36)	119(2.39)	97(6.72)
	2	6(0.13)	124(4.17)	89(1.58)	71(4.24)	4(0.05)	63(1.97)	76(1.60)	68(4.97)	5(0.11)	94(3.07)	86(1.45)	69(4.28)
	2.5	6(0.13)	121(3.57)	70(1.20)	50(3.53)	4(0.05)	63(1.97)	66(1.24)	51(3.49)	5(0.10)	86(2.77)	69(1.02)	53(2.97)
	3	6(0.13)	120(3.69)	55(0.88)	42(2.87)	3(0.04)	49(1.54)	52(0.93)	40(2.69)	5(0.09)	80(2.66)	56(0.75)	43(2.26)
4	6(0.13)	110(3.52)	41(0.56)	33(1.73)	3(0.03)	42(1.30)	39(0.60)	30(1.69)	4(0.08)	72(2.31)	41(0.51)	34(1.60)	
(0.5, 0.1)	1	9(0.24)	175(5.54)	177(4.33)	160(11.96)	9(0.26)	158(4.92)	154(3.69)	131(13.41)	10(0.30)	194(6.96)	193(4.37)	163(14.15)
	1.5	8(0.17)	158(4.58)	118(2.46)	103(8.02)	7(0.14)	116(3.56)	116(2.67)	97(7.22)	8(0.22)	156(4.84)	126(2.45)	104(7.23)
	2	8(0.17)	147(4.24)	93(1.80)	79(5.23)	5(0.09)	91(2.76)	88(1.89)	74(5.94)	7(0.17)	135(4.13)	92(1.40)	71(4.85)
	2.5	8(0.17)	142(4.35)	72(1.27)	59(4.26)	4(0.07)	75(2.21)	71(1.42)	59(4.55)	6(0.14)	123(3.89)	74(1.07)	55(2.56)
	3	7(0.16)	139(4.06)	60(1.00)	46(2.62)	3(0.05)	66(1.93)	59(1.12)	49(3.10)	6(0.12)	117(3.61)	58(0.79)	48(2.73)
4	7(0.15)	140(4.17)	45(0.70)	38(2.32)	3(0.04)	54(1.56)	44(0.75)	38(2.46)	5(0.11)	105(3.36)	44(0.52)	36(1.74)	
(0.5, 0.7)	1	8(0.22)	163(5.09)	201(4.94)	179(18.47)	7(0.16)	133(4.03)	176(4.39)	153(14.35)	8(0.22)	153(5.61)	226(5.45)	196(16.13)
	1.5	7(0.18)	146(4.66)	135(2.84)	120(9.32)	5(0.10)	103(3.41)	126(3.03)	111(8.93)	7(0.18)	122(4.23)	145(2.98)	116(8.46)
	2	7(0.15)	131(4.60)	107(2.11)	90(6.77)	5(0.08)	86(2.75)	99(2.09)	80(5.87)	6(0.14)	113(3.51)	111(3.01)	94(6.02)
	2.5	6(0.14)	127(4.17)	82(1.47)	64(4.05)	4(0.07)	72(2.30)	79(1.51)	71(4.66)	5(0.12)	106(3.66)	86(1.39)	74(4.45)
	3	6(0.13)	124(3.95)	70(1.18)	62(4.13)	4(0.06)	64(2.24)	67(1.26)	59(3.64)	5(0.11)	96(3.17)	70(0.98)	63(3.34)
4	6(0.13)	119(3.57)	53(0.74)	43(2.52)	3(0.05)	55(1.98)	49(0.81)	39(2.53)	5(0.09)	87(2.93)	53(0.63)	46(2.14)	
(0.7, 0.1)	1	13(0.39)	227(7.58)	222(5.51)	195(19.39)	14(0.40)	204(7.42)	197(5.14)	174(14.37)	15(0.53)	251(9.89)	254(5.86)	203(19.20)
	1.5	11(0.31)	202(7.08)	164(3.61)	149(11.38)	10(0.24)	155(5.08)	147(3.36)	107(8.34)	11(0.36)	193(7.39)	171(3.39)	144(10.39)
	2	10(0.26)	182(6.76)	125(2.58)	107(8.16)	8(0.18)	125(4.14)	123(2.76)	96(7.94)	10(0.28)	175(6.43)	126(2.21)	101(6.56)
	2.5	10(0.25)	178(6.30)	106(2.12)	76(5.45)	6(0.13)	109(3.81)	96(2.01)	78(6.38)	9(0.23)	156(5.76)	100(1.59)	68(4.19)
	3	9(0.23)	168(5.67)	85(1.60)	70(5.02)	5(0.10)	97(3.56)	80(1.65)	74(5.99)	7(0.19)	146(5.16)	82(1.18)	72(4.19)
4	8(0.20)	161(5.34)	62(1.01)	47(2.81)	4(0.07)	78(2.64)	66(1.27)	48(3.30)	7(0.18)	128(4.99)	60(0.80)	43(2.39)	
(0.7, 0.7)	1	11(0.35)	224(7.61)	263(7.09)	246(19.22)	10(0.30)	189(6.17)	228(6.24)	209(19.01)	12(0.41)	210(7.63)	305(7.88)	239(17.03)
	1.5	10(0.30)	189(6.57)	193(4.54)	168(11.39)	8(0.20)	151(5.25)	179(4.32)	171(16.54)	9(0.30)	180(6.58)	227(5.32)	207(17.23)
	2	9(0.26)	177(5.27)	148(2.92)	126(8.68)	6(0.15)	124(3.91)	135(3.21)	101(7.53)	8(0.24)	157(5.17)	159(3.14)	114(8.14)
	2.5	9(0.23)	167(5.11)	119(2.20)	106(6.79)	6(0.13)	111(3.54)	112(2.41)	99(7.42)	7(0.21)	143(4.94)	126(2.24)	114(6.83)
	3	8(0.21)	157(5.37)	99(1.78)	82(4.64)	5(0.10)	97(3.18)	97(2.01)	93(7.82)	7(0.20)	132(4.64)	106(1.77)	87(5.12)
4	8(0.19)	146(4.79)	75(1.23)	66(4.42)	4(0.08)	84(2.61)	72(1.32)	64(4.63)	6(0.16)	115(4.21)	76(1.10)	66(3.65)	

an equal amount to the same direction with $\delta = (\delta, \delta, \delta, \delta, \delta)$; and (iii) bidirectional shift where a mean shift occurs at the first three variables by an equal amount to opposite directions with $\delta = (+\delta, -\delta, -\delta, 0, 0)$. We estimate ARL_0 and ARL_1 using 1000 runs, while EDDs are estimated using 100 runs. All runs are independent. We quantify the shift size by Δ where

$$\Delta = \delta^\top \Sigma^{-1} \delta.$$

For VAR(1) processes, we use the values of H found in Table 1 so that all four procedures have ARL_0 close to the target value 550 for the fairness of comparison in the out-of-control cases. Note that ARL_0 of the DFMM procedure tends to be larger than the target when parameter ϕ is high. It means that one may use a tighter control limit for a given target if she is willing to do the calibration of control limits by trial-

and-error simulation. However, to emphasize the ability of the DFMM procedure that determines control limits analytically, we use the same control limits in Table 1.

Tables 2, 3 and 4 show ARL_1 and estimated EDD for sparse, uniform and bidirectional shifts of VAR(1) processes, respectively. First, both ARL_1 and estimated EDD are smaller for uniform shifts than for sparse shifts in most cases for the same level of shift size Δ . This implies that for the same level of Δ , it is easier to detect a shift when the shift size Δ is distributed over all variables rather than just at one variable. In Table 2, the VAR-CUSUM procedure shows better performances than the VAR-Shewhart procedure, and it is expected because CUSUM-type procedures tend to be more sensitive to small shifts than Shewhart-type procedures. One can also see that although the DFMM procedure has significantly smaller ARL_1 compared to the VAR-Shewhart and VAR-CUSUM procedures, the G-MCUSUM procedure has the smallest ARL_1 for all settings. However, surprisingly, the G-MCUSUM procedure has the largest EDD in many cases among all four procedures, while the DFMM procedure shows the smallest EDD for all settings. We observe a similar tendency in Table 3 and 4 for uniform and bidirectional shifts. By plotting the monitoring statistics of the G-MCUSUM procedure, it is found that the statistics are quite variable at the beginning, which explains the small ARL_1 . However, the statistics become stable and do not respond to a shift quickly once it is stabilized, which explains large EDD values. This phenomenon is caused by the facts that (i) the monitoring statistics of the G-MCUSUM procedure are closely related to the chi-square testing statistics, and (ii) chi-square testing statistics are unstable with a small number of observations but stabilized with a large number of observations.

For non-normal EVAR(1) processes, the VAR-Shewhart and VAR-CUSUM procedures are not considered because their ARL_0 with the control limits found under the VAR assumption are too small, causing too many false alarms. We do not calibrate their control limits, either, because we assume no multivariate time-series fit-

ting in this paper. Thus only the G-MCUSUM and DFMM procedures are compared. ARL_1 /EDD for the three types of shifts are reported in Table 5. Similar to the results of VAR(1) processes, G-MCUSUM has smaller ARL_1 than the DFMM procedure. In terms of EDD, the DFMM procedure has smaller EDD in most cases, but there are a few settings with high cross-correlation parameter (ρ') where the G-MCUSUM procedure shows smaller EDD.

In summary, both DFMM and G-MCUSUM procedures show robust performances to both normal and non-normal multivariate time-series processes as they are distribution-free procedures. However, the G-MCUSUM procedure requires more data and computational effort than the DFMM procedure. In terms of ARL_1 and EDD, the DFMM procedure performs better than the VAR-Shewhart and VAR-CUSUM procedures. Compared to the G-MCUSUM procedure, the DFMM procedure shows larger ARL_1 but smaller EDD. Considering both the easiness of the control limit determination and the performance of ARL_1 and EDD, the DFMM procedure is an appealing, robust, and easy-to-implement procedure.

2.5 Application: Wafer Etching Process

In this section, we consider a monitoring problem for a wafer etching process. Depths at five different locations are measured whenever a produced wafer is sampled. The measurements at five locations over a series of sampled wafers form a five-dimensional process denoted by $\{\mathbf{X}_n = (X_{1n}, X_{2n}, X_{3n}, X_{4n}, X_{5n}) : n = 1, 2, \dots\}$. The purpose of monitoring the wafer etching process is to detect a shift in the mean vector of \mathbf{X}_n as soon as possible after any anomaly occurs. We consider observations collected from two production lines between 08/02/2018 and 01/21/2019.

Figure 3 shows cross-correlations among five locations, an auto-correlation plot, and a histogram of observations collected from one of the five locations. We can see that cross-correlations range from 0.75 to 0.9; an estimated lag-1 correlation

is 0.2; and the histogram of X_{5n} 's does not follow a normal distribution. These findings indicate that the VAR(1) assumption does not hold; thus, we expect the VAR-Shewhart and VAR-CUSUM procedures not to perform well. Nevertheless, we apply all four procedures and set 550 as the target ARL_0 . As the actual data collection frequency is approximately once every four hours, target $ARL_0 = 550$ implies one false alarm every three months on average.

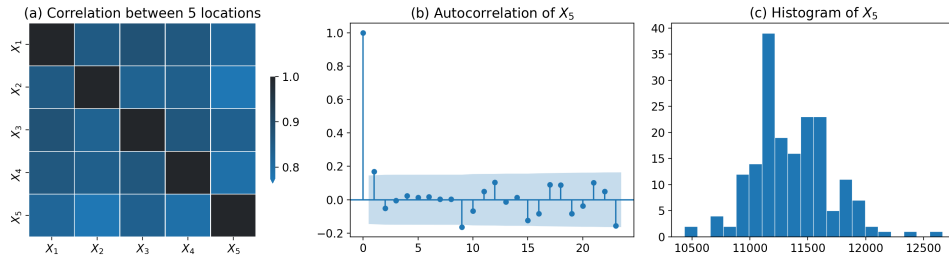


Figure 3: Three characteristics of wafer etching process. (a) cross-correlation matrix of measurements at five locations; (b) auto-correlation of X_{5n} ; (c) histogram of X_{5n}

For the VAR-Shewhart procedure, we fit a VAR(1) process to the collected observations to calculate residuals and determine control limits using a chi-squared distribution with five degrees of freedom. For the VAR-CUSUM procedure, in addition to fitting a VAR(1) process to obtain residuals, we need to estimate the covariance matrix of residuals and find the control limit by trial-and-error simulation from the fitted VAR(1) process. For the G-MCUSUM procedure, a block version of the bootstrap resampling technique by [20] is used to determine its control limit. To implement the DFMM procedure, we need to estimate $\boldsymbol{\mu}_0$, $\boldsymbol{\Sigma}$, ν_0 , σ_Y^2 and Ω_Y^2 using the in-control observations. We set $k = 0.01$ and $m = 50$ based on exploratory experiments. Then we calculate the control limit by (4) with the target ARL_0 equal to 550.

2.5.1 In-control Process

Figure 4 presents the individual components and the corresponding average value of 182 observations collected from production line A, which is believed to be in control

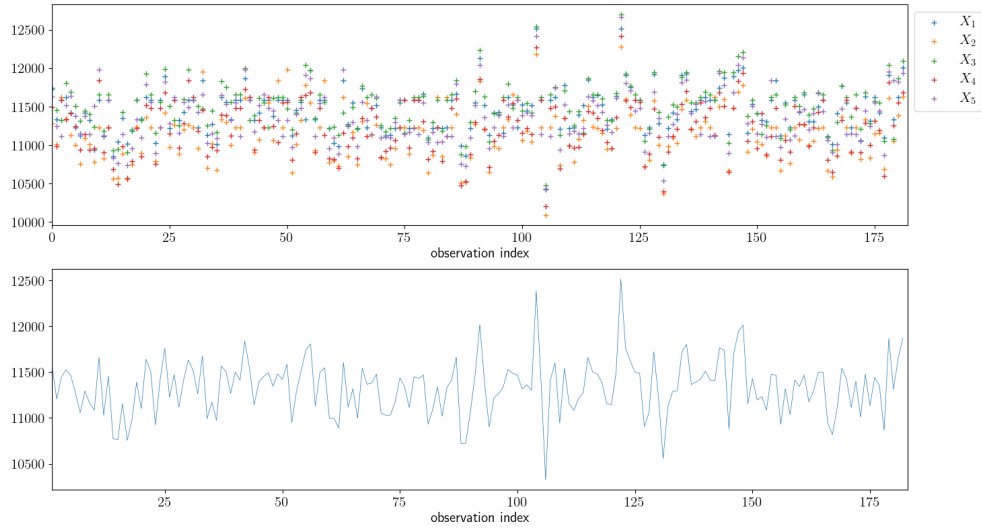


Figure 4: Individual components of 182 in-control observations (top) and their average values (bottom)

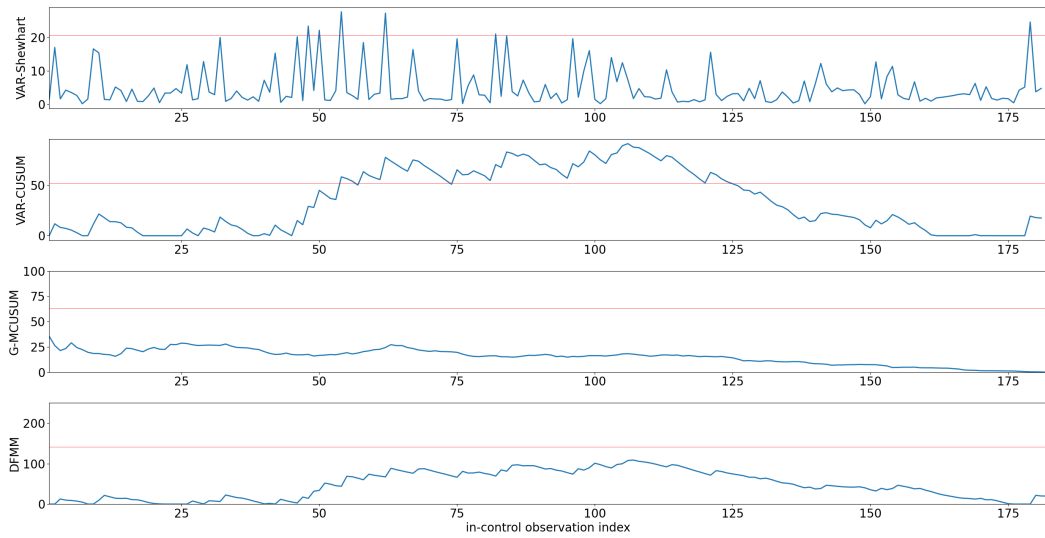


Figure 5: VAR-Shewhart, VAR-CUSUM, G-MCUSUM and DFMM procedures when the monitored process is in-control

during the whole period of interest. This in-control dataset is used to estimate required parameters and determine control limits for the four monitoring procedures. As an examination of in-control performance, the four procedures are then applied to the same dataset with the resulting control limits. Figure 5 shows the monitoring re-

sults of VAR-Shewhart, VAR-CUSUM, G-MCUSUM, and DFMM procedures in the order from top to bottom, where the red lines represent control limits and the blue lines represent monitoring statistics. As shown in the figure, the G-MCUSUM and DFMM procedures do not raise any out-of-control alarms, while both VAR-Shewhart and VAR-CUSUM procedures raise false alarms. This is expected because the VAR(1) assumption does not hold in the training dataset.

2.5.2 Out-of-control Process

Now we consider 721 observations collected from production line B where abnormalities occur during the data collection period. The plot of observations is shown in Figure 6. The record of equipment status sensor data shows that an adjustment was made around 12/01/2018, which corresponds to the 451st observation. After carefully reviewing the equipment sensor data, the management team believes that the process was out of control from 10/14/2018, which corresponds to the 208th observation. After the equipment adjustment performed on 12/01/2018, the process is believed to be back to the in-control status.

The results of applying VAR-Shewhart, VAR-CUSUM, and G-MCUSUM procedures are shown in Figure 7. The red line indicates the control limit, and the blue line represents monitoring statistics. It can be observed that both VAR-Shewhart and VAR-CUSUM procedures raise several false alarms before the 208th observation, which again confirms that these VAR-based procedures are not effective and not reliable in this application. Similarly, the G-MCUSUM procedure raises a strong out-of-control alarm before the 208th observation. This is due to the tendency of the G-MCUSUM procedure being unstable at the early stage of monitoring.

The DFMM procedure is applied to these observations with parameters and the control limit calculated from the in-control observations in Section 2.5.1. Specifically, the procedure is applied to the first 450 observations collected before the equipment

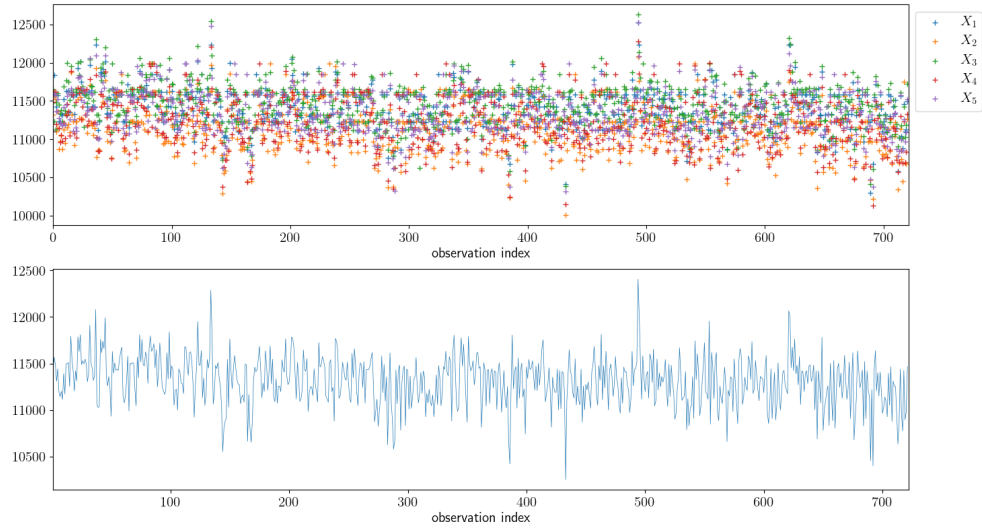


Figure 6: Individual components of 721 observations (top) and their average values (bottom)

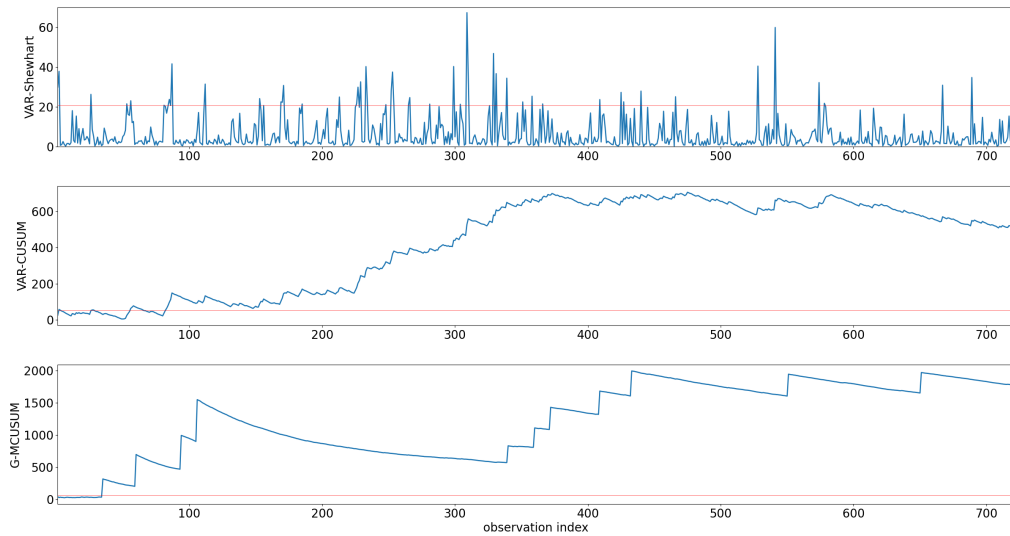


Figure 7: VAR-Shewhart, VAR-CUSUM and G-MCUSUM procedures when the monitored process is out-of-control

adjustment was made to the production line on 12/01/2018. Then, the procedure is restarted and applied to the remaining observations. Figure 8 shows the monitoring results of the DFMM procedure before and after the adjustment in the top and bottom figures, respectively. The DFMM procedure raises a strong out-of-control signal at the

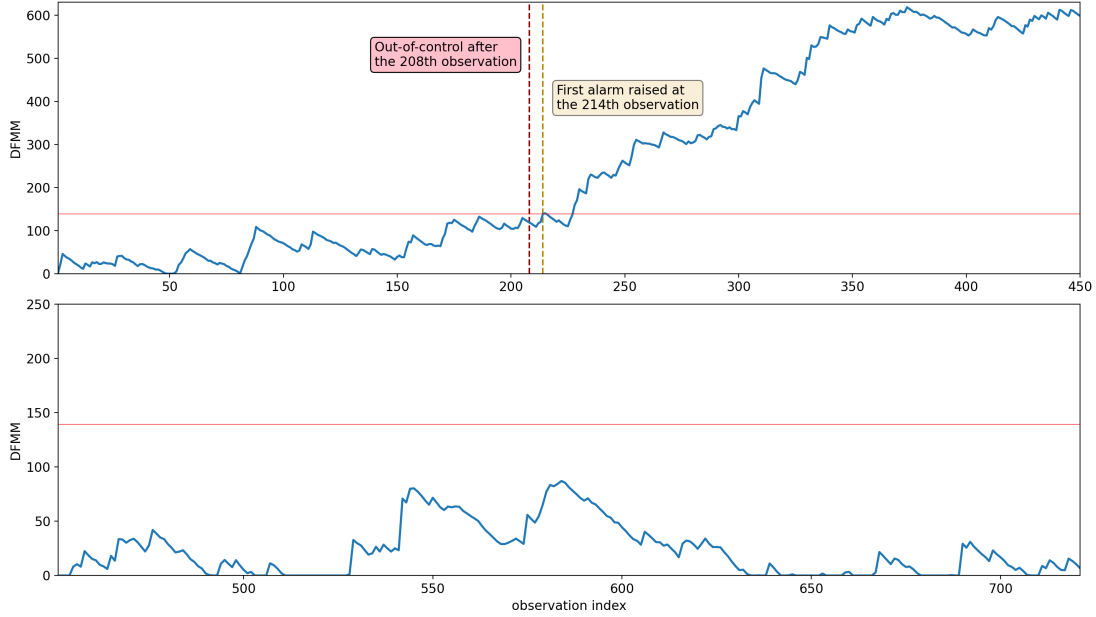


Figure 8: The DFMM procedure before adjustment (top) and after adjustment (bottom)

214th observation, close to when the anomaly occurs. Meanwhile, no out-of-control alarm is raised after the 451st observation.

2.6 Conclusions and Implications

We propose a distribution-free multivariate time-series CUSUM procedure, DFMM, to monitor multivariate observations with general marginals and general cross- and auto-correlation structures. The DFMM procedure is robust to non-normal marginals and can handle general cross- and auto-correlation structures. In addition, its control limit is calculated analytically without the need for heavy modeling or trial-and-error simulation. These are all desirable properties from a practical point of view. We demonstrate the advantage of the DFMM procedure as a distribution-free and robust monitoring scheme using numerical experiments based on simulated processes. Furthermore, the application of the DFMM procedure to a wafer etching process shows our procedure's industrial relevance and practical values.

Besides applications to the manufacturing industry, the proposed procedure can

also be implemented in a wide range of monitoring problems. The data streams generated by various complex systems form multivariate time-series processes. Monitoring these processes may be needed for various reasons such as service quality monitoring, environmental surveillance, health disease monitoring, etc. The DFMM procedure can serve as an easy-to-implement monitoring tool that is distribution-free and robust.

CHAPTER III

DISTRIBUTION-FREE IMAGE MONITORING WHEN THE TARGET PATTERN IS A RANK-ONE MATRIX

Many modern manufacturing processes require image monitoring, where the measurements are two-dimensional multivariate quality attributes. For example in a battery coating process, a sensor is moving back and forth measuring thickness of an electrode slurry layer coated on a foil while the foil is transferring along the direction perpendicular to the movement of the sensor. During one cycle of movement, the sensor continuously collects around 300 spatially correlated measurements. In ideal situation, the electrode slurry is expected to be coated uniformly within designated area on the foils. In other words, the image of the coated foil, which composes of multivariate product attributes, is expected to follow an ideal target pattern defined by the shape of coating area and coating thickness. Note that the target pattern in the battery coating process can be expressed as a matrix with rank one. Similarly, the quality control of liquid crystal display (LCD) screens requires image monitoring to guarantee uniformity of the flat screens and thus the target pattern is again a rank-one matrix. In this chapter, we aim to develop a monitoring procedure for images whose target pattern is a matrix with rank one.

3.1 Problem and Background

In this section, we define the detection problem, introduce a motivating example from the industry and review a relevant distribution-free monitoring procedure for multivariate time-series data.

3.1.1 Problem Definition

Denote a discrete-time image process as $\{\mathbf{Y}_n : n = 1, 2, \dots\}$, where \mathbf{Y}_n is a $w \times p$ matrix. In the example of monitoring a battery coating process, each entry of \mathbf{Y}_n represents the thickness of slurry loading at a point on a foil. Similarly, in the example of monitoring LCD screens, each entry of \mathbf{Y}_n is an intensity at a pixel. Generally, \mathbf{Y}_n is assumed to have the following form:

$$\mathbf{Y}_n = \mathbf{m} + \mathbf{E}_n,$$

where \mathbf{m} is a deterministic matrix, \mathbf{E}_n is a random noise matrix containing entries with mean 0 and a general marginal distribution. Note that correlation could exist between entries of \mathbf{E}_n or between \mathbf{E}_n and \mathbf{E}_{n+h} for $h = 1, 2, \dots$. Let \mathbf{m}_0 be the in-control mean matrix of \mathbf{Y}_n . Also, let $\mathbf{m}_1 = \mathbf{m}_0 + \mathbf{\Delta}$ be the out-of-control mean matrix where $\mathbf{\Delta}$ is a $w \times p$ non-zero matrix. In the example of battery coating or LCD screen monitoring, a target mean matrix m_0 is indeed rank-one because all rows of \mathbf{m}_0 are identical.

The detection problem can be formulated as the following hypothesis test:

$$\begin{aligned} H_0 : \mathbf{Y}_n &= \mathbf{m}_0 + \mathbf{E}_n, \quad n = 1, 2, \dots \\ H_1 : \mathbf{Y}_n &= \begin{cases} \mathbf{m}_0 + \mathbf{E}_n, & n = 1, 2, \dots, \tau, \\ \mathbf{m}_1 + \mathbf{E}_n, & n = \tau + 1, \tau + 2, \dots, \end{cases} \end{aligned} \quad (7)$$

where τ is the change point for the mean value of \mathbf{Y}_n . In practice, \mathbf{m}_0 is known or can be estimated using existing in-control data, but \mathbf{m}_1 is unknown.

Let \mathbf{a}^\top represent the transpose of a vector \mathbf{a} . Then we have the following assumption on \mathbf{m}_0 :

Assumption 2. *The in-control mean matrix \mathbf{m}_0 is a $w \times p$ matrix of rank one. The SVD of \mathbf{m}_0 is*

$$\mathbf{m}_0 = \lambda_0 \mathbf{u}_0 \mathbf{v}_0^\top,$$

where λ_0 is the only non-zero singular value of \mathbf{m}_0 , \mathbf{u}_0 and \mathbf{v}_0 are the corresponding left and right singular vectors, respectively.

3.1.2 Motivating Problem

Our study is motivated by a battery coating process from the manufacturing industry. As described at the beginning of this chapter, the electrode slurry is sprayed on a long foil, and the foil moves forward to the perpendicular direction to the movement of a sensor while the sensor collects the measurements at each trip from one side to the other side. In this dataset, each trip of the sensor collects 298 measurements thus a 298-variate observation vector \mathbf{x}_t is $\mathbf{x}_t = (x_{t,1}, \dots, x_{t,298})$ for $t = 1, 2, \dots$. The observation is collected once every seven seconds on average, and the dataset contains observations collected in three days. Figure 9 shows the heatmap of the thickness measurements of the electrode slurry on two different portions of the foil. Some characteristics of the data are demonstrated in Figure 10. The first figure shows auto-correlation of the first measurements in the observation vectors, the second figure shows cross-correlation among a few selected measurements, and the last figure shows non-normality of the first measurement in the observation vectors. According to these properties, a distribution-free monitoring procedure that can handle both cross- and auto-correlation is desirable in this application. Since only one sample path is available and monitoring needs to be started as soon as any adjustment to the equipment is done, it would be also convenient to have procedures where the control limits are determined analytically without any model-fitting or trial-and-error simulation.

In this paper, we form a matrix that represents an image of the electrode slurry thickness over a portion of the foil by using a $p \times 1$ observation vector \mathbf{x}_t as a row in the matrix. More specifically, let integers $w \geq 1$ and $1 \leq s \leq w$ represent the window

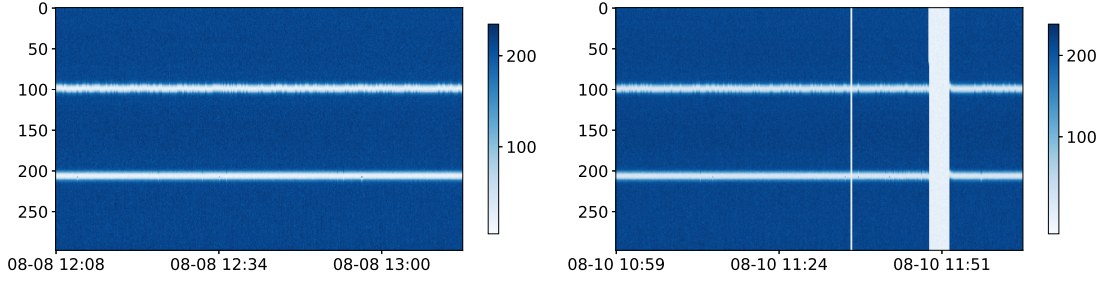


Figure 9: Heatmaps of thickness measurements of the electrode slurry on two different portions of a foil

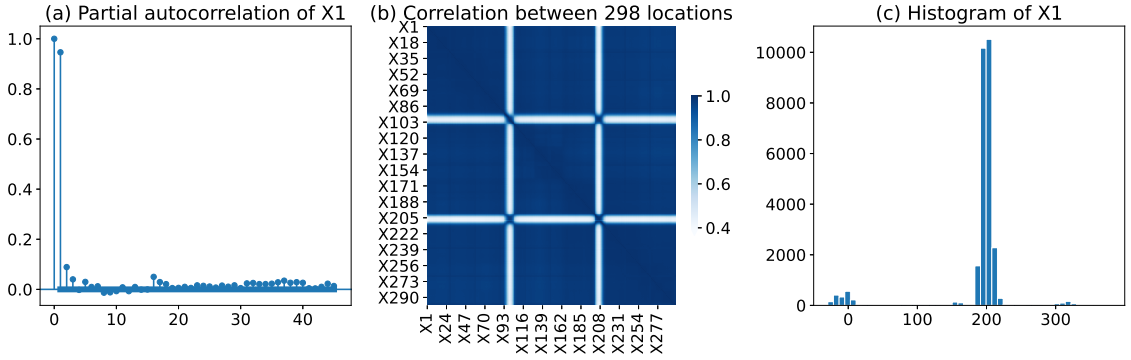


Figure 10: Characteristics of battery coating data

size and the sliding window size. Then an image matrix we get is

$$\mathbf{Y}_n = \begin{bmatrix} \mathbf{x}_{(n-1)s+1}^\top \\ \mathbf{x}_{(n-1)s+2}^\top \\ \vdots \\ \mathbf{x}_{(n-1)s+w}^\top \end{bmatrix}$$

for $n = 1, 2, \dots$ and we monitor the process $\{\mathbf{Y}_n : n = 1, 2, \dots\}$.

3.2 *Distribution-free Image Monitoring Procedure*

In this section, we formulate two new statistics for monitoring \mathbf{Y}_n , which represent images, and propose our image monitoring procedure. Then we provide a theoretical basis for the effectiveness of the monitoring procedure.

3.2.1 Monitoring Statistics

We propose two monitoring statistics utilizing the SVD technique. By Assumption 2, the SVD representation of \mathbf{m}_0 is $\mathbf{m}_0 = \lambda_0 \mathbf{u}_0 \mathbf{v}_0^\top$. Then, the first statistic λ_n^P is constructed by projecting \mathbf{Y}_n along the direction of \mathbf{u}_0 and \mathbf{v}_0 and it is defined as follows:

$$\lambda_n^P = \mathbf{u}_0^\top \mathbf{Y}_n \mathbf{v}_0 \quad \text{for } n = 1, 2, \dots$$

To construct the second statistic λ_n^R , we subtract the in-control mean matrix \mathbf{m}_0 from \mathbf{Y}_n to get a residual matrix:

$$\mathbf{R}_n = \mathbf{Y}_n - \mathbf{m}_0 \quad \text{for } n = 1, 2, \dots$$

and calculate the largest singular value of \mathbf{R}_n , denoted by λ_n^R . When the process is in-control, $\mathbf{R}_n = \mathbf{E}_n$ and the entries of \mathbf{R}_n are random variables with mean 0.

Define

$$\mathbf{z}_n = \begin{bmatrix} \lambda_n^P \\ \lambda_n^R \end{bmatrix}.$$

and consider $\{\mathbf{z}_n : n = 1, 2, \dots\}$, which becomes a multivariate time-series with the two statistics. Then we monitor $\{\mathbf{z}_n : n = 1, 2, \dots\}$ using the DFMM procedure proposed in Chapter 2. Steps for estimating parameters are given in Figure 11 and the full description of the monitoring procedure is give in Figure 12.

Setup Phase for Distribution-free Image Monitoring Procedure

Input: in-control process $\{\mathbf{Y}_n^0 : n = 1, 2, \dots, N\}$, target mean \mathbf{m}_0 if available, target ARL_0 , constant k between 0.01 and 0.1.

1. If \mathbf{m}_0 is not available, estimate it as $\mathbf{m}_0 \leftarrow \frac{1}{N} \sum_{n=1}^N \mathbf{Y}_n^0$. Perform SVD on \mathbf{m}_0 :

$$\mathbf{m}_0 = \lambda_0 \mathbf{u}_0 \mathbf{v}_0^\top.$$

2. For $n = 1, 2, \dots, N$, compute:

$$\lambda_n^P \leftarrow \mathbf{u}_0^\top \mathbf{Y}_n^0 \mathbf{v}_0 \quad \text{and} \quad \mathbf{R}_n^0 \leftarrow \mathbf{Y}_n^0 - \mathbf{m}_0,$$

and perform SVD on \mathbf{R}_n^0 to obtain the largest singular value λ_n^R , and define:

$$\mathbf{z}_n = \begin{bmatrix} \lambda_n^P \\ \lambda_n^R \end{bmatrix}.$$

3. Compute in-control sample mean vector and covariance matrix of \mathbf{z}_n^0 :

$$\bar{\mathbf{z}} \leftarrow \frac{1}{N} \sum_{n=1}^N \mathbf{z}_n^0 \quad \text{and} \quad \mathbf{S}_z \leftarrow \frac{1}{N-1} \sum_{n=1}^N (\mathbf{z}_n^0 - \bar{\mathbf{z}})(\mathbf{z}_n^0 - \bar{\mathbf{z}})^\top.$$

4. For $n = 1, 2, \dots, N$, compute:

$$T_n^2 \leftarrow (\mathbf{z}_n^0 - \bar{\mathbf{z}})^\top \mathbf{S}_z^{-1} (\mathbf{z}_n^0 - \bar{\mathbf{z}}).$$

5. Compute sample mean and standard deviation of in-control T_n^2 :

$$\bar{T}^2 \leftarrow \frac{1}{N} \sum_{n=1}^N T_n^2 \quad \text{and} \quad S_{T^2} \leftarrow \sqrt{\frac{1}{N-1} \sum_{n=1}^N (T_n^2 - \bar{T}^2)^2},$$

and estimate variance parameter Ω^2 of in-control T_n^2 using the CvM estimator formulated by [58].

6. Set $K \leftarrow k S_{T^2}$. Calculate control limit H by solving the following equation:

$$\frac{\Omega^2}{2K^2} \left\{ \exp \left[\frac{2K(H + 1.166\Omega)}{\Omega^2} \right] - 1 - \frac{2K(H + 1.166\Omega)}{\Omega^2} \right\} = \text{ARL}_0. \quad (8)$$

Figure 11: Setup phase for the Distribution-free Image Monitoring Procedure

Distribution-free Image Monitoring Procedure

Input: new data \mathbf{Y}_n to be monitored, values of \mathbf{u}_0 , \mathbf{v}_0 , \mathbf{m}_0 , $\bar{\mathbf{z}}$, \mathbf{S}_z , \bar{T}^2 , K and H from the setup phase.

1. Compute:

$$\lambda_n^P \leftarrow \mathbf{u}_0^\top \mathbf{Y}_n \mathbf{v}_0 \quad \text{and} \quad \mathbf{R}_n = \mathbf{Y}_n - \mathbf{m}_0,$$

and perform SVD on \mathbf{R}_n to obtain the largest singular value λ_n^R . Then define:

$$\mathbf{z}_n \leftarrow \begin{bmatrix} \lambda_n^P \\ \lambda_n^R \end{bmatrix}.$$

2. Compute T^2 statistics:

$$T_n^2 \leftarrow (\mathbf{z}_n - \bar{\mathbf{z}})^\top \mathbf{S}_z^{-1} (\mathbf{z}_n - \bar{\mathbf{z}}).$$

3. Compute CUSUM statistic:

$$S_n^+ = \begin{cases} 0, & \text{if } n = 0, \\ \max\{0, S_{n-1}^+ + (T_n^2 - \bar{T}^2 - K)\}, & \text{if } n \geq 1. \end{cases}$$

4. Raise an out-of-control alarm after the n th observation if $S_n^+ \geq H$.

Figure 12: Algorithmic statement of the Distribution-free Image Monitoring Procedure

3.2.2 Theoretical Analysis

In this subsection, we analyze the in-control and out-of-control mean values of λ_n^P and λ_n^R , and investigate if any shift in mean matrix will be reflected well in the statistics.

Mean of λ_n^P

The in-control and out-of-control mean values of λ_n^P are

$$\mathbb{E}_0[\lambda_n^P] = \mathbb{E}_0[\mathbf{u}_0^\top \mathbf{Y}_n \mathbf{v}_0] = \mathbf{u}_0^\top \mathbb{E}_0[\mathbf{Y}_n] \mathbf{v}_0 = \mathbf{u}_0^\top \mathbf{m}_0 \mathbf{v}_0 = \lambda_0,$$

and

$$\mathbb{E}_1[\lambda_n^P] = \mathbb{E}_1[\mathbf{u}_0^\top \mathbf{Y}_n \mathbf{v}_0] = \mathbf{u}_0^\top \mathbb{E}_1[\mathbf{Y}_n] \mathbf{v}_0 = \mathbf{u}_0^\top \mathbf{m}_1 \mathbf{v}_0 = \lambda_0 + \mathbf{u}_0^\top \mathbf{\Delta} \mathbf{v}_0,$$

where $E_0[\cdot]$ and $E_1[\cdot]$ represent expectation under in-control and out-of-control conditions, respectively. In general, the difference in mean value of λ_n^P can be expressed as

$$E_1[\lambda_n^P] - E_0[\lambda_n^P] = \mathbf{u}_0^\top \Delta \mathbf{v}_0. \quad (9)$$

Mean of λ_n^R

Analyzing the mean value of λ_n^R is more complicated because each entry in \mathbf{R}_n is a random variable with mean 0 and λ_n^R is the largest singular value of a random matrix. We analyze its behavior for a special case where all entries of \mathbf{R}_n are independent and identically distributed (i.i.d.). The following two lemmas analyze the largest singular value of random matrices whose entries are independent.

Lemma 3.2.1 ([62]). *Suppose \mathbf{C} is a $w \times p$ matrix whose entries are i.i.d. random variables having mean zero, variance σ^2 , a finite fourth moment, and $w/p \rightarrow c$ as $p \rightarrow \infty$. Then the largest singular value of $\frac{1}{\sqrt{p}}\mathbf{C}$ converges a.s. to $\sigma(1 + \sqrt{c})$ as $p \rightarrow \infty$.*

Lemma 3.2.2 ([63]). *Suppose \mathbf{D} is a $w \times p$ random matrix with the following form:*

$$\mathbf{D} = \mathbf{B} + \mathbf{C},$$

where \mathbf{B} is a deterministic matrix with the largest singular value denoted as η and \mathbf{C} is a random matrix satisfying the conditions in Lemma 3.2.1. Represent the SVD of \mathbf{B} as $\mathbf{B} = \mathbf{U}\mathbf{S}\mathbf{V}^\top = \mathbf{U}\mathbf{G}^\top$, where $\mathbf{G} = \mathbf{V}\mathbf{S}$. Define

$$\tilde{\mathbf{\Gamma}} = E[\mathbf{C}^\top \mathbf{C}]/w, \quad \tilde{\mathbf{\Sigma}} = \mathbf{G}^\top \tilde{\mathbf{\Gamma}} \mathbf{G}$$

and

$$\hat{\mathbf{\Gamma}} = E[\mathbf{C}\mathbf{C}^\top]/p, \quad \hat{\mathbf{\Sigma}} = \mathbf{U}^\top \hat{\mathbf{\Gamma}} \mathbf{U}.$$

Let

$$\psi = \frac{1}{2} \left[\frac{\sqrt{c}}{\gamma^3 w p} \tilde{\mathbf{\Sigma}}_{11} + \frac{1}{\sqrt{c} \gamma} \hat{\mathbf{\Sigma}}_{11} \right],$$

where $c = \lim_{p \rightarrow \infty} \frac{w}{p}$ and $\gamma = \lim_{p \rightarrow \infty} \frac{\eta}{\sqrt{wp}}$. Then the largest singular value of \mathbf{D} , denoted as λ , can be represented as a sum of four terms:

$$\lambda = \eta + \psi + Z + \varepsilon,$$

where Z is a random centered fluctuation term with mean 0, ψ is a deterministic term depending on the variances of the entries of \mathbf{C} , and $\varepsilon \rightarrow 0$ in probability as $p \rightarrow \infty$.

Lemma 3.2.1 implies that when p is large,

$$\mathbb{E}_0[\lambda_n^R] \approx \sigma(\sqrt{w} + \sqrt{p}).$$

When the process $\{\mathbf{Y}_n : n = 1, 2, \dots\}$ becomes out-of-control and the mean matrix shifts to \mathbf{m}_1 , a residual matrix can be expressed as

$$\mathbf{R}_n = \mathbf{m}_1 - \mathbf{m}_0 + \mathbf{E}_n = \mathbf{\Delta} + \mathbf{E}_n,$$

which can be viewed as a random perturbation of a low rank matrix $\mathbf{\Delta}$. Denote the largest singular value of $\mathbf{\Delta}$ as λ_d . When \mathbf{E}_n has i.i.d. entries with variance σ^2 , $\mathbb{E}[\mathbf{E}_n^\top \mathbf{E}_n]/w$ and $\mathbb{E}[\mathbf{E}_n \mathbf{E}_n^\top]/p$ become diagonal matrices with all diagonal elements being σ^2 . It follows from Lemma 3.2.2 that when p is large, we have

$$\mathbb{E}_1[\lambda_n^R] \approx \lambda_d + \frac{\sigma^2(w+p)}{2\lambda_d}.$$

Then we can approximate the difference of mean values of λ_n^R as:

$$\begin{aligned} \mathbb{E}_1[\lambda_n^R] - \mathbb{E}_0[\lambda_n^R] &\approx \lambda_d + \frac{\sigma^2(w+p)}{2\lambda_d} - \sigma(\sqrt{w} + \sqrt{p}) \\ &\geq \sigma\sqrt{2(w+p)} - \sigma(\sqrt{w} + \sqrt{p}) \\ &= \sigma(\sqrt{2(w+p)} - \sqrt{w} - \sqrt{p}), \end{aligned} \tag{10}$$

where the inequality is true due to $a + b \geq 2\sqrt{ab}$ for any non-negative a and b , and the equality holds when $a = b$, i.e., $\lambda_d = \sigma\sqrt{\frac{w+p}{2}}$. The lower bound in (10) is always non-negative since

$$\sqrt{2(w+p)} \geq \sqrt{w} + \sqrt{p} \iff 2(w+p) \geq w + p + 2\sqrt{wp}$$

$$\begin{aligned} &\iff w + p - 2\sqrt{wp} \geq 0 \\ &\iff (\sqrt{w} - \sqrt{p})^2 \geq 0, \end{aligned}$$

and the equality holds when $w = p$.

The above results hold for a random matrix with i.i.d. entries. We test if the results hold for a random matrix with dependent entries in Section 3.3.2.

Theoretical Comparison

In summary, we have

$$\mathbb{E}_1[\lambda_n^P] - \mathbb{E}_0[\lambda_n^P] = \mathbf{u}_0^\top \mathbf{\Delta} \mathbf{v}_0$$

and

$$\mathbb{E}_1[\lambda_n^R] - \mathbb{E}_0[\lambda_n^R] \gtrsim \sigma(\sqrt{2(w+p)} - \sqrt{w} - \sqrt{p})$$

when $\lambda_d \neq \sigma\sqrt{\frac{w+p}{2}}$ and $w \neq p$ for \mathbf{R}_n . We consider two special cases to show advantages and disadvantages of each statistic:

Case 1: Consider a shift matrix $\mathbf{\Delta}$ with the following SVD:

$$\mathbf{\Delta} = \lambda_d \mathbf{u}_d \mathbf{v}_d^\top,$$

where $\mathbf{u}_0^\top \mathbf{u}_d = 0$. Further, suppose the conditions for equality in (10) do not hold. Then it follows from (9) that the change in the mean value of λ_n^P is:

$$\mathbf{u}_0^\top \mathbf{\Delta} \mathbf{v}_0 = \lambda_d \mathbf{u}_0^\top \mathbf{u}_d \mathbf{v}_d^\top \mathbf{v}_0 = 0.$$

This result implies that λ_n^P cannot detect the shift $\mathbf{\Delta}$ in this case. The same conclusion holds for any shift $\mathbf{\Delta}$ such that $\mathbf{u}_0^\top \mathbf{v}_d = 0$. On the other hand, the shift $\mathbf{\Delta}$ is reflected in the mean of λ_n^R as shown in (10) given that the equality does not hold and thus λ_n^R is a better statistic.

Case 2: Now consider a case where $\mathbf{u}_d = \mathbf{u}_0$ and $\mathbf{v}_d = \mathbf{v}_0$. Then, the mean difference of λ_n^P can be further simplified as λ_d given that \mathbf{u}_0 and \mathbf{v}_0 are unit vectors.

Equation (10) implies that the mean difference of λ_n^R is smaller than λ_d when

$$\frac{\sigma^2(w+p)}{2\lambda_d} - \sigma(\sqrt{w} + \sqrt{p}) < 0 \iff \lambda_d > \frac{\sigma(w+p)}{2(\sqrt{w} + \sqrt{p})}.$$

In this case, λ_n^P is a better statistic because it has a larger shift in the mean value and therefore it is expected to be more responsive to the shift Δ .

As each statistic works better under different shifts, we combine them into a two-dimensional vector \mathbf{z}_n and monitor the vectors, in which case we can determine a control limit for a given target ARL_0 . Note that one may consider monitoring λ_n^P and λ_n^R separately using two CUSUM charts. However, the two CUSUM charts become correlated, which makes it difficult to determine control limits that ensure the overall ARL_0 equal to a target.

3.2.3 Computation Analysis

Most applications of image monitoring, such as the battery coating process discussed in Section 3.1.2, have high-speed data acquisition processes. In order to monitor the processes in an on-line setting, an applicable monitoring procedure needs to have low computational complexity during the monitoring phase.

In the proposed image monitoring procedure, most computational efforts are spent on acquiring λ_n^P and λ_n^R . For λ_n^P , only matrix multiplication is involved in the monitoring phase as the singular vectors of the in-control mean matrix are obtained in the setup phase. For λ_n^R , a matrix subtraction and SVD of the residual matrix are performed for each image matrix. After λ_n^P and λ_n^R are computed, the T^2 statistic are calculated using the in-control mean vector and the inverse covariance matrix which are estimated during the setup phase. All parameters used in the CUSUM chart, including the control limit, are also pre-determined in the setup phase. Therefore, the SVD of a residual matrix is the dominating operation in the monitoring phase. Given that many efficient implementations of SVD have been developed, the proposed

procedure will be suitable for image monitoring applications.

3.3 Numerical Experiments

In this section, we compare the proposed procedure against two baseline procedures using numerical experiments. We introduce testing processes, review the baseline procedures in details, and discuss performances of the three procedures in monitoring images under different cross- and auto-correlation settings.

3.3.1 Simulated Data

We consider two different ways to simulate image processes. Type 1 processes generates vector time-series and use them to form image matrices while Type 2 processes directly generates image matrices.

Type 1: Forming an image matrix from vectors

Inspired by data collection in the battery coating process, we first consider generating a vector auto-regressive (VAR) time-series, from which an image process can be formed using a sliding window over the vector time-series. Specifically, a p -variate vector time-series $\{\mathbf{x}_t\}$ is generated as:

$$\begin{aligned}\mathbf{x}_t &= \boldsymbol{\mu} + \mathbf{e}_t, \\ \mathbf{e}_t &= \boldsymbol{\Phi}\mathbf{e}_{t-1} + \boldsymbol{\epsilon}_t, \quad t = 1, 2, \dots, \\ \mathbf{e}_0 &= \boldsymbol{\epsilon}_0,\end{aligned}\tag{11}$$

where $\boldsymbol{\mu}$ is the marginal mean vector of \mathbf{x}_t , \mathbf{e}_t are the auto-correlated error terms, $\boldsymbol{\Phi}$ is the auto-correlation coefficient matrix, and $\boldsymbol{\epsilon}_t$ are independent and identically distributed (i.i.d.) normal random vector with mean 0 and covariance matrix $\boldsymbol{\Sigma}_\epsilon$. The in-control mean vector is denoted by $\boldsymbol{\mu}_0$ and the out-of-control mean vector is $\boldsymbol{\mu}_1$. Then an image process $\{\mathbf{Y}_n\}$ is defined by a window size w and a sliding window size

s of process $\{\mathbf{x}_t\}$:

$$\mathbf{Y}_n = \begin{bmatrix} \mathbf{x}_{(n-1)s+1}^\top \\ \mathbf{x}_{(n-1)s+2}^\top \\ \vdots \\ \mathbf{x}_{(n-1)s+w}^\top \end{bmatrix}.$$

We set $s = 1$ and thus

$$\mathbf{Y}_n = \begin{bmatrix} \mathbf{x}_n^\top \\ \mathbf{x}_{n+1}^\top \\ \vdots \\ \mathbf{x}_{n+w-1}^\top \end{bmatrix}$$

The in-control mean matrix of \mathbf{Y}_n is a rank-1 matrix:

$$\mathbf{m}_0 = \mathbb{E}_0[\mathbf{Y}_n] = \begin{bmatrix} \boldsymbol{\mu}_0 & \boldsymbol{\mu}_0 & \cdots & \boldsymbol{\mu}_0 \end{bmatrix}^\top.$$

We set $\boldsymbol{\mu}_0$ to have value five for every entry which represents a target of the uniform height as in the battery coating application. We set $p = 200$ and $w = 5$. The auto-correlation coefficient matrix, $\boldsymbol{\Phi}$ is chosen to be a diagonal matrix where $[\boldsymbol{\Phi}]_{ij} = \phi$ if $i = j$ and 0 if $i \neq j$. The diagonal element ϕ takes 0.3 for low auto-correlation or 0.7 for high auto-correlation. Two different structures of cross-correlation are tested by varying the assignments of $\boldsymbol{\Sigma}_\epsilon$ among the following two models with $\rho = 0.3$:

- (i) Tri-diagonal model: $[\boldsymbol{\Sigma}_\epsilon]_{ij} = 1$ if $i = j$; ρ if $|i - j| = 1$; 0 otherwise.
- (ii) Exponential model: $[\boldsymbol{\Sigma}_\epsilon]_{ij} = \rho^{|i-j|}$.

To simulate out-of-control scenarios, a 200-variate vector $\boldsymbol{\delta} = [\delta_j]$ for $j = 1, 2, \dots, 200$ is added to $\boldsymbol{\mu}_0$ and the out-of-control mean vector of \mathbf{x}_t is $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_0 + \boldsymbol{\delta}$. Three shapes of $\boldsymbol{\delta}$ are considered in our experiments:

(i) Sparse: for $1 \leq j \leq 200$,

$$\delta_j = \begin{cases} d, & \text{if } 18 \leq j \leq 22, \\ 0, & \text{otherwise.} \end{cases}$$

(ii) Step: for $1 \leq j \leq 200$,

$$\delta_j = \begin{cases} \frac{d}{3}, & \text{if } 51 \leq j \leq 100, \\ \frac{2d}{3}, & \text{if } 101 \leq j \leq 150, \\ d, & \text{if } 151 \leq j \leq 200, \\ 0, & \text{otherwise.} \end{cases}$$

(iii) Zigzag: for $0 \leq k \leq 4, 1 \leq \ell \leq 20$,

$$\delta_j = \begin{cases} d(1 - \frac{j-40k}{10}), & \text{if } j = 40k + \ell, \\ d(-1 + \frac{j-40k-20}{10}), & \text{if } j = 40k + 20 + \ell. \end{cases}$$

The value of d is adjusted to achieve a different magnitude of a shift defined by $\|\boldsymbol{\delta}\| = \sqrt{\sum_j \delta_j^2}$ for Type 1 simulated processes. From the perspective of image process $\{\mathbf{Y}_n, n = 1, 2, \dots\}$, the in-control and out-of-control mean matrices are as follows:

$$\mathbf{m}_0 = \begin{bmatrix} \boldsymbol{\mu}_0^\top \\ \boldsymbol{\mu}_0^\top \\ \vdots \\ \boldsymbol{\mu}_0^\top \end{bmatrix} \quad \text{and} \quad \mathbf{m}_1 = \begin{bmatrix} \boldsymbol{\mu}_1^\top \\ \boldsymbol{\mu}_1^\top \\ \vdots \\ \boldsymbol{\mu}_1^\top \end{bmatrix} = \mathbf{m}_0 + \boldsymbol{\Delta} \quad \text{where} \quad \boldsymbol{\Delta} = \begin{bmatrix} \boldsymbol{\delta}^\top \\ \boldsymbol{\delta}^\top \\ \vdots \\ \boldsymbol{\delta}^\top \end{bmatrix}.$$

Figure 13 visualizes shift matrix $\boldsymbol{\Delta}$ as 2D images for the sparse, step, and zigzag shifts when $d = 1$.

Type 2: Generating image matrices

We also consider directly generating a series of matrices $\{\mathbf{Y}_n : n = 1, 2, \dots\}$ as:

$$\mathbf{Y}_n = \mathbf{m} + \mathbf{E}_n, \tag{12}$$

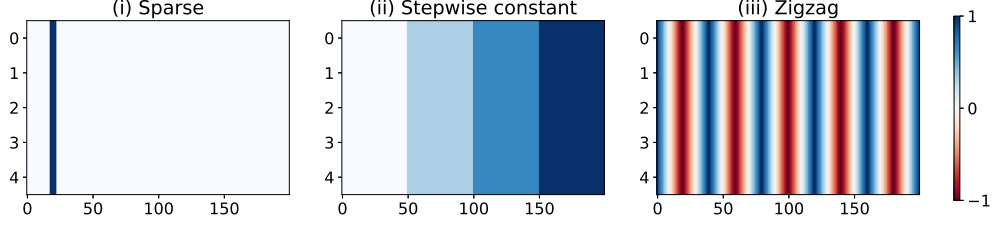


Figure 13: Three shifts of Δ for Type 1 simulated processes when $d = 1$

where \mathbf{Y}_n are $w \times p$ matrices, \mathbf{m} is the mean matrix of \mathbf{Y}_n , and \mathbf{E}_n are matrices of random noises. In our experiments, we set $w = 100$ and $p = 200$. Noise matrices \mathbf{E}_n are generated as i.i.d. random matrices with covariance represented by $\Sigma_c \otimes \Sigma_r$, where Σ_c captures the covariance among columns of \mathbf{E}_n , Σ_r captures that among rows, and \otimes denotes the Kronecker product. Two distributions are considered as the noise distribution:

- (i) Normal noise: each \mathbf{E}_n is generated following a matrix normal distribution with covariance $\Sigma_c \otimes \Sigma_r$. Each entry $[\mathbf{E}_n]_{ij}$ has mean 0 and variance 1.
- (ii) Non-normal noise: a normally distributed matrix \mathbf{E}'_n is generated following the previous normal-noise case. Then each entry $[\mathbf{E}'_n]_{ij}$ is transformed to $[\mathbf{E}_n]_{ij}$ as:

$$[\mathbf{E}_n]_{ij} = -\log(1 - \Psi([\mathbf{E}'_n]_{ij})),$$

where $\Psi(\cdot)$ represents the cumulative distribution of the standard normal random variable. Consequently, each entry $[\mathbf{E}_n]_{ij}$ follows an exponential distribution with mean 1, and entries in $[\mathbf{E}_n]_{ij}$ are correlated.

Note that we do not consider auto-correlation in Type 2 simulated processes, since it is covered by Type 1 processes. Instead, Type 2 processes test the capability of handling non-normal data. Similarly to the setting of Σ_ϵ for Type 1 processes, Σ_c and Σ_r either follow the tri-diagonal model or exponential model with $\rho = 0.3$. In

other words, for each noise distribution, the following two cases of Σ_c are considered:

$$\begin{bmatrix} 1 & 0.3 & 0 & \cdots & 0 \\ 0.3 & 1 & 0.3 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0.3 & 1 & 0.3 \\ 0 & \cdots & 0 & 0.3 & 1 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 0.3 & 0.3^2 & \cdots & 0.3^{p-1} \\ 0.3 & 1 & 0.3 & \cdots & 0.3^{p-2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0.3^{p-2} & \cdots & 0.3 & 1 & 0.3 \\ 0.3^{p-1} & \cdots & 0.3^2 & 0.3 & 1 \end{bmatrix},$$

and Σ_r is set to have the same form as Σ_c but with different dimensions. The in-control mean matrix \mathbf{m}_0 is set to five for every entry.

To simulate different 2D out-of-control scenarios, shifting matrices $\Delta = [\Delta_{ij}]$ are added to the in-control mean matrix \mathbf{m}_0 : $\mathbf{m}_1 = \mathbf{m}_0 + \Delta$. Specifically, we consider the following five shifting patterns:

(i) Sparse:

$$\Delta_{ij} = \begin{cases} d, & \text{if } 8 \leq i \leq 13 \text{ and } 18 \leq j \leq 23, \\ 0, & \text{otherwise.} \end{cases}$$

(ii) Chessboard: for $0 \leq k \leq 4, 1 \leq \ell \leq 10, 0 \leq r \leq 9, 1 \leq s \leq 5$,

$$\Delta_{ij} = \begin{cases} d, & \text{if } j = 40k + 10 + \ell \text{ and } i = 10r + s, \\ d, & \text{if } j = 40k + 20 + \ell \text{ and } i = 10r + 5 + s, \\ -d, & \text{if } j = 40k + 30 + \ell \text{ and } i = 10r + s, \\ -d, & \text{if } j = 40k + \ell \text{ and } i = 10r + 5 + s, \\ 0, & \text{otherwise.} \end{cases}$$

(iii) Ring-wise constant: for $k \geq 0, 0 \leq \ell \leq 3$,

$$\Delta_{ij} = \begin{cases} d, & \text{if } \lfloor \sqrt{(i-50)^2 + (j-100)^2} \rfloor = 12k + \ell, \\ -d, & \text{if } \lfloor \sqrt{(i-50)^2 + (j-100)^2} \rfloor = 12k + 8 + \ell, \\ 0, & \text{otherwise.} \end{cases}$$

(iv) Row-wise sine: for $1 \leq j \leq 200$,

$$\Delta_{ij} = \begin{cases} d \sin \frac{j\pi}{10}, & \text{if } 1 \leq i \leq 30, \\ d \sin \frac{j\pi}{5}, & \text{if } 31 \leq i \leq 60, \\ d \sin \frac{3j\pi}{10}, & \text{if } 61 \leq i \leq 100. \end{cases}$$

(v) Column-wise sine: for $1 \leq i \leq 100$,

$$\Delta_{ij} = \begin{cases} d \sin \frac{i\pi}{5}, & \text{if } 1 \leq j \leq 60, \\ d \sin \frac{2i\pi}{5}, & \text{if } 61 \leq j \leq 120, \\ d \sin \frac{3i\pi}{5}, & \text{if } 121 \leq j \leq 200. \end{cases}$$

The value of d can be varied to achieve a different magnitude of a shift defined by $\|\Delta\| = \sqrt{\sum_{i,j} \Delta_{ij}^2}$ for Type 2 simulated processes. Figure 14 visualizes these five shifting patterns when $d = 1$.

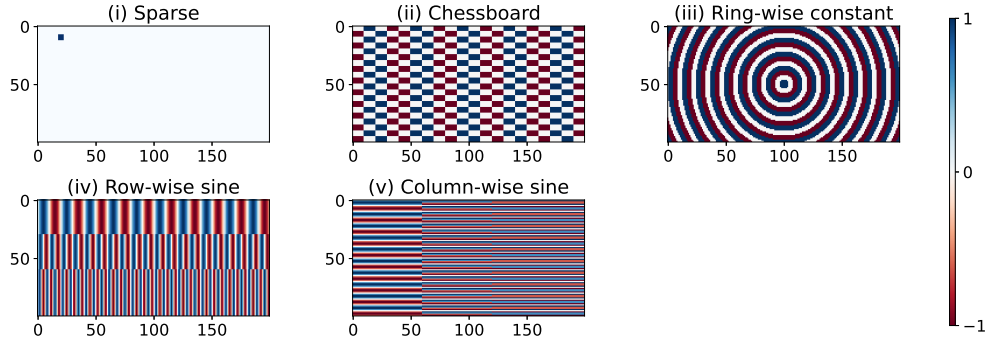


Figure 14: Five shift patterns of Δ for Type 2 simulated processes when $d = 1$

3.3.2 Verifying Theoretical Results for Dependent Entries

In Section 3.2.2, we show that a shift in the mean pattern is properly captured by λ_n^R using Lemmas 3.2.1 and 3.2.2. However, the results are under the assumption that entries in a matrix are i.i.d. In this section, we verify if the results hold for a matrix with dependent entries by comparing the in-control and out-of-control means of λ_n^R based on Lemmas 3.2.1 and 3.2.2 with actual mean values estimated from simulated

data. The comparison is conducted using a Type 1 simulated process with $\phi = 0.3$, the tri-diagonal model with $\rho = 0.3$, and a sparse shift with $\|\boldsymbol{\delta}\| = 5$.

Note that we have $p = 200$ and $w = 5$ with $\sigma^2 = 1$ and we get the theoretical difference between $E_0[\lambda_n^R]$ and $E_1[\lambda_n^R]$ is approximately equal to $\sqrt{2(5 + 200)} - \sqrt{5} - \sqrt{200} = 3.87$ by (10). From the simulated data with dependent entries, we get $E_0[\lambda_n^R] \approx 19.31$ and $E_1[\lambda_n^R] \approx 22.14$ and the difference between the means is 2.83. We also test simulated data with independent entries with matching means and marginal variances with $p = 200$ and $w = 5$. From the simulated data with independent entries, we get $E_0[\lambda_n^R] \approx 15.74$ and $E_1[\lambda_n^R] \approx 18.30$ and the difference between the means is 2.56, which is very similar to the difference we obtained for simulated data with dependent entries.

Although this experiment does not provide any general conclusion, it provides evidence that λ_n^R captures a shift in the mean pattern well even when the assumption of i.i.d. entries is violated.

3.3.3 Baseline Procedures

According to [21], image monitoring is sometimes considered as an extension of profile monitoring. Therefore, in our experiment, we choose one procedure for profile monitoring and another procedure for image monitoring and use them as baseline procedures.

For the first baseline procedure, we consider the multivariate exponentially weighted moving average (MEWMA) chart due to [37], which is based on fitting a regression model to monitor auto-correlated and general linear 1D profiles. For Type 1 simulated processes, each observation \boldsymbol{x}_t can be treated as a 1D profile data, and the MEWMA chart can be directly applied to monitor the series of 1D profiles. For Type 2 simulated processes, each observation \boldsymbol{Y}_n is a matrix thus it needs to be vectorized before the MEWMA chart can be applied. Specifically, a $w \times p$ matrix \boldsymbol{Y}_n is vectorized as

a vector of length wp as follows:

$$\text{vec}(\mathbf{Y}) = (Y_{11}, Y_{12}, \dots, Y_{1p}, Y_{21}, Y_{22}, \dots, Y_{2p}, \dots, Y_{w1}, Y_{w2}, \dots, Y_{wp}).$$

Then these vectors are treated as profiles and the MEWMA chart can be applied to monitor the series of the vectors. The control limit of the MEWMA chart is chosen by trial-and-error simulation to achieve a target ARL_0 . A grid search is applied to tune the smoothing parameter of the MEWMA chart to achieve the best detection performance, and it is set to 0.1.

For the second baseline procedure, we consider an image monitoring procedure due to [45], which is based on the principle of the multivariate generalized likelihood ratio (MGLR) test. This MGLR procedure is motivated by a problem of monitoring products whose target quality is described by a specific pattern. A good example is an uniform pattern in LED screens, which is similar to the monitoring problem of battery coating. For the purpose of dimension reduction, the MGLR procedure first divides the original image into q equal-size rectangle areas, and then focus on monitoring the mean shift of the vector of mean values calculated from the q areas by applying the generalized likelihood ratio test. For Type 1 simulated processes, the MGLR procedure is applied to matrices formed by the vector time-series $\{\mathbf{x}_t\}$ with a window w and a sling window size $s = 1$. For Type 2 simulated processes, the MGLR procedure is directly applied to $\{\mathbf{Y}_n\}$. The control limit of the MGLR procedure is chosen by trial-and-error simulation for a given target ARL_0 .

3.3.4 Experiment Results

In-control Performance

For each setting we tested, an in-control sample path is generated as training data and we use them to estimate the required parameters for the two baseline procedures and the DFIM procedure. The training data for each Type 1 simulated process has

50,000 observations while each Type 2 simulated process has 30,000 observations. The control limit of the DFIM procedure is calculated analytically by equation (8) once all parameters are estimated from the training data. However, the two baseline procedures need trial-and-error simulation to determine their control limits. Thus we generate 1000 in-control independent sample paths and use them to calibrate their control limits for a target $ARL_0 = 1000$. We also use the same 1000 in-control sample paths to get actual ARL_0 for the DFIM procedure with the analytically determined control limits. Table 6 shows the control limits of all procedures and their corresponding estimated ARL_0 's for each setting of simulated processes. Although all three procedures show that they achieve actual ARL_0 close to target 1000, the amount of data and time required to determine control limits is much larger for the two baseline procedures compared to the DFIM procedure.

Out-of-control Performance

For each out-of-control scenario, 1000 independent sample paths are generated and the three monitoring procedures are applied to estimate ARL_1 as a measure of the detection performance. Tables 7, 8 and 9 compare ARL_1 's of the three procedures under various correlation levels for Type 1 and Type 2 simulated processes.

Table 7 shows that the DFIM procedure has the smallest ARL_1 for the sparse and zigzag shifts. It is noteworthy that the two baseline procedures fail to detect

Table 6: Control limit (H) and actual ARL_0 for various settings of simulated processes with target $ARL_0 = 1000$ (standard deviation in parentheses)

Process	Setting	MEWMA		MGLR		DFIM	
		H	ARL_0	H	ARL_0	H	ARL_0
Type 1	$\phi = 0.3$, Tri-diagonal	14.12	999.3 (949.82)	62.63	1004.0 (1005.44)	107.15	1000.7 (838.79)
	$\phi = 0.7$, Tri-diagonal	12.77	1003.4 (1006.92)	80.52	1000.7 (960.77)	125.45	1000.4 (822.89)
	$\phi = 0.3$, Exponential	16.58	999.9 (1018.11)	62.45	1002.6 (982.43)	107.43	1005.5 (846.57)
	$\phi = 0.7$, Exponential	15.1	998.3 (980.56)	81.08	1000.6 (1027.23)	129.54	998.5 (881.62)
Type 2	Normal, Tri-diagonal	23.74	1004.0 (955.21)	79.39	1001.9 (1041.45)	59.25	1004.8 (871.27)
	Normal, Exponential	31.85	999.1 (959.62)	79.55	1003.7 (1050.18)	54.84	1007.6 (897.15)
	Non-normal, Tri-diagonal	21.73	1004.7 (1005.09)	79.88	995.4 (1019.76)	55.85	1000.7 (830.95)
	Non-normal, Exponential	28.40	1002.8 (1001.63)	80.05	996.0 (1013.58)	54.24	998.2 (850.07)

Table 7: ARL_1 for out-of-control Type 1 simulated processes (standard deviation in parentheses)

Setting	Shape of δ	$\ \delta\ $	MEWMA	MGLR	DFIM
$\phi = 0.3$ Tri-diagonal	Sparse	2	291.7 (291.72)	447.9 (688.02)	210.7 (144.20)
		5	47.9 (41.00)	24.6 (20.19)	12.8 (5.47)
	Step	1	48.1 (39.90)	154.3 (143.42)	99.3 (53.03)
		2	11.0 (7.64)	18.4 (13.91)	25.3 (11.51)
	Zigzag	2	960.0 (939.82)	1015.7 (1029.34)	274.3 (176.25)
		5	937.7 (911.54)	1031.1 (1028.94)	14.4 (6.54)
$\phi = 0.7$ Tri-diagonal	Sparse	2	688.9 (697.18)	721.5 (715.70)	679.8 (577.89)
		5	225.5 (212.05)	227.6 (220.13)	108.0 (62.34)
	Step	1	218.2 (205.62)	557.4 (542.06)	330.3 (241.16)
		2	55.1 (48.65)	177.4 (175.92)	103.8 (59.99)
	Zigzag	2	964.7 (950.86)	1031.8 (1049.91)	758.3 (652.91)
		5	989.2 (1011.31)	998.2 (966.88)	134.6 (80.48)
$\phi = 0.3$ Exponential	Sparse	2	330.0 (334.51)	315.6 (320.62)	224.2 (147.39)
		5	58.2 (53.52)	30.5 (26.58)	13.4 (6.06)
	Step	1	56.7 (51.52)	167.4 (164.19)	113.4 (63.63)
		2	12.9 (8.58)	22.3 (17.69)	28.9 (13.41)
	Zigzag	2	1024.7 (1056.13)	971.4 (944.37)	285.1 (192.45)
		5	1029.3 (1049.97)	939.8 (928.26)	14.5 (6.79)
$\phi = 0.7$ Exponential	Sparse	2	732.6 (721.64)	736.6 (740.58)	636.7 (525.47)
		5	266.7 (263.71)	314.9 (447.98)	114.6 (67.55)
	Step	1	271.6 (278.13)	666.2 (763.75)	404.7 (414.44)
		2	68.1 (63.77)	233.4 (325.82)	115.7 (68.84)
	Zigzag	2	1052.6 (1084.03)	848.2 (812.43)	714.9 (585.80)
		5	1013.2 (1081.01)	807.5 (719.61)	138.4 (99.18)

the zigzag shift because their ARL_1 's are close to target ARL_0 1000. When δ is the stepwise shift, the DFIM procedure has ARL_1 larger than the MEWMA procedure but smaller than the MGLR procedure. Tables 8 and 9 demonstrate more scenarios where the DFIM procedure has significantly smaller ARL_1 than the two baseline procedures. Specifically, the DFIM procedure outperforms the two baseline procedures for all cases of out-of-control Type 2 simulated process with normal noises. For Type 2 process with non-normal noises, the DFIM procedure has the smallest ARL_1 except for sparse and ring shifts. For the chessboard and row-wise and column-wise sine shifts, the baseline procedures fail to detect them while the DFIM procedure effectively detects them with small ARL_1 such as 1.8 or 1.9 for some cases.

Overall, the DFIM procedure can be set up quickly, requiring less time and data

Table 8: ARL_1 for various out-of-control Type 2 simulated processes with normal noises (standard deviation in parenthesis)

Setting	Shape of Δ	$\ \Delta\ $	MEWMA	MGLR	DFIM
Normal Tri-diagonal	Sparse	10	191.5 (175.72)	180.5 (173.13)	134.4 (71.8)
		20	47.2 (39.09)	13.1 (7.67)	1.9 (0.77)
	Chessboard	10	1038.8 (996.77)	983.9 (944.25)	271.9 (171.43)
		20	1008.5 (1022.65)	986.5 (949.56)	7.2 (2.66)
	Ring	20	218.9 (213.86)	56.1 (52.89)	35.2 (13.31)
		30	100.1 (90.13)	11.9 (6.62)	5.2 (1.72)
	Sine (row)	20	976.3 (1008.57)	984.5 (963.12)	12.7 (4.45)
		30	990.2 (1037.35)	1039.9 (1089.72)	1.8 (0.61)
	Sine (column)	20	1003.7 (1057.11)	982.6 (1021.21)	36.8 (13.89)
		30	982.4 (1001.09)	998.7 (969.84)	3.0 (0.97)
Normal Exponential	Sparse	10	257.6 (259.43)	270.7 (253.01)	131.4 (73.09)
		20	64.7 (54.17)	21.0 (15.12)	2.2 (0.88)
	Chessboard	10	1031.4 (1001.81)	1091.2 (1055.10)	218.5 (135.70)
		20	958.4 (906.65)	1050.2 (1052.08)	7.0 (2.63)
	Ring	20	284.1 (289.60)	101.8 (101.10)	32.6 (12.67)
		30	139.8 (129.21)	19.2 (14.44)	5.2 (1.79)
	Sine (row)	20	958.6 (934.14)	1043.6 (1006.06)	12.0 (4.26)
		30	962.0 (973.97)	973.6 (950.01)	1.9 (0.63)
	Sine (column)	20	1042.0 (1044.64)	1073.1 (1079.59)	52.6 (22.52)
		30	997.3 (985.75)	1015.8 (1006.50)	4.2 (1.36)

Table 9: ARL_1 for various out-of-control Type 2 simulated processes with non-normal noises (standard deviation in parenthesis)

Setting	Shape of Δ	$\ \Delta\ $	MEWMA	MGLR	DFIM
Non-normal Tri-diagonal	Sparse	10	163.1 (157.93)	174.9 (174.62)	379.3 (268.84)
		20	42.3 (37.43)	11.5 (5.75)	63.8 (25.36)
	Chessboard	10	1027.8 (1080.70)	1026.8 (996.98)	34.0 (12.28)
		20	988.6 (1009.45)	1017.6 (982.24)	3.0 (0.94)
	Ring	20	187.3 (175.85)	55.0 (46.61)	95.9 (44.69)
		30	86.4 (78.36)	11.4 (5.77)	23.0 (7.98)
	Sine (row)	20	931.1 (934.68)	1035.6 (1100.92)	5.6 (1.85)
		30	962.2 (967.95)	1023.7 (1015.07)	1.6 (0.54)
	Sine (column)	20	920.4 (923.48)	1047.7 (1078.07)	5.8 (1.84)
		30	980.2 (980.59)	1022.8 (1022.38)	1.6 (0.54)
Non-normal Exponential	Sparse	10	233.8 (214.66)	245.8 (237.97)	453.2 (367.35)
		20	59.2 (51.47)	17.3 (11.99)	78.0 (35.07)
	Chessboard	10	996.7 (976.34)	964.0 (944.68)	41.3 (15.91)
		20	965.5 (959.08)	1005.1 (1001.87)	3.7 (1.14)
	Ring	20	261.1 (249.10)	87.5 (77.77)	119.6 (60.42)
		30	128.7 (118.86)	15.7 (10.22)	28.8 (10.32)
	Sine (row)	20	1035.4 (1014.62)	943.7 (895.29)	7.1 (2.36)
		30	968.3 (910.26)	1011.3 (1047.13)	1.9 (0.60)
	Sine (column)	20	972.4 (969.22)	964.6 (923.39)	7.3 (2.28)
		30	949.2 (977.24)	999.3 (992.26)	1.9 (0.58)

than the baseline procedures to estimate required parameters and determine control limits for a given target ARL_0 . It also effectively detects various out-of-control shifts with the smallest ARL_1 than the baseline procedures in most cases we tested. The DFIM procedure shows a competitive performance even when it is not the best one and can detect shifts that the baseline procedures fail to detect.

3.4 Battery Coating Process

In this section, we demonstrate the performance of the DFIM procedure in a real application using the battery coating process introduced in Section 3.1.2. We consider a time-series dataset of battery coating thickness measurements collected from a real production line. Since only one sample path is available, trial-and-error simulation is not possible and thus the two baseline models are not implemented.

Batter coating data measuring thicknesses of slurry coating are available from August/8 to August/11. The production line performed a maintenance around August/10 between 12:00 and August/11 0:00. The production manager reviewed other indicators and believes that a shift occurred around August/10 12:00. So we choose battery coating data between August/8 12:00 and August/9 0:00 as the training data and use them to set up the DFIM procedure. Then we apply it to monitor the process afterwards.

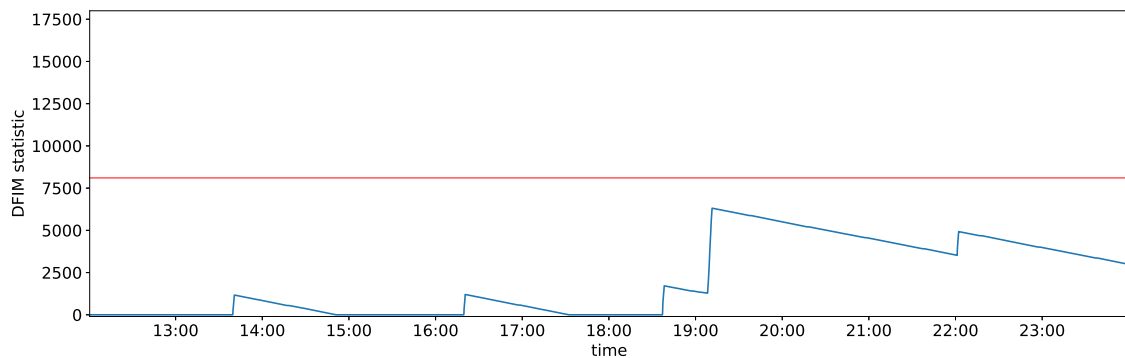


Figure 15: The DFIM procedure on training data from August/8 12:00 to August/9 0:00

After estimating required parameters and calculating the control limit of the DFIM procedure from the training data, the DFIM procedure is applied to the training data and the remaining data separately and their results are shown in Figures 15 and 16, respectively. The red lines represent the control limits, and the blue lines represent the monitoring statistics of the DFIM procedure. During the training period, no false alarm is raised by the DFIM procedure. On the other hand, an out-of-control alarm is raised right before August 10 12:00, which demonstrates the effective detection power of the DFIM procedure.

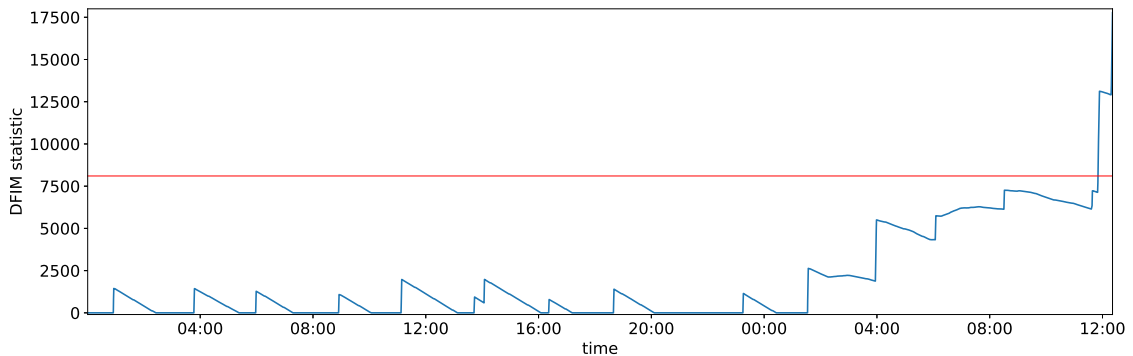


Figure 16: The DFIM procedure on out-of-control data from August/9 0:00 to August/10 13:00

3.5 Conclusion

We propose a new distribution-free procedure based on SVD for monitoring an image process in the presence of cross- and auto-correlation and general marginals. The determination of control limits for the proposed procedure does not require trial-and-error simulation, which ensures the efficient implementation of the procedure in practice. Numerical experiments on simulated and real processes demonstrate the advantages of our procedure compared with two baseline procedures and its effectiveness in anomaly detection. The DFIM procedure is directly motivated by an application of monitoring a battery coating process and it can be easily generalized to a broad

range of applications where image monitoring is needed.

CHAPTER IV

FEASIBILITY DETERMINATION WITH GAUSSIAN PROCESS FOR MULTIPLE STOCHASTIC CONSTRAINTS

We consider the problem of finding a set of feasible inputs in the presence of constraints on multiple performance measures. For example, a manager may want to identify a set of control parameters, such as temperature and pressure levels, that ensure that produced products have mean deviations from multiple target quality measures smaller than certain constant thresholds. The constraints on the mean deviations are stochastic in that they can be estimated only through noisy observations either from real data or simulated data. Feasibility determination with stochastic constraints arises in many industrial applications where similar inputs are more likely to have similar performance measures and noisy observations are expensive.

This chapter proposes a Bayesian procedure for feasibility determination in the presence of multiple stochastic constraints. More specifically, we extend the procedure due to [1] to multiple constraints. Each performance measure function over inputs is modeled as a GP, and we use a value-of-information (VOI) function as an acquisition function for the decision on which input to test next. One of the bottlenecks for implementing the proposed procedure is the calculation of the VOI function, which can take long. To accelerate the computation, we propose another version with an approximation for the value-of-information function that is quick and easy to calculate. We also consider three strategies to fit GPs to performance measure functions: fitting an independent GP to each performance measure function, fitting a multi-task GP to all performance measure functions, and a hybrid strategy using independent GPs first and switching to a multi-task GP. We prove the convergence of our proposed

procedures and show their effectiveness with three fitting strategies by numerical experiments.

4.1 Problem Formulation

We define our problem and notation in Section 4.1.1 and review the Bayesian feasibility determination procedure for a single stochastic constraint due to [1] in Section 4.1.2.

4.1.1 Problem and Notation

We consider an input space $\mathbf{X} \subset \mathbb{R}^m$ where \mathbf{X} is a bounded finite set. Each input $\mathbf{x} \in \mathbf{X}$ has k performance measures with mean functions $\mu^{(1)}(\mathbf{x}), \dots, \mu^{(k)}(\mathbf{x})$ and sampling standard deviation $\sigma^{(1)}, \dots, \sigma^{(k)}$. For each performance measure k , we assume the sampling standard deviation $\sigma^{(k)}$ is a constant for all inputs. Our goal is to identify a set $\mathbb{F} \subset \mathbf{X}$, called a feasible set, such that $\mu^{(j)}(\mathbf{x}) \leq d_j$ for all $\mathbf{x} \in \mathbb{F}$ and all $j \in \{1, \dots, k\}$ where d_j 's are pre-specified constant thresholds.

We make the following assumption for each performance measure j .

Assumption 3. For any $\mathbf{x} \in \mathbf{X}$, $\mu^{(j)}(\mathbf{x}) \sim N(\delta^{(j)}(\mathbf{x}), \Sigma^{(j)}(\mathbf{x}, \mathbf{x}))$, and the collection $\mu^{(j)} \sim GP(\delta^{(j)}, \Sigma^{(j)})$. $GP(\delta^{(j)}, \Sigma^{(j)})$ is a Gaussian process with mean function $\delta^{(j)}$ and covariance function $\Sigma^{(j)}$ defined as:

$$\begin{aligned}\delta^{(j)}(\mathbf{x}) &= \mathbb{E}[\mu^{(j)}(\mathbf{x})], \\ \Sigma^{(j)}(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(\mu^{(j)}(\mathbf{x}) - \delta^{(j)}(\mathbf{x}))(\mu^{(j)}(\mathbf{x}') - \delta^{(j)}(\mathbf{x}'))],\end{aligned}$$

for any $\mathbf{x}, \mathbf{x}' \in \mathbf{X}$.

We take the Bayesian approach by placing a GP with prior mean $\delta_0^{(j)}(\cdot)$ and covariance $\Sigma_0^{(j)}(\cdot, \cdot)$ on each performance measure j . At each stage denoted by n , an input $\mathbf{x}_n \in \mathbf{X}$ is chosen which maximizes an acquisition function, and we obtain new

observations of performance measures from the input, denoted by $Y_n^{(1)}, \dots, Y_n^{(k)}$. The observations are assumed to satisfy the following assumption.

Assumption 4. For each performance measure j , $Y_n^{(j)} = \mu^{(j)}(\mathbf{x}_n) + \epsilon_n^{(j)}$ where $\epsilon_n^{(j)}$ are i.i.d. normal noise with mean zero standard deviation $\sigma^{(j)}$. Thus, $Y_n^{(j)} | \mu^{(j)}(\mathbf{x}_n) \stackrel{i.i.d.}{\sim} N(\mu^{(j)}(\mathbf{x}_n), (\sigma^{(j)})^2)$ and $Y_n^{(j)} \stackrel{i.i.d.}{\sim} N(\delta^{(j)}(\mathbf{x}_n), \Sigma^{(j)}(\mathbf{x}, \mathbf{x}) + (\sigma^{(j)})^2)$.

These observations are then used to compute the posterior mean $\delta_n^{(j)}(\cdot)$ and covariance $\Sigma_n^{(j)}(\cdot, \cdot)$. These steps are repeated until the stage counter n reaches the maximum stage allowed N . An estimated feasible set \mathbf{F}_n is determined according to the posterior means as follows:

$$\mathbf{F}_n = \{\mathbf{x} \in \mathbf{X} : \delta_n^{(j)}(\mathbf{x}) \leq d_j, \forall j = 1, \dots, k\}.$$

Let $\mathbf{F}_n^{(j)} = \{\mathbf{x} : \delta_n^{(j)}(\mathbf{x}) \leq d_j\}$, then $\mathbf{F}_n = \bigcap_{j=1}^k \mathbf{F}_n^{(j)}$.

4.1.2 Feasibility Determination for One Constraint

For solving feasibility determination with a single constraint (i.e. $k = 1$), [1] proposes a Bayesian sequential procedure with a GP and uses a value-of-information (VOI) function to decide which input to choose for sampling at each stage. An estimated feasible set for constraint 1 is denoted by $\mathbf{F}^{(1)}$ where the superscript (1) represents the first (and the only) constraint. The superscript is changed to (j) for $j = 1, 2, \dots, k$ when we consider multiple constraints. For $\mathbf{F}^{(1)}$, a reward function $R^{(1)}(\cdot)$ for constraint 1 is defined as follows:

$$R^{(1)}(\mathbf{F}^{(1)}) = \sum_{\mathbf{x} \in \mathbf{F}^{(1)}} \mathbf{1}_{\{\mu^{(1)}(\mathbf{x}) \leq d_1\}} + \sum_{\mathbf{x} \notin \mathbf{F}^{(1)}} \mathbf{1}_{\{\mu^{(1)}(\mathbf{x}) > d_1\}}. \quad (13)$$

where $\mathbf{1}_{\{A\}} = 1$ if A is true, and 0 otherwise.

The estimated feasible set at stage n is defined as

$$\mathbf{F}_n^{(1)} = \arg \max_{\mathbf{F}^{(1)} \subset \mathbf{X}} \mathbb{E}_n[R^{(1)}(\mathbf{F}^{(1)})]$$

and it can be shown that $\mathbf{F}_n^{(1)} = \{\mathbf{x} \in \mathbf{X} : \delta_n^{(1)}(\mathbf{x}) \leq d_1\}$ maximizes the expected reward.

Following the Gaussian assumption, the expected reward at stage n is computed as:

$$\begin{aligned} \mathbb{E}_n[R^{(1)}(\mathbf{F}_n^{(1)})] &= \sum_{\mathbf{x} \in \mathbf{F}_n^{(1)}} \Pr\{\mu^{(1)}(\mathbf{x}) \leq d_1\} + \sum_{\mathbf{x} \notin \mathbf{F}_n^{(1)}} \Pr\{\mu^{(1)}(\mathbf{x}) > d_1\} \\ &= \sum_{\mathbf{x} \in \mathbf{F}_n^{(1)}} \Phi\left(\frac{d_1 - \delta_n^{(1)}(\mathbf{x})}{\sqrt{\Sigma_n^{(1)}(\mathbf{x})}}\right) + \sum_{\mathbf{x} \notin \mathbf{F}_n^{(1)}} \left[1 - \Phi\left(\frac{d_1 - \delta_n^{(1)}(\mathbf{x})}{\sqrt{\Sigma_n^{(1)}(\mathbf{x})}}\right)\right]. \end{aligned}$$

Let

$$p_n^{(1)}(\mathbf{x}; \mathbf{F}_n^{(1)}) = \begin{cases} \Phi\left(\frac{d_1 - \delta_n^{(1)}(\mathbf{x})}{\sqrt{\Sigma_n^{(1)}(\mathbf{x})}}\right), & \text{if } \mathbf{x} \in \mathbf{F}_n^{(1)}, \\ 1 - \Phi\left(\frac{d_1 - \delta_n^{(1)}(\mathbf{x})}{\sqrt{\Sigma_n^{(1)}(\mathbf{x})}}\right), & \text{if } \mathbf{x} \notin \mathbf{F}_n^{(1)}. \end{cases}$$

We can then simplify the expected reward as:

$$\mathbb{E}_n[R^{(1)}(\mathbf{F}_n^{(1)})] = \sum_{\mathbf{x} \in \mathbf{X}} p_n^{(1)}(\mathbf{x}; \mathbf{F}_n^{(1)}).$$

At stage n , define the VOI of an input \mathbf{x} as:

$$\text{VOI}_n^{(1)}(\mathbf{x}) = \mathbb{E}_n \left[\mathbb{E}_{n+1}[R^{(1)}(\mathbf{F}_{n+1}^{(1)}) \mid \mathbf{x}_{n+1} = \mathbf{x}] - \mathbb{E}_n[R^{(1)}(\mathbf{F}_n^{(1)})] \right] \quad (14)$$

$$= \mathbb{E}_n \left[\sum_{\mathbf{x}' \in \mathbf{X}} p_n^{(1)}(\mathbf{x}'; \mathbf{F}_{n+1}^{(1)}) \mid \mathbf{x}_{n+1} = \mathbf{x} \right] - \sum_{\mathbf{x}' \in \mathbf{X}} p_n^{(1)}(\mathbf{x}'; \mathbf{F}_n^{(1)}) \quad (15)$$

$$= \sum_{\mathbf{x}' \in \mathbf{X}} \left\{ \mathbb{E}_n \left[p_n^{(1)}(\mathbf{x}'; \mathbf{F}_{n+1}^{(1)}) \mid \mathbf{x}_{n+1} = \mathbf{x} \right] - p_n^{(1)}(\mathbf{x}'; \mathbf{F}_n^{(1)}) \right\}. \quad (16)$$

[1] derives the formula for computing $\mathbb{E}_n \left[p_n^{(1)}(\mathbf{x}'; \mathbf{F}_{n+1}^{(1)}) \mid \mathbf{x}_{n+1} = \mathbf{x} \right]$ and $\text{VOI}_n^{(1)}(\mathbf{x})$ based on $\delta_n^{(1)}$ and $\Sigma_n^{(1)}$. The input for sampling at stage $n+1$ is chosen as:

$$\mathbf{x}_{n+1} = \arg \max_{\mathbf{x} \in \mathbf{X}} \text{VOI}_n^{(1)}(\mathbf{x}). \quad (17)$$

One naive way to find the solution to (17) is by enumeration, which is computing the value of $\text{VOI}_n^{(1)}(\mathbf{x})$ for every \mathbf{x} and finding the \mathbf{x} with the largest value of $\text{VOI}_n^{(1)}(\mathbf{x})$. The enumeration is easy to implement but could be computationally expensive. As

stage n goes to infinity, [1] shows that the expected reward $\mathbb{E}_n[R^{(1)}(\mathbf{F}_n^{(1)})]$ converges to its maximum almost surely which, in turn, implies that $\mathbf{F}_n^{(1)}$ converges to \mathbb{F} .

4.2 Feasibility Determination for Multiple Constraints

We generalize the procedure due to [1] to handle feasibility determination with multiple performance measures and propose an approximated VOI to accelerate the computation of VOI and the selection of the next input for sampling at each stage.

4.2.1 VOI of Multiple Constraints

In the presence of multiple constraints, an estimated feasible set at stage n is the intersection of multiple estimated feasible sets for each constraint. That is, let $\mathbf{F}_n^{(j)} = \{\mathbf{x} : \delta_n^{(j)}(\mathbf{x}) \leq d_j\}$ denote an estimated feasible set for constraint j . Then, $\mathbf{F}_n = \bigcap_{j=1}^k \mathbf{F}_n^{(j)}$. Analogously, we define the VOI of an input for multiple constraints as the summation of VOI's calculated for each individual constraint as discussed in Section 4.1.2. More specifically, define the reward function for constraint j as

$$R^{(j)}(\mathbf{F}_n^{(j)}) = \sum_{\mathbf{x} \in \mathbf{F}_n^{(j)}} \mathbf{1}_{\{\mu^{(j)}(\mathbf{x}) \leq d_j\}} + \sum_{\mathbf{x} \notin \mathbf{F}_n^{(j)}} \mathbf{1}_{\{\mu^{(j)}(\mathbf{x}) > d_j\}},$$

and the VOI for constraint j as

$$\begin{aligned} \text{VOI}_n^{(j)}(\mathbf{x}) &= \mathbb{E}_n \left[\mathbb{E}_{n+1} [R^{(j)}(\mathbf{F}_{n+1}^{(j)}) \mid \mathbf{x}_{n+1} = \mathbf{x}] - \mathbb{E}_n [R^{(j)}(\mathbf{F}_n^{(j)})] \right] \\ &= \sum_{\mathbf{x}' \in \mathbf{X}} \left\{ \mathbb{E}_n \left[p_n^{(j)}(\mathbf{x}'; \mathbf{F}_{n+1}^{(j)}) \mid \mathbf{x}_{n+1} = \mathbf{x} \right] - p_n^{(j)}(\mathbf{x}'; \mathbf{F}_n^{(j)}) \right\}. \end{aligned} \quad (18)$$

Then the VOI of an input \mathbf{x} at stage n is defined as the sum of all $\text{VOI}_n^{(j)}(\mathbf{x})$ for $j = 1, 2, \dots, k$:

$$\text{VOI}_n(\mathbf{x}) = \sum_{j=1}^k \text{VOI}_n^{(j)}(\mathbf{x}). \quad (19)$$

Our procedure, namely the GP feasibility determination procedure for multiple constraints (GPFDM), incorporates the VOI function (19) into the feasibility deter-

mination procedure due to [1] and its description is given in Algorithm 2.

- 1 **Setup:** Specify the input space \mathbf{X} , thresholds d_1, d_2, \dots, d_k , the number of initial inputs to sample n_0 , and maximum stage N . Also, specify prior $\delta_0^{(1)}, \Sigma_0^{(1)}, \dots, \delta_0^{(k)}, \Sigma_0^{(k)}$.
- 2 **Initialization:** Sample n_0 number of \mathbf{x} 's randomly from \mathbf{X} and obtain $Y_0^{(1)}, Y_0^{(2)}, \dots, Y_0^{(k)}$ at each sampled \mathbf{x} . Then use them to get posterior $\delta_{n_0}^{(1)}, \Sigma_{n_0}^{(1)}, \dots, \delta_{n_0}^{(k)}, \Sigma_{n_0}^{(k)}$. Noise variances are estimated using maximum likelihood estimators. Set $n = n_0$.
- 3 **Update:** Choose $\mathbf{x}_{n+1} = \arg \max_{\mathbf{x} \in \mathbf{X}} \text{VOI}_n(\mathbf{x})$. Take $Y_{n+1}^{(1)}, Y_{n+1}^{(2)}, \dots, Y_{n+1}^{(k)}$ at \mathbf{x}_{n+1} . Compute posterior parameters $\delta_{n+1}^{(1)}, \Sigma_{n+1}^{(1)}, \dots, \delta_{n+1}^{(k)}, \Sigma_{n+1}^{(k)}$ and find $\mathbf{F}_{n+1}^{(j)} = \{\mathbf{x} \in \mathbf{X} : \delta_{n+1}^{(j)}(\mathbf{x}) \leq d_j\}$ for $j = 1, 2, \dots, k$. Set $n = n + 1$.
- 4 **Termination Rule:** If $n < N$, go back to **Update**. Otherwise, stop sampling and go to **Feasibility Decisions**.
- 5 **Feasibility Decisions:** Return the feasible set as $\mathbf{F}_n = \bigcap_{j=1}^k \mathbf{F}_n^{(j)}$.

Algorithm 2: Procedure GPFDM

4.2.2 Accelerated Computation of VOI

Let $|A|$ represent the cardinality of a set A . Then the VOI function can be expressed as a summation of $|\mathbf{X}|$ terms using equations (18) and (19) as follows:

$$\text{VOI}_n(\mathbf{x}) = \sum_{\mathbf{x}' \in \mathbf{X}} \sum_{j=1}^k \left\{ \mathbb{E}_n[p_n^{(j)}(\mathbf{x}'; \mathbf{F}_{n+1}^{(j)}) | \mathbf{x}_{n+1} = \mathbf{x}] - p_n^{(j)}(\mathbf{x}'; \mathbf{F}_n^{(j)}) \right\}. \quad (20)$$

The computation of $\mathbb{E}_n[p_n^{(j)}(\mathbf{x}'; \mathbf{F}_{n+1}^{(j)}) | \mathbf{x}_{n+1} = \mathbf{x}]$ over all $\mathbf{x}' \in \mathbf{X}$ for each $j \in \{1, \dots, k\}$ is time-consuming. When $|\mathbf{X}|$ is large, it takes a significant amount of time to compute and maximize $\text{VOI}_n(\mathbf{x})$ in each stage. To make the GPFDM procedure computationally tractable when the input set is large, we propose an approximation of the VOI which can be quickly computed without deteriorating the accuracy of feasibility determination. The idea is to replace the whole input set \mathbf{X} with a subset of \mathbf{X} in the equation (20), so that we only need to compute $\mathbb{E}_n[p_n^{(j)}(\mathbf{x}'; \mathbf{F}_{n+1}^{(j)}) | \mathbf{x}_{n+1} = \mathbf{x}]$ for a small number of $\mathbf{x}' \in \mathbf{X}$. Let $\mathcal{X} = \{\text{inputs sampled by stage } n\} \cup \{\mathbf{x}\}$ be the set of inputs that have been sampled up to stage n and a candidate input \mathbf{x} . The VOI

can be approximated by

$$\text{VOI}_n^*(\mathbf{x}) = \sum_{\mathbf{x}' \in \mathcal{X}} \sum_{j=1}^k \left\{ \mathbb{E}_n[p_n^{(j)}(\mathbf{x}'; \mathbf{F}_{n+1}^{(j)}) | \mathbf{x}_{n+1} = \mathbf{x}] - p_n^{(j)}(\mathbf{x}'; \mathbf{F}_n^{(j)}) \right\}. \quad (21)$$

By replacing the whole input set \mathbf{X} with a subset \mathcal{X} , the number of terms required to compute in the summation is significantly reduced. In addition, our numerical study shows that $\text{VOI}_n(\mathbf{x})$ and $\text{VOI}_n^*(\mathbf{x})$ tend to be maximized at the same input \mathbf{x} , which indicates the sampling decisions based on the two VOI's tend to be same and the performance of the GPFDM procedure will not be severely affected by the approximation. The numerical study is given in Section 4.4.

4.3 Convergence Analysis

In this section, we prove the convergence of the \mathcal{GPFDM} procedure with the exact and approximated VOI's. Consider the sum of expected reward functions for all constraints which is given as

$$\sum_{j=1}^k \mathbb{E}_n[R^{(j)}(\mathbf{F}_n^{(j)})] = \sum_{j=1}^k \mathbb{E}_n \left[\sum_{\mathbf{x} \in \mathbf{F}_n^{(j)}} \mathbf{1}_{\{\mu^{(j)}(\mathbf{x}) \leq d_j\}} + \sum_{\mathbf{x} \notin \mathbf{F}_n^{(j)}} \mathbf{1}_{\{\mu^{(j)}(\mathbf{x}) > d_j\}} \right].$$

We show that the sum converges to its maximum almost surely under the GPFDM procedure as n goes to infinity, which implies the GPFDM procedure determines the feasibility set correctly in the limit. Note that the maximum of $\mathbb{E}_n[R^{(j)}(\mathbf{F}_n^{(j)})]$ is $|\mathbf{X}|$ for each individual constraint j , and the maximum of $\sum_{j=1}^k \mathbb{E}_n[R^{(j)}(\mathbf{F}_n^{(j)})]$ is $k|\mathbf{X}|$.

We first introduce fundamental results for the feasibility determination procedure for a single constraint due to [1], based on which we prove the convergence of the GPFDM procedure.

Lemma 4.3.1. *For any constraint j , if an input $\mathbf{x} \in \mathbf{X}$ is sampled infinitely many times, then*

$$\lim_{n \rightarrow \infty} \text{VOI}_n^{(j)}(\mathbf{x}) = 0.$$

Lemma 4.3.2. For any constraint j , if $\lim_{n \rightarrow \infty} \text{VOI}_n^{(j)}(\mathbf{x}) = 0$ for all $\mathbf{x} \in \mathbf{X}$, then $\mathbb{E}_n[R^{(j)}(\mathbf{F}_n^{(j)})]$ converges to its maximum almost surely as n goes to infinity.

The proofs of Lemmas 4.3.1 and 4.3.2 are given in [1].

Lemma 4.3.3. Under the GPFDM procedure,

$$\lim_{n \rightarrow \infty} \text{VOI}_n(\mathbf{x}) = 0$$

for all $\mathbf{x} \in \mathbf{X}$.

Proof. Let $\tilde{\mathbf{X}} = \{\mathbf{x} \in \mathbf{X} : \lim_{n \rightarrow \infty} \text{VOI}_n(\mathbf{x}) = 0\}$. We first show $\tilde{\mathbf{X}}$ is not empty. As n goes to infinity, there must exist an input $\tilde{\mathbf{x}} \in \mathbf{X}$ that is sampled for infinitely many times. According to Lemma 4.3.1, $\lim_{n \rightarrow \infty} \text{VOI}_n^{(j)}(\tilde{\mathbf{x}}) = 0$ for any individual constraint j if $\tilde{\mathbf{x}}$ is sampled infinitely many times. It follows that

$$\lim_{n \rightarrow \infty} \text{VOI}_n(\tilde{\mathbf{x}}) = \lim_{n \rightarrow \infty} \sum_{j=1}^k \text{VOI}_n^{(j)}(\tilde{\mathbf{x}}) = 0,$$

and thus $\tilde{\mathbf{x}} \in \tilde{\mathbf{X}}$.

Next we prove by contradiction that $\tilde{\mathbf{X}} = \mathbf{X}$. Suppose $\tilde{\mathbf{X}} \neq \mathbf{X}$. Then, for any $\mathbf{x}' \in \mathbf{X} \setminus \tilde{\mathbf{X}}$, $\lim_{n \rightarrow \infty} \text{VOI}_n(\mathbf{x}') > 0$. It follows that there exists at least one constraint j for which $\lim_{n \rightarrow \infty} \text{VOI}_n^{(j)}(\mathbf{x}') > 0$. According to the contraposition of Lemma 4.3.1, there exists an integer $T_{\mathbf{x}'} > 0$ such that $\mathbf{x}_n \neq \mathbf{x}'$ for all $n > T_{\mathbf{x}'}$. Let

$$\hat{T} = \max_{\mathbf{x}' \in \mathbf{X} \setminus \tilde{\mathbf{X}}} T_{\mathbf{x}'}$$

Then $\mathbf{x}_n \in \tilde{\mathbf{X}}$ for all $n > \hat{T}$. Since the GPFDM procedure chooses an input with the maximal VOI value to sample in each stage, it follows that

$$\text{VOI}_n(\mathbf{x}_n) \geq \text{VOI}_n(\mathbf{x}')$$

for all $\mathbf{x}' \in \mathbf{X} \setminus \tilde{\mathbf{X}}$ and all $n > \hat{T}$. Since $\lim_{n \rightarrow \infty} \text{VOI}_n(\mathbf{x}) = 0$ for all $\mathbf{x} \in \tilde{\mathbf{X}}$, there exists an integer $T_\xi > 0$ for any constant $\xi > 0$ such that

$$\xi > \text{VOI}_n(\mathbf{x}_n) \geq \text{VOI}_n(\mathbf{x}')$$

for all $n > T_\xi$ and all $\mathbf{x}' \in \mathbf{X} \setminus \tilde{\mathbf{X}}$. This contradicts $\lim_{n \rightarrow \infty} \text{VOI}_n(\mathbf{x}') > 0$. Therefore $\tilde{\mathbf{X}} = \mathbf{X}$. \square

Theorem 4.3.4. *As n goes to infinity, $\sum_{j=1}^k \mathbb{E}_n[R^{(j)}(\mathbf{F}_n^{(j)})]$ converges to its maximum almost surely under the GPFDM procedure.*

Proof. According to [1], for any constraint j and any finite n , $\text{VOI}_n^{(j)}(\mathbf{x}) > 0$ for all $\mathbf{x} \in \mathbf{X}$. It follows from Lemma 4.3.3 that under the GPFDM procedure,

$$\lim_{n \rightarrow \infty} \text{VOI}_n^{(j)}(\mathbf{x}) = 0$$

for all $\mathbf{x} \in \mathbf{X}$ and all $j \in \{1, \dots, k\}$. From Lemma 4.3.2, we have $\lim_{n \rightarrow \infty} \mathbb{E}_n[R^{(j)}(\mathbf{F}_n^{(j)})] = |\mathbf{X}|$ almost surely, and consequently,

$$\lim_{n \rightarrow \infty} \sum_{j=1}^k \mathbb{E}_n[R^{(j)}(\mathbf{F}_n^{(j)})] = k|\mathbf{X}|$$

almost surely under the GPFDM procedure. \square

Corollary 4.3.1. *As n goes to infinity, $\sum_{j=1}^k \mathbb{E}_n[R^{(j)}(\mathbf{F}_n^{(j)})]$ converges to its maximum almost surely under the GPFDM procedure using the approximated VOI defined in (21).*

Theorem 4.3.4 and Corollary 4.3.1 imply that as stage n goes to infinity, the GPFDM procedure is expected to correctly identify the feasibility set.

4.4 Numerical Results

In this section we demonstrate the effectiveness of the approximated VOI and the GPFDM procedure numerically.

For fitting GPs in the \mathcal{GPFDS} procedure, we consider three strategies:

- (1) Independent GPs: Fit k number of independent GPs to predict $\mu^{(1)}(\cdot), \mu^{(2)}(\cdot), \dots, \mu^{(k)}(\cdot)$.

- (2) Multi-task GP: Fit a multi-task GP to predict all performance measures. We adopt a multi-task GP with covariance defined by the intrinsic model of coregionalization from [64]. Then the covariance of a multi-task GP is modeled as:

$$\Sigma_{\text{multi}} = \Sigma_{\text{t}} \otimes \Sigma_{\text{x}},$$

where \otimes denotes the Kronecker product, Σ_{x} measures relation between inputs, and Σ_{t} measures the relation between tasks which are performance measures in our problem. This covariance model can take into account the underlying relation between performance measures to provide better predictions. However, it requires more estimation effort compared to independent GPs as it requires the estimation of the covariance matrix among performance measures in addition to that among inputs.

- (3) Hybrid GP: Fit independent GPs for the first ten stages and then fit a multi-task GP afterwards. This hybrid version is considered because an estimated covariance matrix among performance measures is likely to be inaccurate and poor with only a few observations available during early stage and it may negatively affect the performance of the GPFDM procedure with a multi-task GP.

All three procedures use the approximated VOI defined in (21).

The performance of the procedure is tested on two numerical examples. The first example in Section 4.4.1 has a one-dimensional input space with two performance measures and the second example in Section 4.4.2 has a two-dimensional input space with two performance measures.

4.4.1 One-Dimensional Example

In the first example, we consider a problem with a one-dimensional input space and two constraints. The input space is defined as $\mathbf{X} = \{0, 0.1, 0.2, \dots, 20.0\}$. For each $x \in \mathbf{X}$, two mean performance measures associated with the constraints are defined

by the following two functions:

$$\begin{aligned}\mu^{(1)}(x) &= \sin(0.4789\pi x), \\ \mu^{(2)}(x) &= \frac{x}{10} \sin(0.189\pi x) + \sin(0.4789\pi x).\end{aligned}$$

We aim to identify inputs satisfying $\mu^{(1)}(x) \leq 0$ and $\mu^{(2)}(x) \leq 0$. Figure 17 visualizes the two mean functions and feasible regions with respect to the two constraints by shaded regions.

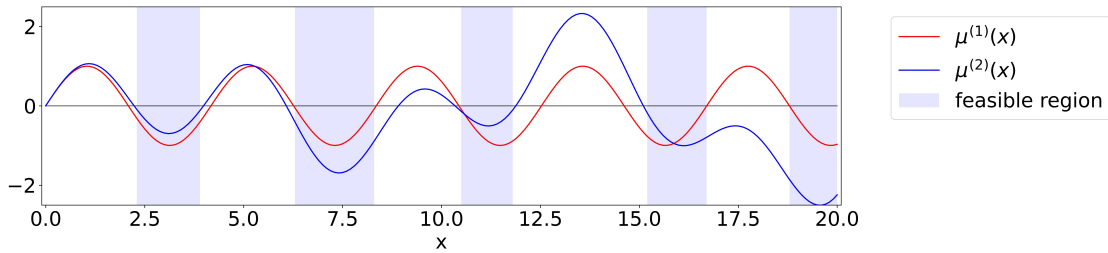


Figure 17: The mean functions and feasible region for Example 1.

The maximum stage is set to $N = 50$, and the number of initial inputs to sample is $n_0 = 5$. We first compare the performance of the GPFDM procedure using approximated VOI against exact VOI. Figure 18 shows average proportions of inputs with correct feasibility decisions over 50 macro-replications (PCD) against the number of stages. Both procedures fit independent GPs. The average running time per replication is 3550 seconds for exact VOI and 480 seconds for approximated VOI.

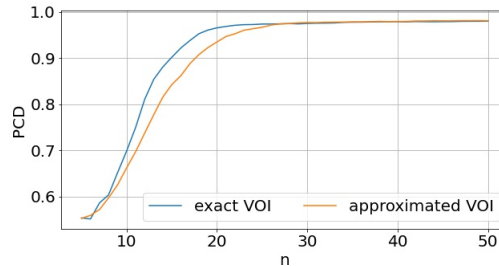


Figure 18: PCD vs. n for verifying approximated VOI using an example with a one-dimensional input space and two constraints.

The approximate VOI works well and we consider GPFDM with the approximated

VOI for comparing different fitting strategies. We test five settings of sampling standard deviations: (1) $\sigma^{(1)} = \sigma^{(2)} = 0.1$; (2) $\sigma^{(1)} = \sigma^{(2)} = 0.5$; (3) $\sigma^{(1)} = \sigma^{(2)} = 1$; (4) $\sigma^{(1)} = 0.1, \sigma^{(2)} = 0.5$; and (5) $\sigma^{(1)} = 0.5, \sigma^{(2)} = 0.1$. All procedures with the three GP fitting strategies start with the same n_0 initial inputs and the experiment is repeated 50 times (so the number of macro-replications is also 50).

Figure 19 shows PCD against the number of stages.

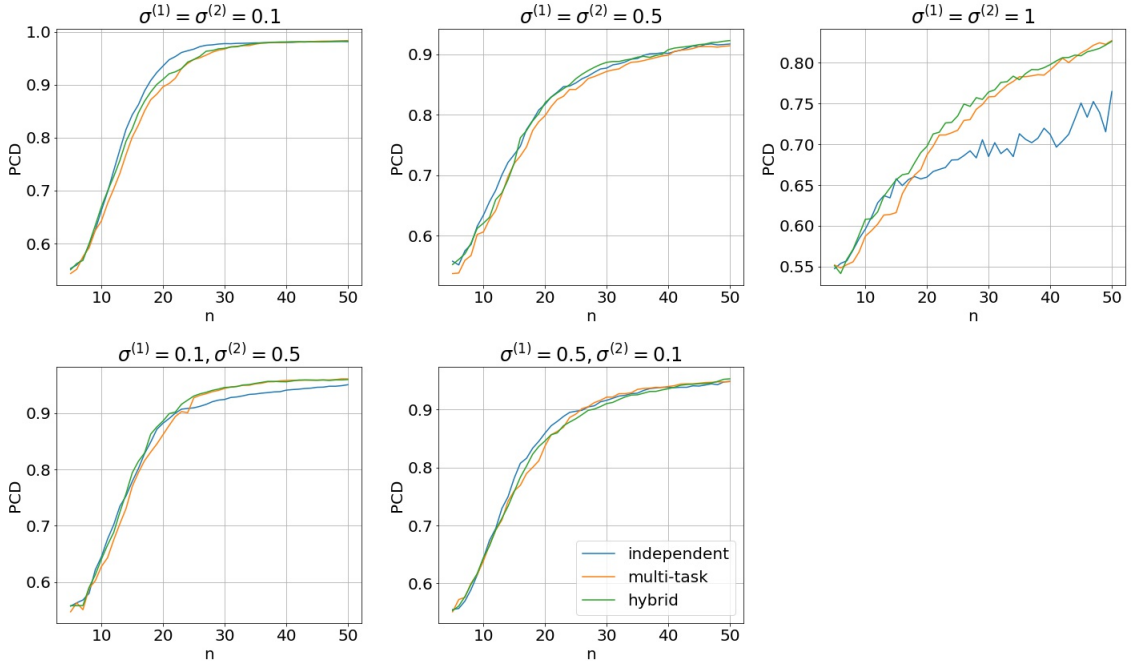


Figure 19: PCD vs. n for comparing fitting strategies using an example with a one-dimensional input space and two constraints.

When sampling variation is small as $\sigma^{(1)} = \sigma^{(2)} = 0.1$, it is relatively easy to accurately estimate the mean performance measures, and independent GPs are capable of producing good predictions. Even though the two performance measures demonstrate high correlation between the performance measures as shown in Figure17, a multi-task GP does not perform any better than the independent GPs in this case because it introduces more estimation error and could suffer more from the inaccurate estimation of the covariance matrix Σ_t . In fact, it slightly worsens the performance according to the first subplot of Figure 19.

When sampling variation is large as $\sigma^{(1)} = \sigma^{(2)} = 1$, it is much harder to estimate the mean performance measures. The top right subplot of Figure 19 shows that utilizing the relation between the performance measures through a multi-task GP results in much better PCD than using independent GPs. The hybrid fitting strategy seems to capture advantages of the independent GPs and the multi-task GP, showing the most robust performance over all cases tested in the numerical study.

4.4.2 Two-Dimensional Example

The second example considers a feasibility determination problem with a two-dimensional input space and two constraints. The input space is defined as $\mathbf{X} = \{-1.0, -0.8, \dots, 1.0\}^2$. The true mean performance measures are defined by the following functions which are adapted from the Holder Table function and the Goldstein-Price function from [65]:

$$\mu^{(1)}(\mathbf{x}) = 0.7 - \left| \sin(x_1) \cos(x_2) \exp \left(\left| 1 - \frac{\sqrt{x_1^2 + x_2^2}}{\pi} \right| \right) \right|,$$

$$\mu^{(2)}(\mathbf{x}) = \frac{1}{2.427} \left[\log \left([1 + (\bar{x}_1 + \bar{x}_2 + 1)^2 (19 - 14\bar{x}_1 + 3\bar{x}_1^2 - 14\bar{x}_2 + 6\bar{x}_1\bar{x}_2 + 3\bar{x}_2^2)] \right. \right. \\ \left. \left. [30 + (2\bar{x}_1 - 3\bar{x}_2)^2 (18 - 32\bar{x}_1 + 12\bar{x}_1^2 + 48\bar{x}_2 - 36\bar{x}_1\bar{x}_2 + 27\bar{x}_2^2)] \right) - 8.693 \right] - 2,$$

where $\bar{x}_i = 4x_i - 2$ for $i = 1, 2$. Figure 20 visualizes the two mean functions and the feasible region defined by $\mu^{(1)}(\mathbf{x}) \leq 0$ and $\mu^{(2)}(\mathbf{x}) \leq 0$.

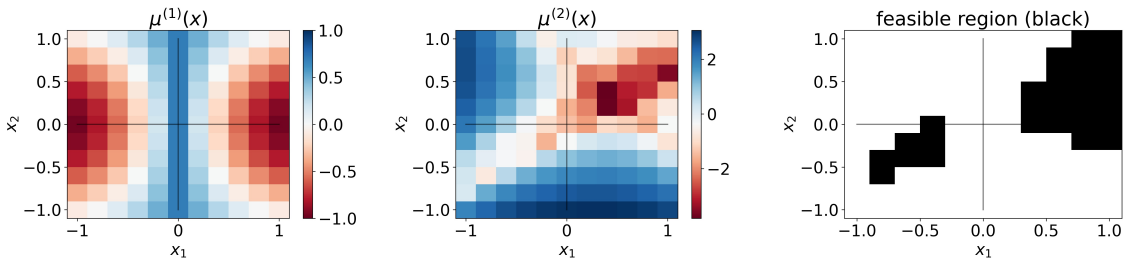


Figure 20: The mean functions and feasible region for Example 2.

Similar to the first example, we set $N = 50, n_0 = 5$ and the same five settings of

sampling standard deviations from Section 4.4.1 are considered. We first compare the performance of the GPFDM procedure using approximated VOI against exact VOI, and result is shown in Figure 21. The average running time per replication is 1273 seconds for exact VOI and 292 seconds for approximated VOI.

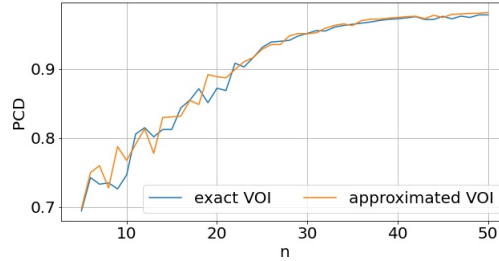


Figure 21: PCD vs. n for verifying approximated VOI using an example with a two-dimensional input space and two constraints.

Results of comparing three fitting strategies are shown in Figure 22. All procedures use approximated VOI.

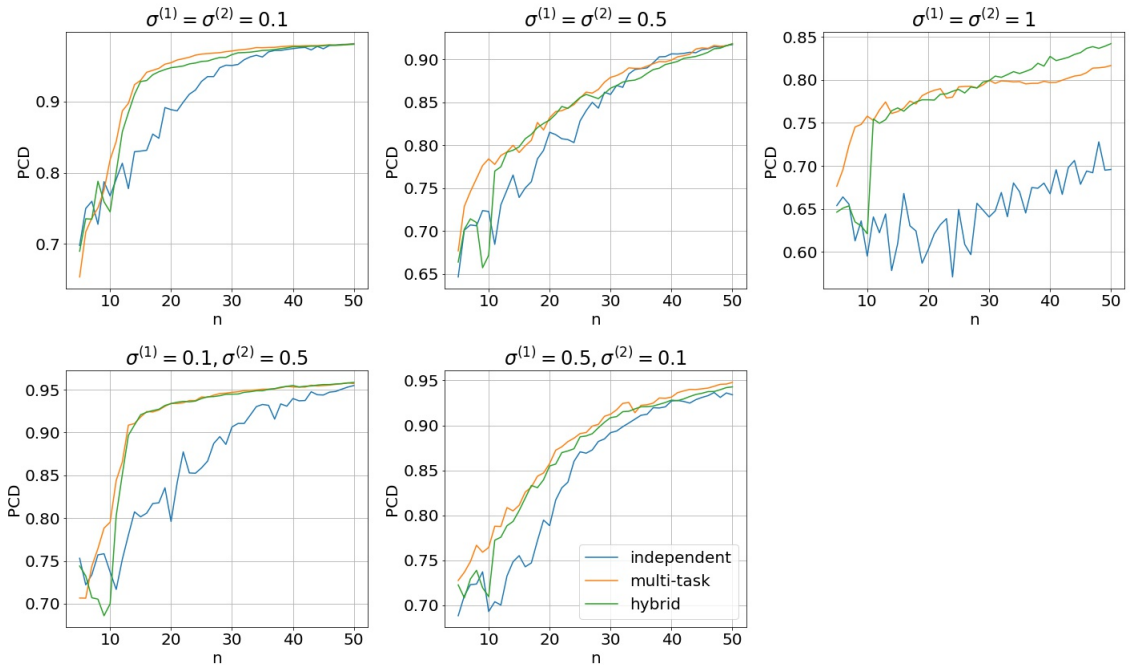


Figure 22: PCD vs. n for comparing fitting strategies using an example with a two-dimensional input space and two constraints.

This example is more difficult than the first example as it has a higher dimension

of inputs and more complicated structure of mean performance functions. In this example, a multi-task GP takes the advantage of considering the underlying relation between two performance measures and performs better than independent GPs for all settings of sampling standard deviations. Considering PCDs over all ranges of n , the hybrid fitting strategy performs best overall.

4.5 Conclusion

In this study, we propose the GPFDM procedure that utilizes GPs for feasibility determination in the presence of multiple stochastic constraints. Sampling decisions are made sequentially based on a VOI function calculated over all individual constraints. We also propose an approximated VOI function to accelerate the computation of calculating the VOI function at every stage. The GPFDM procedure is expected to correctly determine the feasibility of all inputs asymptotically when the number of stages goes to infinity. We compare three different fitting strategies for GPs via numerical experiments. The numerical study shows that the GPFDM procedure works better when fitting strategies with a multi-task GP are utilized for a difficult problem with either high sampling variances or complicated mean performance functions.

CHAPTER V

CONCLUSION

In Chapter 2, we present the DFMM procedure for monitoring a general multivariate time-series without assuming a specific marginal distribution or correlation structure. The control limit of the procedure can be determined analytically based on the target in-control performance, which is a desired property to make the procedure easy to generalize in real practice. Our studies of simulated processes and a real-data example of wafer etching process show that the procedure is effective and competitive compared to baseline models.

In Chapter 3, we propose the DFIM procedure by formulating two monitoring statistics from image data and applying the DFMM procedure on the statistics. The DFIM procedure enjoys the benefits of two different statistics and the desired distribution-free property of the DFMM procedure. Numerical experiments with simulated processes demonstrate that the DFIM procedure is an appealing option to adopt for various scenarios in terms of the amount of data and time required to set up and shift detection performance. One assumption the DFIM procedure makes is the target pattern of image data is a rank-one matrix. A possible direction for future research is to generalize this assumption to allow more complex pattern to be the target of monitoring.

In Chapter 4, we present the GPFDM procedure that utilizes GPs to solve feasibility determination problems with multiple constraints. The GPFDM procedure follows the Bayesian framework, choosing an input to sample in each stage based on a VOI function, and fitting GPs to each performance measure to help decide feasibility for the whole input set. It is expected to correctly determine the feasible set

if unlimited samples can be acquired. For future research, comparison between the proposed procedure and other state-of-the-art procedures in the literature could be conducted. Numerical experiments compares three versions of the GPFDM procedure adopting different strategies of fitting GPs. For future research, comparison between the proposed procedure and other state-of-the-art procedures in the literature could be conducted.

APPENDIX I

APPENDIX FOR CHAPTER 2

1.1 *Distribution-free Variance Parameter Estimator*

This section gives an overview of the overlapping Cramér-von Mises (CvM) estimator, incorporated into our proposed method. See [58] for other types of variance estimators for Ω_Y^2 . We choose the CvM estimator because it has better statistical properties, smaller variance, and smaller bias than other estimators for the variance parameter.

The CvM estimator is formulated using the standardized time series (STS) method. After dividing in-control data $\{Y_n^0 : n = 1, 2, \dots, N\}$ into $N - m + 1$ overlapping batches that are composed of m consecutive observations as demonstrated in Figure 23(a), the standardized time series from overlapping batch i for $i = 1, 2, \dots, N - m + 1$ is defined as:

$$T_{i,m}^O(t) = \frac{\lfloor mt \rfloor (\bar{Y}_{i,m}^O - \bar{Y}_{i,\lfloor mt \rfloor}^O)}{\sqrt{m}}, \quad 0 \leq t \leq 1 \quad (22)$$

where

$$\bar{Y}_{i,j}^O = \frac{1}{j} \sum_{\ell=0}^{j-1} Y_{i+\ell} \quad \text{for } j = 1, 2, \dots, m.$$

Then from each overlapping batch $i = 1, 2, \dots, N - m + 1$, an estimator of Ω_Y^2 is

$$\begin{aligned} \text{(a) Overlapping: } & \underbrace{Y_1, Y_2, Y_3, \dots, Y_m}_{\text{batch1}}, \underbrace{Y_{m+1}, \dots}_{\text{batch2}}, \dots \\ \text{(b) Non-overlapping: } & \underbrace{Y_1, Y_2, Y_3, \dots, Y_m}_{\text{batch1}}, \underbrace{Y_{m+1}, Y_{m+2}, Y_{m+3}, \dots, Y_{2m}}_{\text{batch2}}, \dots \end{aligned}$$

Figure 23: Overlapping v.s. non-overlapping batches with size m

computed by

$$C_i^O = \frac{1}{m} \sum_{k=1}^m g\left(\frac{k}{m}\right) \left[T_{i,m}^O\left(\frac{k}{m}\right) \right]^2, \quad (23)$$

where g is a weighting function. See [58] for conditions that g should satisfy. The overlapping CvM estimator for Ω_Y^2 is finally given as the average of C_i^O 's from (23) for all overlapping batches:

$$\mathcal{C}^O = \frac{1}{N-m+1} \sum_{i=1}^{N-m+1} C_i^O. \quad (24)$$

The asymptotic bias and variance of estimator \mathcal{C}^O are shown in [58] for various weighting function g and they recommend $g(t) = -24 + 150t - 150t^2$, which makes the CvM estimator first-order unbiased.

1.2 Batch Size Selection

The determination of batch size is essential for the accurate estimation of the variance parameter, which, in turn, is critical for the control limit determination for implementing the DFMM procedure. A good batch size m should be large enough so that the standardized time series (STS) area statistics, Z_i , of non-overlapping batches are approximately independent and normally distributed [13]. Then such a batch size would ensure good statistical properties such as low bias and low variance of a CvM estimator of the variance parameter. When the amount of available training data is sufficiently large, the algorithm given in Algorithm 3 determines m . The algorithm is from [13], and it combines the von Neumann test [66] for randomness and the Shapiro-Wilk normality test [67] with a pre-specified significance level.

Input: training univariate dataset $\{Y_n^0 : n = 1, \dots, N\}$

- 1 Take an initial sample of size $n' \leftarrow 4096$ and divide into $b \leftarrow 256$ non-overlapping batches of size $m \leftarrow 16$. Set the randomness test size $\alpha_{\text{ran}} \leftarrow 0.20$, the initial normality test size $\alpha_{\text{nor}}(1) \leftarrow 0.05$, the normality test parameter $\gamma \leftarrow 0.184206$, and the normality test iteration counter $k \leftarrow 1$.

- 2 For $i = 1, \dots, b$:

$$Z_i \leftarrow \frac{1}{m^{3/2}} \sum_{j=1}^m f\left(\frac{j}{m}\right) j (\bar{Y}_{i,m}^N - \bar{Y}_{i,j}^N)$$

where $\bar{Y}_{i,j}^N \leftarrow \frac{1}{j} \sum_{\ell=1}^j Y_{m(i-1)+\ell}^0$ for $j = 1, 2, \dots, m$, and $f(t) = \sqrt{840}(3t^2 - 3t + 1/2)$ for $t \in [0, 1]$.

- 3 Apply the von Neumann test for randomness [66] to the $\{Z_i : i = 1, 2, \dots, b\}$ using the significance level α_{ran} . If the randomness test is passed, then go to step 5; otherwise, go to step 4.
- 4 Increase the batch size m and update the sample size n' :

$$m \leftarrow \lfloor \sqrt{2}m \rfloor, n' \leftarrow bm.$$

- a. If $n' > N$, then return the final batch size $m \leftarrow \lfloor N/20 \rfloor$ and stop.
 - b. Otherwise, obtain required additional observations from the training data set, and go to step 2.
- 5 Apply the Shapiro-Wilk normality test [67] to the $\{Z_i : i = 1, 2, \dots, b\}$ using the significance level $\alpha_{\text{nor}}(k) \leftarrow \alpha_{\text{nor}}(1) \exp[-\gamma(k-1)^2]$. If the normality test is passed, then return the final batch size m and stop; otherwise, go to step 6.
 - 6 Increase the normality-test iteration counter k , batch size m , and sample size n' :

$$k \leftarrow k + 1, m \leftarrow \lfloor \sqrt{2}m \rfloor, n' \leftarrow bm.$$

- a. If $n' > N$, then return the final batch size $m \leftarrow \lfloor N/20 \rfloor$ and stop.
- b. Otherwise, obtain required additional observations from the training data set, recompute Z_i 's and go to step 5.

Algorithm 3: Algorithmic statement of batch size determination [13]

The values of m used in the numerical experiments with VAR(1) and EVAR(1) processes are obtained by applying the batch size determination algorithm in Algorithm 3. Specifically, for each setting of simulated processes in Section 2.4, we use the pre-generated 1000 runs of 10,000 observations each and apply the batch size determination algorithm to each run. Table 10 reports the average values of the resulting m from the 1000 runs.

Table 10: Average values of batch size m from 1000 runs

(ϕ, ρ)	(0.3, 0.1)	(0.3, 0.7)	(0.5, 0.1)	(0.5, 0.7)	(0.7, 0.1)	(0.7, 0.7)
VAR(1)	27	24	37	31	63	70
EVAR(1)	405	437	470	485	500	500

REFERENCES

- [1] J. He, “Bayesian approach for feasibility determination and spatiotemporal scheduling,” Ph.D. dissertation, Georgia Institute of Technology, 2020.
- [2] L. C. Alwan, “Effects of autocorrelation on control chart performance,” *Communications in statistics-Theory and Methods*, vol. 21, no. 4, pp. 1025–1049, 1992.
- [3] C. M. Mastrangelo, G. C. Runger, and D. C. Montgomery, “Statistical process monitoring with principal components,” *Quality and reliability engineering international*, vol. 12, no. 3, pp. 203–210, 1996.
- [4] J. E. Jarrett and X. Pan, “The quality control chart for monitoring multivariate autocorrelated processes,” *Computational Statistics & Data Analysis*, vol. 51, no. 8, pp. 3862–3870, 2007.
- [5] K. Harris, K. Triantafyllopoulos, E. Stillman, and T. McLeay, “A multivariate control chart for autocorrelated tool wear processes,” *Quality and Reliability Engineering International*, vol. 32, no. 6, pp. 2093–2106, 2016.
- [6] A. Kalgonda and S. Kulkarni, “Multivariate quality control chart for autocorrelated processes,” *Journal of Applied Statistics*, vol. 31, no. 3, pp. 317–327, 2004.
- [7] O. Bodnar and W. Schmid, “CUSUM control schemes for multivariate time series,” in *Frontiers in Statistical Quality Control 8*, H.-J. Lenz and P.-T. Wilrich, Eds. Heidelberg: Physica-Verlag HD, 2006, pp. 55–73, ISBN: 978-3-7908-1687-7.
- [8] J. Chen, S.-H. Kim, and Y. Xie, “S 3 t: A score statistic for spatiotemporal change point detection,” *Sequential Analysis*, vol. 39, no. 4, pp. 563–592, 2020.
- [9] B. Biller and B. L. Nelson, “Modeling and generating multivariate time-series input processes using a vector autoregressive technique,” *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 13, no. 3, pp. 211–237, 2003.
- [10] M. C. Cario and B. L. Nelson, “Modeling and generating random vectors with arbitrary marginal distributions and correlation matrix,” Citeseer, Tech. Rep., 1997.
- [11] B. Biller, “Copula-based multivariate input models for stochastic simulation,” *Operations research*, vol. 57, no. 4, pp. 878–892, 2009.

- [12] S.-H. Kim, C. Alexopoulos, K.-L. Tsui, and J. R. Wilson, "A distribution-free tabular CUSUM chart for autocorrelated data," *IIE Transactions*, vol. 39, no. 3, pp. 317–330, 2007.
- [13] J. Lee, C. Alexopoulos, D. Goldsman, S.-H. Kim, K.-L. Tsui, and J. R. Wilson, "Monitoring autocorrelated processes using a distribution-free tabular CUSUM chart with automated variance estimation," *IIE Transactions*, vol. 41, no. 11, pp. 979–994, 2009.
- [14] J. Li, X. Zhang, and D. R. Jeske, "Nonparametric multivariate CUSUM control charts for location and scale changes," *Journal of Nonparametric Statistics*, vol. 25, no. 1, pp. 1–20, 2013.
- [15] W. Jiang, S. W. Han, K.-L. Tsui, and W. H. Woodall, "Spatiotemporal surveillance methods in the presence of spatial correlation," *Statistics in Medicine*, vol. 30, no. 5, pp. 569–583, 2011.
- [16] M. L. Lee, D. Goldsman, S.-H. Kim, and K.-L. Tsui, "Spatiotemporal biosurveillance with spatial clusters: Control limit approximation and impact of spatial correlation," *IIE Transactions*, vol. 46, no. 8, pp. 813–827, 2014.
- [17] M. L. Lee, D. Goldsman, and S.-H. Kim, "Robust distribution-free multivariate cusum charts for spatiotemporal biosurveillance in the presence of spatial correlation," *IIE Transactions on Healthcare Systems Engineering*, vol. 5, no. 2, pp. 74–88, 2015.
- [18] J. Chen, C. Park, S.-H. Kim, and Y. Xie, "To reduce or not to reduce: A study on spatio-temporal surveillance," *Environmental and Ecological Statistics*, vol. 26, no. 3, pp. 217–238, 2019.
- [19] P. Qiu, "Distribution-free multivariate process control based on log-linear modeling," *IIE Transactions*, vol. 40, no. 7, pp. 664–677, 2008.
- [20] L. Xue and P. Qiu, "A nonparametric CUSUM chart for monitoring multivariate serially correlated processes," *Journal of Quality Technology*, pp. 1–14, 2020.
- [21] F. M. Megahed, W. H. Woodall, and J. A. Camelio, "A review and perspective on control charting with image data," *Journal of Quality Technology*, vol. 43, no. 2, pp. 83–98, 2011.
- [22] W. H. Woodall, D. J. Spitzner, D. C. Montgomery, and S. Gupta, "Using control charts to monitor process and product quality profiles," *Journal of Quality Technology*, vol. 36, no. 3, pp. 309–320, 2004.

- [23] W. H. Woodall, "Current research on profile monitoring," *Production*, vol. 17, pp. 420–425, 2007.
- [24] T.-C. Chang and F.-F. Gan, "Monitoring linearity of measurement gauges," *Journal of Statistical Computation and Simulation*, vol. 76, no. 10, pp. 889–911, 2006.
- [25] C. Zou, F. Tsung, and Z. Wang, "Monitoring general linear profiles using multivariate exponentially weighted moving average schemes," *Technometrics*, vol. 49, no. 4, pp. 395–408, 2007.
- [26] J. Zhu and D. K. Lin, "Monitoring the slopes of linear profiles," *Quality Engineering*, vol. 22, no. 1, pp. 1–12, 2009.
- [27] R. Noorossana, A. Amiri, S. VAGHEFI, and E. Roghanian, "Monitoring process performance using linear profiles," in *Proceedings of the 3rd International Industrial Engineering Conference, Tehran, Iran*, 2004.
- [28] R. Kazemzadeh, R. Noorossana, and A. Amiri, "Monitoring polynomial profiles in quality control applications," *The International Journal of Advanced Manufacturing Technology*, vol. 42, no. 7, pp. 703–712, 2009.
- [29] J. Williams, J. B. Birch, W. H. Woodall, and N. M. Ferry, "Statistical monitoring of heteroscedastic dose-response profiles from high-throughput screening," *Journal of agricultural, biological, and environmental statistics*, vol. 12, no. 2, pp. 216–235, 2007.
- [30] J. M. Moguerza, A. Muñoz, and S. Psarakis, "Monitoring nonlinear profiles using support vector machines," in *Iberoamerican congress on pattern recognition*, Springer, 2007, pp. 574–583.
- [31] B. M. Colosimo and M. Pacella, "A comparison study of control charts for statistical monitoring of functional data," *International Journal of Production Research*, vol. 48, no. 6, pp. 1575–1601, 2010.
- [32] R. Noorossana, A. Vaghefi, and M. Dorri, "The effect of non-normality on performance of linear profile monitoring," in *2008 IEEE International Conference on Industrial Engineering and Engineering Management*, IEEE, 2008, pp. 262–266.
- [33] S. T. A. Niaki, M. Khedmati, and M. E. Soleymanian, "Statistical monitoring of autocorrelated simple linear profiles based on principal components analysis," *Communications in Statistics-Theory and Methods*, vol. 44, no. 21, pp. 4454–4475, 2015.

- [34] J. Lee, Y. Hur, S.-H. Kim, and J. R. Wilson, "Monitoring nonlinear profiles using a wavelet-based distribution-free cusum chart," *International Journal of Production Research*, vol. 50, no. 22, pp. 6574–6594, 2012.
- [35] H. Wang, S.-H. Kim, X. Huo, Y. Hur, and J. R. Wilson, "Monitoring nonlinear profiles adaptively with a wavelet-based distribution-free cusum chart," *International Journal of Production Research*, vol. 53, no. 15, pp. 4648–4667, 2015.
- [36] M. Khedmati and S. T. A. Niaki, "Phase ii monitoring of general linear profiles in the presence of between-profile autocorrelation," *Quality and Reliability Engineering International*, vol. 32, no. 2, pp. 443–452, 2016.
- [37] Y.-H. T. Wang and Y. Lai, "Monitoring of autocorrelated general linear profiles," *Journal of Statistical Computation and Simulation*, vol. 89, no. 3, pp. 519–535, 2019.
- [38] K. Wang and F. Tsung, "Using profile monitoring techniques for a data-rich environment with huge sample size," *Quality and reliability engineering international*, vol. 21, no. 7, pp. 677–688, 2005.
- [39] C.-J. Lu and D.-M. Tsai, "Automatic defect inspection for leds using singular value decomposition," *The International Journal of Advanced Manufacturing Technology*, vol. 25, no. 1, pp. 53–61, 2005.
- [40] H. Yan, K. Paynabar, and J. Shi, "Anomaly detection in images with smooth background via smooth-sparse decomposition," *Technometrics*, vol. 59, no. 1, pp. 102–114, 2017.
- [41] —, "Image-based process monitoring using low-rank tensor decomposition," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 216–227, 2014.
- [42] M. Koosha, R. Noorossana, and F. Megahed, "Statistical process monitoring via image data using wavelets," *Quality and Reliability Engineering International*, vol. 33, no. 8, pp. 2059–2073, 2017.
- [43] D. Eslami, H. Izadbakhsh, O. Ahmadi, and M. Zarinbal, "Spatial-nonparametric regression: An approach for monitoring image data," *Communications in Statistics-Theory and Methods*, pp. 1–24, 2021.
- [44] F. M. Megahed, L. J. Wells, J. A. Camelio, and W. H. Woodall, "A spatiotemporal method for the monitoring of image data," *Quality and Reliability Engineering International*, vol. 28, no. 8, pp. 967–980, 2012.

- [45] Z. He, L. Zuo, M. Zhang, and F. M. Megahed, “An image-based multivariate generalized likelihood ratio control chart for detecting and diagnosing multiple faults in manufactured products,” *International Journal of Production Research*, vol. 54, no. 6, pp. 1771–1784, 2016.
- [46] F. Amirkhani and A. Amiri, “A novel framework for spatiotemporal monitoring and post-signal diagnosis of processes with image data,” *Quality and Reliability Engineering International*, vol. 36, no. 2, pp. 705–735, 2020.
- [47] L. Zuo, Z. He, and M. Zhang, “An ewma and region growing based control chart for monitoring image data,” *Quality Technology & Quantitative Management*, vol. 17, no. 4, pp. 470–485, 2020.
- [48] D. R. Jones, M. Schonlau, and W. J. Welch, “Efficient global optimization of expensive black-box functions,” *Journal of Global optimization*, vol. 13, no. 4, pp. 455–492, 1998.
- [49] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger, “Gaussian process optimization in the bandit setting: No regret and experimental design,” *arXiv preprint arXiv:0912.3995*, 2009.
- [50] J. Snoek, H. Larochelle, and R. P. Adams, “Practical bayesian optimization of machine learning algorithms,” *Advances in neural information processing systems*, vol. 25, 2012.
- [51] L. C. Tiao, A. Klein, M. W. Seeger, E. V. Bonilla, C. Archambeau, and F. Ramos, “Bore: Bayesian optimization by density-ratio estimation,” in *International Conference on Machine Learning*, PMLR, 2021, pp. 10 289–10 300.
- [52] Y. Xie and D. Siegmund, “Sequential multi-sensor change-point detection,” in *2013 Information Theory and Applications Workshop (ITA)*, IEEE, 2013, pp. 1–20.
- [53] J. M. Cashore, L. Kumarga, and P. I. Frazier, “Multi-step bayesian optimization for one-dimensional feasibility determination,” *arXiv preprint arXiv:1607.03195*, 2016.
- [54] J. He and S.-H. Kim, “A new reward function for bayesian feasibility determination,” in *2019 Winter Simulation Conference (WSC)*, IEEE, 2019, pp. 3480–3491.
- [55] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2005.

- [56] P. W. Glynn and D. L. Iglehart, “Large-sample theory for standardized time series: An overview,” in *Proceedings of the 17th conference on Winter simulation*, 1985, pp. 129–134.
- [57] W. Whitt, “Stochastic-process limits: An introduction to stochastic-process limits and their application to queues,” *Space*, vol. 500, pp. 391–426, 2002.
- [58] C. Alexopoulos, N. T. Argon, D. Goldsman, G. Tokol, and J. R. Wilson, “Overlapping variance estimators for simulation,” *Operations Research*, vol. 55, no. 6, pp. 1090–1103, 2007.
- [59] D. C. Montgomery, *Introduction to statistical quality control*. John Wiley & Sons, 2007.
- [60] R. B. Crosier, “Multivariate generalizations of cumulative sum quality-control schemes,” *Technometrics*, vol. 30, no. 3, pp. 291–303, 1988.
- [61] Y. Mei, “Sequential change-point detection when unknown parameters are present in the pre-change distribution,” *The Annals of Statistics*, vol. 34, no. 1, pp. 92–122, 2006.
- [62] Y.-Q. Yin, Z.-D. Bai, and P. R. Krishnaiah, “On the limit of the largest eigenvalue of the large dimensional sample covariance matrix,” *Probability theory and related fields*, vol. 78, no. 4, pp. 509–521, 1988.
- [63] W. Bryc and J. W. Silverstein, “Singular values of large non-central random matrices,” *Random Matrices: Theory and Applications*, vol. 9, no. 04, p. 2 050 012, 2020.
- [64] K. Swersky, J. Snoek, and R. P. Adams, “Multi-task bayesian optimization,” *Advances in neural information processing systems*, vol. 26, 2013.
- [65] S. Surjanovic and D. Bingham, *Virtual library of simulation experiments: Test functions and datasets*, Retrieved July 26, 2022, from <http://www.sfu.ca/~ssurjano>, 2013.
- [66] J. Von Neumann, “Distribution of the ratio of the mean square successive difference to the variance,” *The Annals of Mathematical Statistics*, vol. 12, no. 4, pp. 367–395, 1941.
- [67] S. S. Shapiro and M. B. Wilk, “An analysis of variance test for normality (complete samples),” *Biometrika*, vol. 52, no. 3/4, pp. 591–611, 1965.