Detecting Change in Complex Process Systems with Phase Space Methods

by

Paul Jacobus Botha

Thesis submitted in partial fulfilment
of the requirements for the Degree

of

MASTER OF SCIENCE IN CHEMICAL ENGINEERING

in the Department of Process Engineering
at the University of Stellenbosch

Supervised by

Prof C. Aldrich

Stellenbosch
December 2006

Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety or in part submitted it at any university for a degree.

Signature: Date:

Summary

Model predictive control has become a standard for most control strategies in modern process plants. It relies heavily on process models, which might not always be fundamentally available, but can be obtained from time series analysis. The first step in any control strategy is to identify or detect changes in the system, if present. The detection of such changes, known as dynamic changes, is the main objective of this study. In the literature a wide range of change detection methods has been developed and documented. Most of these methods assume some prior knowledge of the system, which is not the case in this study. Furthermore a large number of change detection methods based on process history data assume a linear relationship between process variables with some stochastic influence from the environment. These methods are well developed, but fail when applied to nonlinear dynamic systems, which is focused on in this study.

A large number of the methods designed for nonlinear systems make use of statistics defined in phase space, which led to the method proposed in this study. The correlation dimension is an invariant measure defined in phase space that is sensitive to dynamic change in the system. The proposed method uses the correlation dimension as test statistic with and moving window approach to detect dynamic changes in nonlinear systems.

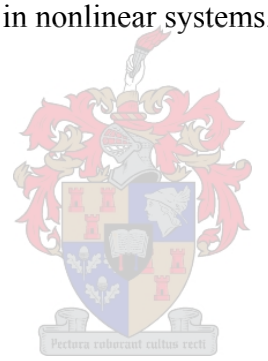
The proposed method together with two dynamic change detection methods with different approaches was applied to simulated time series data. The first method considered was a change-point algorithm that is based on singular spectrum analysis. The second method applied to the data was mutual cross prediction, which utilises the prediction error from a multilayer perceptron network. After the proposed method was applied to the data the three methods' performance were evaluated.

Time series data were obtained from simulating three systems with mathematical equations and observing one real process, the electrochemical noise produced by a corroding system. The three simulated systems considered in this study are the Belousov-Zhabotinsky reaction, an autocatalytic process and a predatory-prey model. The time series obtained from observing a single variable was considered as the only information available from the systems. Before the change detection methods were applied to the time series data the phase spaces of the systems were reconstructed with time delay embedding.

All three the methods were able to do identify the change in dynamics of the time series data. The change-point detection algorithm did however produce a haphazard

behaviour of its detection statistic, which led to multiple false alarms being encountered. This behaviour was probably due to the distribution of the time series data not being normal. The haphazard behaviour reduces the ability of the method to detect changes, which is aggravated by the presence of chaos and instrumental or measurement noise. Mutual cross prediction is a very successful method of detecting dynamic changes and is quite robust against measurement noise. It did however require the training of a multilayer perceptron network and additional calculations that were time consuming. The proposed algorithm using the correlation dimension as test statistic with a moving window approach is very useful in detecting dynamic changes. It produced the best results on the systems considered in this study with quick and reliable detection of dynamic changes, even in then presence of some instrumental noise.

The proposed method with the correlation dimension as test statistic was the only method applied to the real time series data. Here the method was successful in distinguishing between two different corrosion phenomena. The proposed method with the correlation dimension as test statistic appears to be a promising approach to the detection of dynamic change in nonlinear systems.



Opsomming

Model voorspellende beheer het ’n standaard geword in moderne prosesaanlegte vir meeste beheerstrategieë. Die strategie steun swaar op die prosesmodelle, wat nie altyd fundamenteel beskikbaar is nie, maar afgelei kan word van tydreeksdata. Die eerste stap in enige beheerstrategie is om ’n verandering in die dinamika van ’n sisteem te identifiseer, indien teenwoordig. Die identifisering van sulke veranderinge, bekend as dinamies veranderinge, is die hoofdoel van hierdie studie. In die literatuur is daar volop metodes wat verandering in sisteme identifiseer, maar meeste van hierdie metodes neem aan dat daar vooraf kennis van die sisteem bestaan, wat nie die geval is in hierdie studie nie. Meeste van die metodes wat wel op prosesdata gebaseer is, neem aan dat daar ’n lineêre verwantskap tussen die prosesveranderlikes is, met ’n stochastiese invloed vanaf die omgewing op die sisteem. Hierdie metodes werk egter nie wanneer dit op nielineêre sisteme toegepas word nie, wat die hooffokus is van hierdie studie.

‘n Groot aantal metodes ontwikkel vir nielineêre stelsels, gebruik van statistieke wat ‘n faseruimte gedefinieer word. Dit het gelei tot die ontwikkeling van die metode wat in hierdie studie voorgestel word. Die korrelasiedimensie is ’n onveranderlike kwantiteit wat in ‘n faseruimte gedefinieer word en sy onveranderlikheid word beïnvloed wanneer daar ’n verandering in die sisteem plaas vind.

In hierdie studie word die voorgestelde metode en twee soortgelyke metodes gebruik om veranderinge in tydreeksdata, van gesimuleerde sisteme te identifiseer. Die een metode wat saam met die voorgestelde metode getoets word is ’n veranderingspuntopsporingalgoritme wat gebaseer is op singuliere spektrumanalise. Die ander een is ’n gesamentlike kruisvoorspellingalgoritme, wat gebruik maak van ’n multilaagperseptron- neurale netwerk. Die vermoë van die drie metodes om veranderinge op te spoor word met mekaar vergelyk.

Die tydreeksdata is verkry vanaf simulasies met wiskundige vergelykings vir drie sisteme, sowel as regte tydreeksdata wat vanaf ’n laboratorium eksperiment verkry is. Die gesimuleerde sisteme wat ondersoek is, is die Belousov-Zhabotinsky reaksie, ’n autokatalitiese sisteem en ’n jagter-prooisisteem. Die regte tydreeksdata wat ondersoek is, is elektrochemiese geraas wat gegenereer is in ’n korrosiesisteem. Die gemete waarnemings van ’n enkele veranderlike word beskou as die tydreeksdata en die enigste informasie wat beskikbaar is oor die sisteem. Om die interne dinamika van die sisteme te beskryf word die tydreeksdata in faseruimte ingebed.

Al drie die metodes het die veranderinge in die dinamika van die sisteme geïdentifiseer deur die tydreekse te analiseer. Die veranderingspuntopsporing algoritme het ’n variërende gedrag van sy opsporingstatistiek getoon. Dit het veroorsaak dat die algoritme baie vals alarms genereer waar geen verandering plaas gevind het nie. Hierdie variërende gedrag van die opsporingstatistiek is as gevolg van die tydreeksdata wat nie normaal verspreid is nie. Die variërende gedrag word vererger wanneer die sisteem chaotiese gedrag toon of meetgeraas by die tydreeksdata gevoeg word. Die gesamentlike kruisvoorspelling algoritme is ’n baie suksesvolle metode om veranderinge in die dinamika van ’n sisteem te identifiseer, al is daar geraas teenwoordig in die tydreeksdata. ’n Nadeel van die metode is die vereiste van addisionele berekeninge en die opleiding van ’n multilaagperseptron wat tydrowend kan wees. Die voorgestelde algoritme wat gebruik maak van die korrelasiedimensie as toetsstatistiek en ’n bewegende vensterbenadering het die beste vermoë getoon om verandering in die dinamika van die sisteme te identifiseer. Die voorgestelde algoritme kan dinamiese verandering betroubaar en vinnig identifiseer in die teenwoordigheid van meetgeraas.

Die voorgestelde algoritme is die enigste algoritme wat op die elektrochemiese geraas toegepas is. Die voorgestelde algoritme het verskillende korrosieverskynsels geïdentifiseer uit die tydreeksdata. Die voorgestelde algoritme met die korrelasiedimensie as toetsstatistiek lyk na ’n belowende benadering om verandering in die dinamika van ’n sisteem te identifiseer.

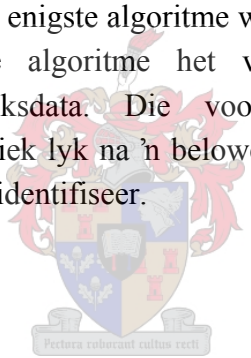
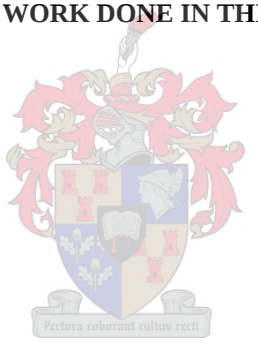


Table of contents



List of symbols

Symbol	Description
a	Constant
b	Constant
$C_N(\varepsilon)$	Correlation function
$C_q(\varepsilon)$	Correlation integral
C_T	Autocorrelation function
$D_{n,I,p,q}$	Sum of squared distances (change point detection algorithm)
$\tilde{D}_{n,I,p,q}$	Normalised sum of squared distances (change point detection algorithm)
d	Dimension
d_e	Embedding dimension
d_q	Generalised dimension
d_0	Box-counting dimension
d_1	Information dimension
d_2	Correlation dimension
H_q	Block entropies
$\overline{h}$	Threshold value
h_q	Order-q entropy
I	Set of nonzero indices
$\bar{I}$	Set of nonzero indices
I_T	Average mutual information
i	Index variable
j	Index variable
K	Integer quantity
k	True dimension
L	Hyperplane
$\overline{L}$	Horizontal extent of an attractor
l	Integer quantity
M	Lag parameter
m	Window width
MSE	Mean square prediction error
N	Integer quantity
n	Index variable
$\hat{n}$	Index variable
P	Eigenvectors
p	Start location of test sample
Q	Test sample length
$\overline{Q}$	Measured quantity
q	Integer quantity

r	Constant
S	Lag-covariance matrix
S^2	Variance of population
$S(\Delta n)$	Stretching factor
S^l	Segment of length l
s	Observed scalar value
T	Time delay or lag
t	Index variable corresponding to time
U	Orthonormal eigenvectors
V	Eigenvectors
W	CUSUM-type statistic
$\overline{X}$	Trajectory matrix
X	Observed variable
Xs	Prediction value of X
x	Vector
$\overline{x}$	Mean of population
Y	Observed variable
$\overline{Y}$	Reconstructed trajectory matrix
Ys	Predicted value of Y
y	Vector
$\hat{y}$	Predicted value
Z	Vector
z	Time series corresponding to a signal

Greek symbol	**Description**
β	Point in embedded space
$\overline{\beta}$	Coefficient
δ	Distance between two points
ε	Side length hypercubes
$\overline{\varepsilon}_t$	Time series corresponding to noise
Φ	Flow on an open subset
γ	Test statistic
η	Uncorrelated Gaussian increments (noise)
κ	Constant
λ	Eigenvalues
$\overline{\lambda}$	Lyapunov exponent
μ	Normalised sum of squared distances (change point detection algorithm)
σ^2	Standard deviation of population
Λ	Eigenvalues

Acronym	Description
ARMA	Autoregressive moving average
CUSUM	Cumulative sum
FNN	False nearest neighbours
MLP	Multilayer perceptron
MPC	Model predictive control
MSPC	Multivariate statistical process control
NHN	Number of hidden nodes
PCA	Principal component analysis
PCs	Principal components
SIC	Schwartz Information Criterion
SPC	Statistical process control
SSA	Singular spectrum analysis
SVD	Singular value decomposition

Acknowledgements

1. Prof. Chris Aldrich, my study leader.
2. Gorden Jemwa, for always helping me to understand the theory.
3. JP Barnard and Prof C. Aldrich for the Quickident Toolbox which was used for most of the phase space calculations.
4. Juliana Steyl, for all the administrative work.
5. My parents, for presenting me with the opportunity to conduct this study.
6. The NRF, for financial support.
7. My fellow postgrad students.

1. Introduction

Model predictive control (MPC) has become a standard control solution for most modern process plants (Sotomayor & Odloak, 2005). MPC is a class of computer controlled algorithms that utilises an explicit process model to predict the future response of the plant. The controller outputs are calculated so as to minimize the difference between the predictive process response and the desire response and to minimize deviation from set points. At each sampling instant, the control calculations are repeated and the predictions updated based on current measurements (Perry & Green, 1997).

The importance of the process model is obvious and they are generally obtained either from a fundamental understanding of the system or from process history data. Fundamental models require extensive research and due to the complexity (nonlinearity) of most systems such models are rarely obtainable. Thus the importance of time series analysis is evident in obtaining models from process history data. A better understanding of the physical phenomena producing a particular time series is usually obtained from time series analysis, which enables the prediction of future behaviour of the process.

The first step in any control strategy is fault detection, which is the main goal of this study. A fault can generally be described as a change in an observed variable or process parameter from an acceptable range associated with the process. In the context of this study these faults are also referred to as dynamic changes or non-stationarities and these terms will be used interchangeably throughout the thesis. Generally there are three classes of failures or malfunctions that may result in a fault (Venkatasubramanian et al., 2003a):

- **Gross parameter changes:** Parameter changes arise when there is a disturbance entering the process from the environment through one or more independent variables. Examples of such parameter changes are the change in concentration of a reactant into a reactor or the change in a heat transfer coefficient due to fouling of a heat exchanger.

- **Structural changes:** Structural changes are caused by failures in equipment and change the process itself. They result in a change in the information flow between various variables. A stuck valve or broken or leaking pipe is an example of such a change.

- **Malfunctioning sensors:** Malfunctioning sensors my result in the observed state of a variables being different from the actual state of the variable in the system. Sensor faults make the plant partially unobservable, while actuator faults make the plant partially uncontrollable. This may lead to processes to operate far from optimal conditions or can cause saturation of manipulated valves (Sotomayor & Odloak, 2005).

1.1. Process fault detection and diagnostic algorithms

Venkatasubramanian et al. (2003a) classify fault detection and diagnostic algorithms based on the priori knowledge used. Now the priori knowledge needed for fault detection and diagnostics are a set of relationships between observations and failures. This may be an explicitly defined lookup table or inferred from some source of domain knowledge. The priori domain knowledge may either be developed from a fundamental understanding of the process, or gained from past experienced with the process. This is referred to as model-based or process-history based priori knowledge respectively.

Model-based priori knowledge can broadly be divided into qualitative and quantitative models. These models are developed from a fundamental understanding of the system. Qualitative models are derived from first principles and expressed as qualitative functions centred around different units in the process. Quantitative models are expressed as mathematical relationships between inputs and outputs of the system. Work by Sotomayor and Odloak (2005) and Bloch et al. (1995) focuses on such quantitative model-based fault diagnosis methods for malfunctioning sensors.

Process history based priori knowledge on the other hand assume no knowledge of the model before hand but only that a large amount of historical process data are available. There are also a number of different approaches proposed in the literature based on historical process data. Pranatyasto and Qin (2001) proposed a method for sensor fault diagnosis based on principal component analysis. Yu et al. (1999) make use a radial basis function neural network in their approach to sensor fault diagnosis.

The schematic diagram in Figure 1.1 classifies different fault detection and diagnostic algorithms based on priori knowledge of the process.

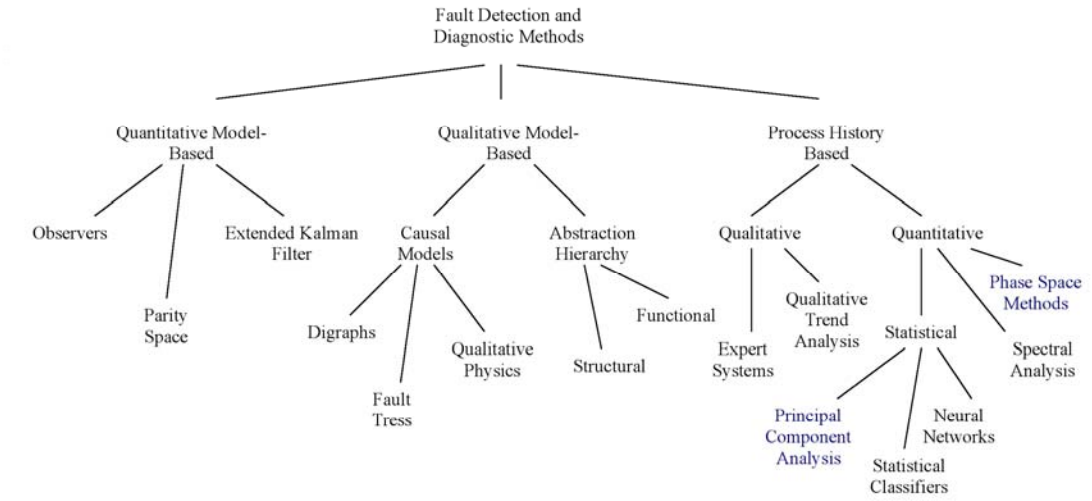


Figure 1.1: Classification of fault detection and diagnostic algorithms[1].

In order to select a desired fault detection and diagnostic algorithm, different approaches are compared against some common characteristics to benchmark the various methods. Some of the desirable characteristics looked for in fault detection and diagnostic algorithms are (Venkatasubramanian et al., 2003a):

Quick detection and diagnosis: An algorithm should be quick to detect and diagnose faults in process systems. There are however conflicting goals in quick detection of faults and tolerable amount of false alarms. A system designed for quick detection is usually very sensitive to noise and can lead to disruptions in normal operations.

Isolability: The ability of the detection and diagnostic system to distinguish between different faults.

Robustness: The detection and diagnostic system should be robust to various noise and uncertainties. The performance should degrade gradually and not abruptly with an increase in noise and uncertainties.

Novelty identifiability: In the case of abnormal events the detection and diagnostic system should be able to identify these events as novel or unknown.

Classification error estimate: A useful practical requirement is to project confidence levels with decisions made by the system.

Adaptability: Processes may change due to external inputs, structural changes or production quantities and the detection and diagnostic system should be adaptable to such changes.

[1] Inspired by Venkatasubramanian et al. (2003a)

Explanation facility: The ability of the detection and diagnostic system to provide explanations on the origin of the fault.

Modelling requirements: Modelling should be kept to a minimum to ensure fast and easy deployment of detection and diagnostic systems.

Storage and computational requirements: A reasonable balance between computational and storage requirements would usually be preferred.

Multiple fault identifiability: Identifying multiple faults is an important but difficult requirement for detection and diagnostic systems to ensure the right decisions are made by the system.

1.2. Detection of change in nonlinear dynamic systems in the literature

Recent interest in nonlinear systems has lead to great progress being made in the development of methods for change detection. In the literature a number of different methods are presented depending on the approach taken. In this section some of these methods are identified and discussed briefly.

Nonlinear models are rarely available for nonlinear systems and one approach taken by Bhagwat et al. (2003) is a multi-linear model-based fault detection scheme. It involves decomposition of nonlinear transient systems into multiple linear modelling regimes. Kalman filters and open-loop observers are used for state estimation and residual generation based on the resulting linear models. Analysis of residuals using thresholds, fault lags and logic charts enables detection of faults. Another approach by Sobajic et al. (2003) in the absence of a fundamental model is the use of neural networks, which are capable of modelling highly nonlinear systems. These models are then used to detect changes in the dynamic process conditions.

A model-free fault detection has been proposed by Fenu and Parisini (1999). An entirely different way of applying kernel regression are used which makes it possible for the kernel smoother to detect abrupt changes in nonlinear system.

Principal component analysis is widely applied in multi-variate statistical process control but results in substantial information loss when applied to nonlinear systems. This has lead to the development several nonlinear principal components analysis methods that improve data extraction when nonlinear correlations among variables exist. Maulud et al. (2006) proposed a multi-scale principal component strategy that utilises the optimal wavelet decomposition to simplify the overall structure.

The monitoring of financial markets has also shown strong nonlinearity and historical and sequential CUSUM change-point tests are proposed by Andreou and Ghysels (2005). Another change-point detection algorithm proposed by Moskvina and Zhigljavsky (2000) defines a CUSUM-type statistic. The algorithm is based on singular spectrum analysis to obtain a subspace and calculating the detection statistic, which is the Euclidean distance between the subspace and phase space vectors.

The field of medicine also encounters nonlinear systems frequently and recurrence quantification analysis, developed by Zbilut and Webber (1992)[2] to quantify the recurrent plot, has been used to detect onset of muscle fatigue, predict the occurrence of cardiac arrhythmia and identify putatively different physiological states (Mario et al., 2003). Recurrence quantification analysis is used by Mario et al. (2003) to detect small deterministic changes in the electroencephalogram of rabbits. Casdagli (1997) describes the use of recurrence plots to detect small change in the driving force of a deterministic system.

The field of electrochemical noise analysis is domination by spectral analysis in the literature. These are also process history based methods and there two very common approaches taken to distinguish between different corrosion phenomena. The first is by using power density plots obtained from the fast Fourier transform or maximum entropy method (Anita et al., 2006; Park & Kwon, 2005; Greisiger & Schauer, 2000). The second is energy distribution plots obtained from wavelet analysis techniques (Cai et al., 2005; Cao et al., 2006; Lui et al., 2006; Zang et al., 2005; Gomez-Duran & Macdonald, 2006; Greisiger & Schauer, 2000). Greisiger and Schauer (2000) also apply nonlinear statistics such fractal dimensions and Lyapunov exponents to electrochemical noise data.

A novel approach taken by Schreiber (1997) with mutual cross prediction is based on the similarity between parts of the time series themselves, rather than similarity of parameters derived from the time series by local averaging. In this approach the cross-prediction error is evaluated, which is the predictability of one segment using another as database.

The dynamics of a nonlinear system is characterised by its attractor in phase space. A Method by Hively et al. (1999) construct discrete density distributions of phase space points on the attractor and measure the dissimilarity between density distributions via χ^2 statistics. Epureanu and Yin (2004) also uses probability density functions of sampled attractors for structural health monitoring, based on the identification of changes in the vibration characteristics due to changes in material and stiffness properties of structures. Two methods by Yu et al. (1999) and Kennel (1997) quantifies non-stationarity in terms of the nearest neighbours in phase space.

[2] Cited in Mario et al. (2003).

By far the most common approach to detect dynamic changes in nonlinear systems in the literature is the used of the following three nonlinear statistics: Lyapunov exponents, correlation dimension and various entropies. Lyapunov exponents are used by Phillips (2005) to distinguish between new steady states of geomorphic systems. A method by Übeyli and Güler (2004) uses Lyapunov exponents as inputs to a multilayer perceptron neural network to detect changes in the chaotic electrocardiogram of patients with partial epilepsy.

Continuous multiresolution entropy proposed by Torres et al. (2001) computes the entropy evolution by means of sliding windows at each scale of the continuous wavelet transform of the given signal. The continuous multiresolution entropy is sensitive to changes in the dynamic complexity of attractors and is used in varies methods (Torres et al., 2006; Rosso & Mairal, 2002; Torres & Gamero, 2000; Torres et al., 2003; Añino et al., 2003; Wu & Chen, 1999) to detect change in nonlinear dynamic systems.

The correlation dimension as test for changes in dynamic nonlinear and chaotic systems is found extensively throughout the literature due to its ease of calculation. The correlation dimension has been used for machine health monitoring (Graig et al., 2000), condition monitoring of robot joints (Trendafilova & Van Brussel, 2001), vibration fault diagnosis of rolling element bearings (Logan & Mathew, 1996), monitoring and surveillance of nuclear power plants (Montesino et al., 2003) and structural health monitoring (Nichols & Virgin, 2003). A method by Manuca and Savit (1996) uses the correlation integral, which is the first step in correlation dimension estimates, itself to test for non-stationarity in processes.

The popularity in the literature and ease of calculation of the correlation dimension has lead to the development of a method, with the correlation dimension as test statistic, for detecting change in nonlinear systems proposed in this study.

1.3. Objectives, scope and approach

The main objective of the study is to detect changes in complex process systems. That is to detect dynamic changes in the deterministic structure of the system producing the signal (time series). These may be instantaneous or relatively slow dynamic changes with respect to the sampling period.

Linear stochastic systems are a well developed and documented field in the literature, with one of the most comprehensive studies probably by Basseville and Nikiforov (1993). This study focuses on dynamic nonlinear systems that have a dominant deterministic part with the possibility of chaotic behaviour. The data considered are the time series obtained from a monitoring a single variable of the

system. It is also assumed that no prior knowledge of the system under investigation is available. The detection algorithms considered is thus process history based.

The three techniques considered in the study are quantitative statistics (Figure 1.1). The first technique considered is a change-point detection algorithm based on singular spectrum analysis (SSA) as proposed by Moskvina and Zhigljavsky (2000). This algorithm is very similar to the principal component analysis branch in Figure 1.1. The two phase space methods considered in this study are mutual cross prediction, a method proposed by Schreiber (1997), and the correlation dimension as test statistic. Previous work at this institution by Bezuidenhout (2004) has proposed the correlation dimension as a potential test statistic to detect dynamic changes.

The approach is not only limited to simulated systems, but one experimental system is also considered in the study. The three different methods specified for dynamic change detection are applied to the signals from the simulated systems. Different levels of instrumental noise are also added to the simulated data, since real data are never without the presence of noise. From the results obtained by the analysis the performance of the different methods are evaluated and compared on their ability to detect different dynamic changes, response time required and robustness to noise.

The analysis of the experimental data is however limited to the method of correlation dimension as test statistic, since the format of the data is not suited for the other two methods.

The general hypothesis of this thesis is that change in nonlinear dynamic systems can be detected by monitoring of appropriate statistics describing the topology of the attractor of the dynamic system. This hypothesis will be examined through the following specific objectives:

- To conduct a literature survey of methods used to detect change in nonlinear dynamic systems.
- The development of a phase space methodology to detect changes in nonlinear dynamic systems.
- Comparison of the proposed algorithm with others on simulated case studies.

1.4. Thesis layout

First the basic theory behind phase space analysis is discussed in chapter 2. This includes time delay embedding, embedding by singular spectrum analysis and invariant characteristics of the dynamic process. The approaches of the different change detection methods are discussed in chapter 3. A description of the simulated

systems, to which the change detection methods are applied, is given in chapter 4. In chapter 5 the different dynamic change detection methods are applied to simulated data. The application of the correlation dimensions as test statistic to the experimental data is discussed in chapter 6. Finally the conclusions and limitations of the dynamic change detection methods, based on the results obtained, are discussed in chapter 7.

2. Nonlinear time series analysis with phase space methods

A time series is a sequence of time ordered measurements usually at equally spaced time intervals. These measurements can be obtained from a physical system or a computer simulated system. Analyzing a time series usually has two goals: Characterizing the system and predicting its future behaviour. The most interesting systems are usually those producing time series that show irregular behaviour as time progresses. Say the possible states of a system are represented by points in a finite dimensional phase space. Then the transition from one state x_{t_1} at time t_1 to its state at t_2 is governed by a deterministic rule that can be described in continuous time as a set of ordinary differential equations:

$$\frac{dx}{dt} = F(x_t) \tag{2.1}$$

or in discrete time as:

$$x_{n+1} = f(x_n) \tag{2.2}$$

A dynamic system can then be seen as any set of ordinary differential equations giving the time evolution of the state of the system from knowledge of its previous history. Thus for a purely deterministic system all future states can be predicted once its present state is fixed. In real-world systems pure determinism is rather unlikely, since all systems interact with their surroundings.

A more traditional approach to irregularity in time series is that external random influences, known a noise, may be acting on the system. The irregularity can easily be explained by the external random influences, while the structure found in the sequence is defined by linear dynamic rules. The most general linear model is the autoregressive moving average (ARMA) process (Schreiber, 1999):

$$x_n = \sum_{i=1}^{M} a_i x_{n-i} + \sum_{i=0}^{N} b_i \eta_{n-1} \tag{2.3}$$

where η_n are uncorrelated Gaussian increments. Although this linear stochastic description might seem attractive there are many situations where it fails since the linear equations can only lead to exponential growth or periodic oscillation (Schreiber, 1999).

Chaos theory indicates that there are nonlinear, chaotic systems producing very irregular data with purely deterministic equations (Kantz & Schreiber, 1997). Thus chaotic motion can be described as the evolution of the state of the system with sensitive dependence on initial conditions. Chaotic behaviour occurs very often in chemical engineering processes and even though long term prediction of such systems is impossible, there are certain invariant characteristics that can describe the system qualitatively.

2.1. Phase space reconstruction

Every state of a dynamic system can be described uniquely by a point in phase space. In order to describe the evolution of the states of the system in phase space, referred to as the trajectory or an attractor of the system, all relevant dynamic variables should be measured. This is seldom possible in practical cases where only a limited number of variables is available. Say a system is generated by k differential equations producing a flow in Euclidean space[3] R^k. When n independent quantities $\overline{Q}_1,\overline{Q}_2,...,\overline{Q}_n$ can be measured simultaneously, each point in phase space is associated with a point in Euclidean space R^n. The measurement function

$$F(state) = (\overline{Q}_1,\overline{Q}_2,...,\overline{Q}_n) \tag{2.4}$$

then maps R^k to R^n. If F is an embedding, it is a map that does not collapse points (one-to-one map) or tangent directions (Sauer et al., 1991). There is a number of theorems which specify the precise conditions when trajectories in the reconstructed phase space are equivalent to that in the original phase space.

However, in many cases it is only possible to observe as few as one dynamic variable of a dynamic system. The problem is going from this scalar measurement (time series) to the reconstructed phase space in which invariant quantities can be measured. Takens (1983)[4] dealt with delay coordinate maps and developed the delay embedding theorem, which makes phase space reconstruction of a single variable's time series possible.

[3] Euclidean space – A generalization of the 2- and 3-dimensional metric spaces where the generalization applies to the concepts of distance, length and angle to a coordinate system in any number of dimensions.
[4] Cited in Abarbanel (1996) and Kantz and Schreiber (1997).

2.1.1. Time delay embedding

Takens theorem as extended by Sauer et al. (1991) is formally described as:

Theorem 2.5 (Fractal delay embedding prevalence theorem): *Let Φ be a flow on an open subset $\overline{U}$ of $\Re^k$, and let A be a compact subset of $\overline{U}$ of box-counting dimension d_0. Let $d_e > 2d_0$ be an integer, and let $T > 0$. Assume that A contains at most a finite number of equilibria, no periodic orbits of Φ of period T or $2T$, at most finitely many periodic orbits of period $3T, 4T, ..., nT$, and that the linearization of those periodic orbits have distinct eigenvalues. Then for almost every smooth function h on $\overline{U}$, $h : \overline{U} \to \Re$, the delay coordinate map $F(h, \Phi, T) : \overline{U} \to \Re^{d_e}$:*

$$F(h,\Phi,T)(x) = (h(x), h(\Phi_{-T}(x)), h(\Phi_{-2T}(x)), ..., h(\Phi_{-(d_e-1)}(x))) \tag{2.5}$$

is:

1. One-to-one on A.
2. An immersion[5] on each compact subset C of a smooth manifold contained in A.

The original formulation of Takens embedding theorem required that $d_e > 2k+1$ but is replaced with $d_e > 2d_0$. Furthermore the word "generic" is replaced with "prevalent" meaning for almost all smooth functions h on $\overline{U}$ are an embedding.

Now an observed scalar signal $s(t) = h(x(t))$, given the measuring function $h : \overline{U} \to \Re$, with a sampling time t_s that results in a time series $s_n = s(nt_s)$, can be used to reconstruct the states of the system (Hegger et al., 1998):

$$y_n = (s_n, s_{n-T}, s_{n-2T}, ..., s_{n-(d_e-1)T}) \tag{2.6}$$

where T and d_e are the time delay and embedding dimension respectively and $n = 1, ..., N$. The reconstruction is also illustrated in Figure 2.1. The phase space vectors y_n obtained from the reconstruction replace the scalar time series s_n and produce an attractor that describes the dynamic behaviour of the system under investigation. Thus invariant quantities such as fractal dimensions, Lyapunov exponents and entropies are obtained from the reconstructed phase space which are identical to those in the original phase space (Kantz & Schreiber, 1997).

[5] Immersion - A smooth map F on A is an immersion if the derivative map $DF(x)$ is one-to-one at every point x of A.

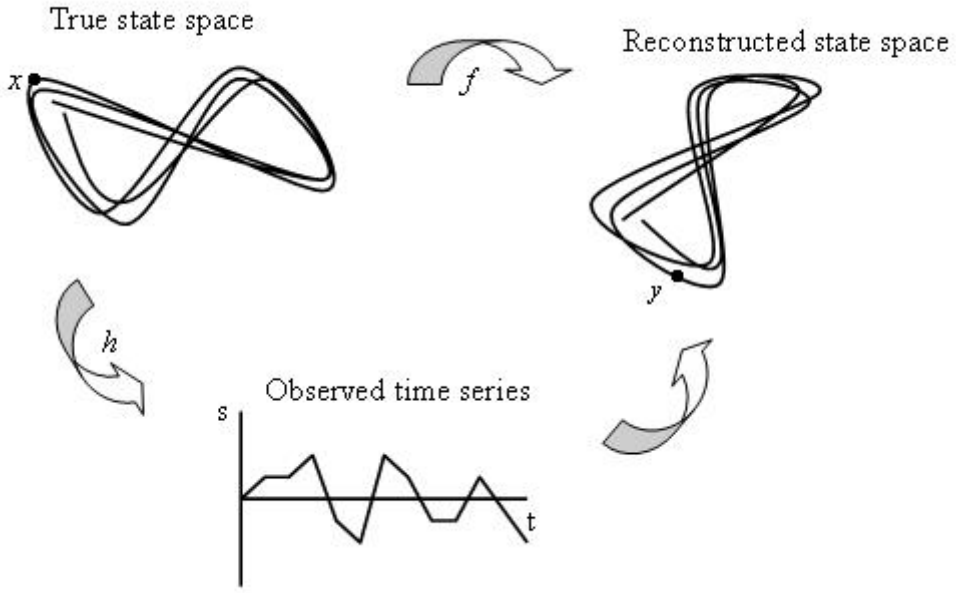


Figure 2.1: State space reconstruction by time delay embedding[6].

Making use of equation 2.6 to perform time delay embedding, requires two parameters to be specified. These two parameters are the time delay T and embedding dimension d_e. Care should be taken when selecting values for these parameters and some general techniques exist to assist in the decision-making.

(a). Choosing the optimal time delay

Embedding data with different time delays will produce reconstructions that are diffeomorphically[7] equivalent, but geometrically different. This is illustrated in Figure 2.2. A too small time delay leads to a strong correlation of successive elements of the delay vectors. This cause all the delay vectors to be clustered around the diagonal in the $\Re^{d_e}$ space, unless the embedding dimension is very large. A too large time delay will lead to a reconstruction with excessive folds in the data. This will bring states close together in the reconstruction that is not close together in the actual phase space. When noise is present in the system, as with all practical systems, the effect is just aggravated. Thus choosing a suitable time delay is a crucial when any useful information wants to be gathered from the reconstruction.

Two commonly used methods to find the optimum time delay are the linear autocorrelation function and the average mutual information statistic.

[6] Inspired by Aldrich (2002).
[7] Diffeomorphism – A smooth mapping (one-to-one) with a smooth inverse.

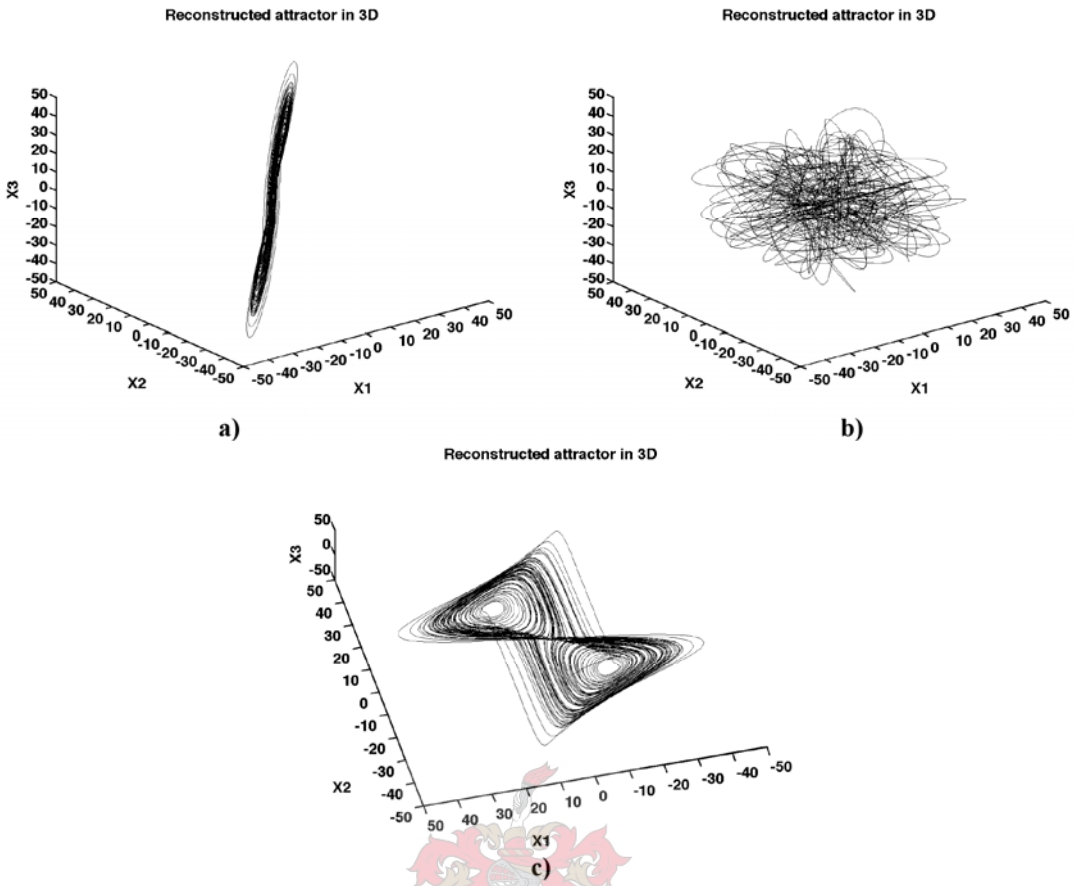


Figure 2.2: Phase space reconstructions obtained from different time delays a) too small time delay b) too large time delay c) optimal time delay.

(i). Linear autocorrelation function

The autocorrelation function is defined as (Kantz & Schreiber, 1997):

$$C_T = \frac{\frac{1}{N}\sum_{n=1}^{N}(x_n - \bar{x})(x_{n+T} - \bar{x})}{\sigma^2} \tag{2.7}$$

Where $\bar{x} = \frac{1}{N}\sum_{n=T}^{N} x_n$ and $\sigma^2 = \frac{1}{N}\sum_{n=T}^{N}(x_n - \bar{x})^2$. This function indicates the expectation of observing x_{n+T} a time T after observing x_n. The autocorrelation function decays with an increase in the time delay T. Thus at a time delay T where the autocorrelation function C_T first reach its first zero indicates that the two coordinates are linearly uncorrelated and is a good estimate to use as time delay for an embedding. This may indicate no relation on the nonlinear independence of the coordinates, but at least give a good estimation of a time delay.

(ii). Average mutual information

The average mutual information is a popular method of determining the time delay. It uses the idea of information theory to define the optimal time delay. The average mutual information is the information we already possess about a value x_{n+T} if we know x_n. It is given by (Fraser and Swinney, 1986):

$$I_T = \sum_{n=T}^{N} P(x_n, x_{n+T}) \log_2 \left[\frac{P(x_n, x_{n+T})}{P(x_n)P(x_{n+T})} \right] \tag{2.8}$$

Where $P(\cdot)$ and $P(\cdot,\cdot)$ are the individual probability and joint probability densities respectively. When two measurements are completely independent of each other, then $P(x_n, x_{n+T})$ factorizes to $P(x_n, x_{n+T}) = P(x_n)P(x_{n+T})$ which leads to equation 2.8 tending to zero. Fraser and Swinney (1986) suggested that the first minimum of I_T is used as the time delay of the embedding. At this point the coordinates are sufficiently independent of each other to spread the reconstruction, but not so independent that they have no connection with each other.

Figure 2.3 illustrates how different results are obtained for a signal analysed by the two methods. When the reconstructed attractors are studied, the time delay suggested by the autocorrelation function caused the attractor to have excessive folds. The reconstructed attractor obtained from the time delay by the average mutual information will be preferred in this case since the attractor is unfolded nicely. The two methods will not always suggest different time delay values and in such instances any one of the values may be used.

Fraser and Swinney (1986) however state that the first minimum of the average mutual information is a superior choice to that of the first zero of the autocorrelation function.

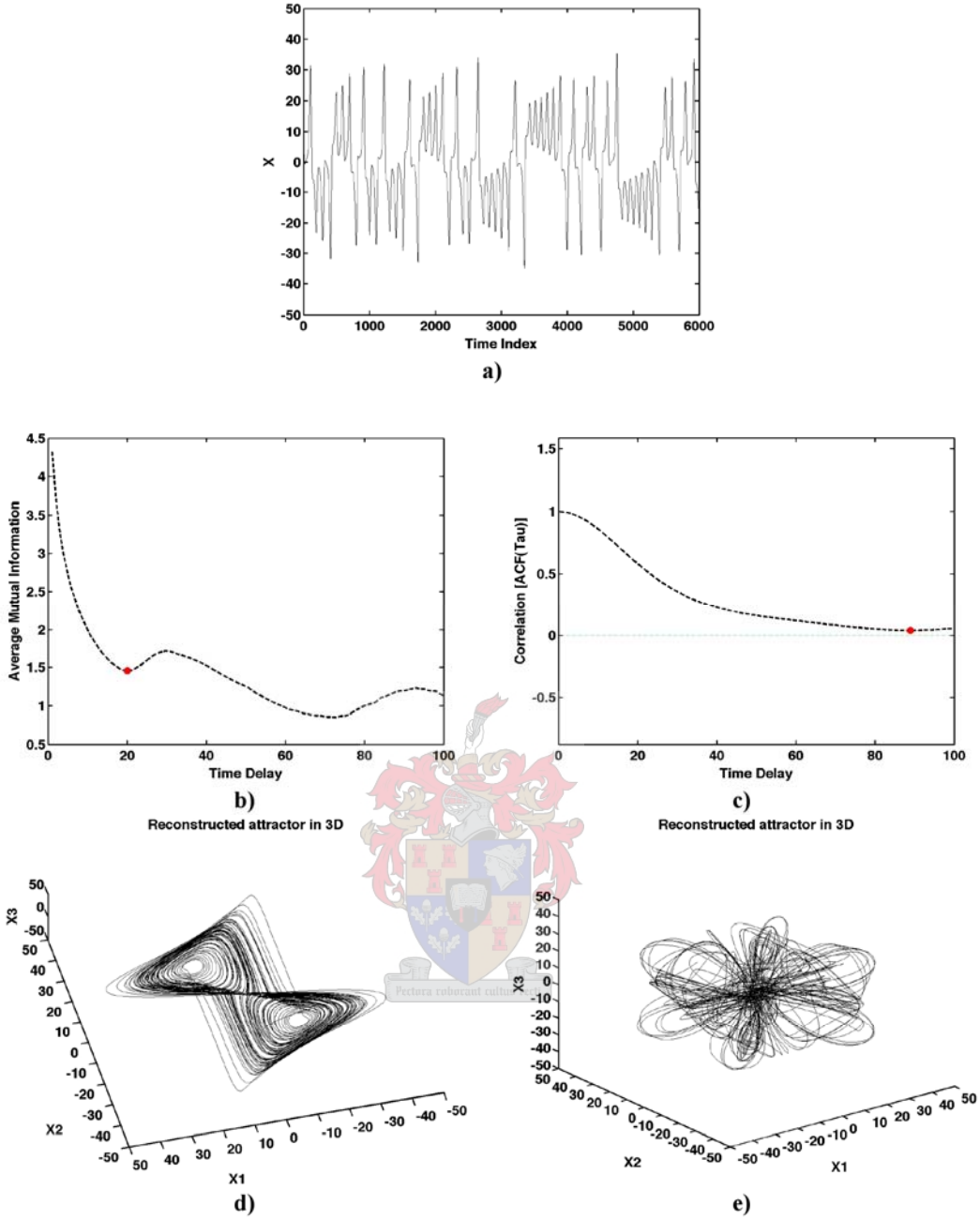


Figure 2.3: Two methods to determine the time delay and their embeddings a) time series b) average mutual information $T = 20$ c) autocorrelation function $T = 89$ d) attractor for $T = 20$ e) attractor for $T = 89$.

(b). Choosing the optimal dimension

The optimum embedding dimension is the smallest value for d_e that provides a proper reconstruction. When a too small value is selected for the embedding dimension d_e the reconstruction lacks information and leads to an improper reconstructed phase space. Just selecting an arbitrary large embedding dimension will also provide some additional practical problems. A considerable increase in

computational time are experienced for a too large embedding dimension since it increases exponentially with an increase in embedding dimension.

Although Takens (1983) originally states that an embedding dimension $d_e > 2k+1$ is required for the proper unfolding of an attractor Sauer et al. (1991) showed that $d_e > 2d$ will be sufficient. Since the true dimension k and box-counting dimension d is rarely available Kennel et al. (1992) developed the algorithm of false nearest neighbours (FNN) to determine the minimum embedding dimension. In contrast to the theorems the algorithm of FNN might suggest embedding dimension less than $2d$ for certain systems. This is particularly useful when limited computing resources are available.

False nearest neighbours appear only when the attractor is viewed in too a small embedding dimension. The idea of the algorithm is too identify neighbours in a reconstructed state space with a dimension of d_e and if they fail to be neighbours in a reconstruction with a dimension of $d_e + 1$ they are false neighbours. The minimum embedding dimension d_e is thus where the algorithm fails to identify any false nearest neighbours.

The method of false nearest neighbours tends to fail on excessively noisy data and Kennel and Abarbanel (2002) further developed an algorithm called false nearest strands. The idea behind the method is the same as false nearest neighbours except pairs of strands are identified as neighbours. This method tends to provide corrections for time series that show a high degree of autocorrelation, over sampled data and sparsely populated regions of an attractor. These effects make the estimate of the embedding dimension d_e by the false nearest neighbours algorithm less accurate.

2.1.2. Embedding by singular spectrum analysis (SSA)

Singular spectrum analysis is an alternative approach to obtain a systems reconstructed phase space. In theory both methods, time delay embedding and SSA, give an embedding that is equivalent but with limited amount of noisy data it is not always the case. SSA does not relay as heavily on the calculation of the time delay that may be influenced considerably in the presence of a fair amount of noise. This makes the embedding by SSA a very attractive method since noise is usually present in data obtained from practical systems.

SSA is base on performing a singular value decomposition (SVD) of the trajectory matrix obtained from the original time series. The basic SSA algorithm consists of four steps: embedding, SVD, grouping and reconstruction (Moskvina & Zhigljavsky, 2000).

Step 1: Embedding

The embedding is performed in a similar way than time delay embedding. The data is embedding with a time delay of $T=1$ and embedding dimension that is at least equal to the point of linear decorrelation of the data. This point is at fist minimum or first zero crossing of the autocorrelation function. Now let $x_1, x_2, ..., x_n$ be a time series of length N. Let M be an integer value that is at least equal to the point of linear decorrelation of the data and set $K = N - M + 1$. The M-lagged vectors can then be defined as:

$$\overline{X}_j = (x_j, x_{j+1}, ..., x_{j+M-1})^T \text{ for } j = 1,2,...,K \tag{2.9}$$

The trajectory matrix is then:

$$\overline{X} = (x_{i+j-1})_{i,j=1}^{M,K} = [\overline{X}_1, \overline{X}_2, ..., \overline{X}_K] \tag{2.10}$$

Step 2: SVD of the trajectory matrix

The singular value decomposition of the matrix $\overline{X}$ is done through obtaining the eigenvalues and eigenvectors of the lag-covariance matrix:

$$S = \overline{X}\overline{X}^T \text{ of size } M \times M \tag{2.11}$$

The eigenvalues $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_M$ of S are arranged in decreasing order with their corresponding orthonormal eigenvectors $U_1, U_2, ..., U_M$ of S. Let d be the number of nonzero eigenvalues λ_i, then the eigenvectors or principal components (PCs) of matrix $\overline{X}\overline{X}^T$ are:

$$V_i = \overline{X}^T U_i \text{ for } i = 1,2,...,d \tag{2.12}$$

The result obtained from the SVD is a reconstructed d dimensional state vector that is a projection onto the first d principal components of the trajectory matrix:

$$\overline{Y} = \overline{Y}_1 + \overline{Y}_2 + ... + \overline{Y}_d \tag{2.13}$$

where

$$\overline{Y}_i = \sqrt{\lambda_i} U_i V_i^T \text{ for } i = 1,2,...,d \tag{2.14}$$

Step 3: Grouping

The set of nonzero indices $\{1,2,...,d\}$ is divided into two groups:

$$I = \{1,2,...,l\} \text{ and } \bar{I} = \{l+1, l+2, ..., d\} \tag{2.15}$$

where the first l components describe and the signal and the lower $d-l$ components correspond to noise. Thus the result of the step provides a reconstruction of the signal:

$$\overline{Y}_I = \sum_{i \in I} \overline{Y}_i \tag{2.16}$$

Step 4: Reconstruction

When the SSA is used for noise reduction purposes this step is used to reconstruct the time series. By averaging over the diagonals $i+j = constant$ of the matrices $\overline{Y}_I$ and $\overline{Y}_{\bar{I}}$ two series z_t and $\bar{\varepsilon}_t$ are obtained respectively. Thus the SSA decomposition of the original time series into two series is:

$$x_t = z_t + \bar{\varepsilon}_t \text{ where } t = 1,2,...,N \tag{2.17}$$

The series z_t again correspond to the signal and the residual series ε_t to noise.

2.2. Invariant characteristics of the dynamics

One approach towards characterization of the dynamics of a system is based on estimating invariant characteristics of a system in phase space. By invariant is meant that these characteristics are independent of changes in the initial conditions of the orbit and are independent of the coordinate system in which the attractor is observed. Thus for a particular system there should be no change in the quantity of the measure whether it is estimated in the original or any reconstructed phase space. This makes it possible to estimate these characteristic quantities for experimental data even though the true phase space might be unknown.

Three major groups of classifiers that have emerged are the fractal dimensions, Lyapunov exponents and entropy. Fractal dimensions are characteristics of the geometric structure of the attractor. That is to relate the way the points on the attractor are distributed in the d_e-dimensional space. The Lyapunov exponents are a characteristic of how the orbits of an attractor move apart or together under the evolution of the dynamics (Abarbanel, 1996). The entropy in turn is a notion of the degree of uncertainty in being able to predict a future state of the system.

2.2.1. Fractal dimensions

Dimensions are a characteristic of the geometric structure of the attractor. The attractors of simple periodic and quasi-periodic systems have simple geometries such as set of points, closed curves and torii (Judd, 1992). Attractors of chaotic dynamic systems however have fractal geometries and are called strange attractors. In general a

strange attractor has a small scale structure that is repeated on arbitrarily small length scales. This is illustrated in Figure 2.4 and is known as self-similar sets of an attractor.

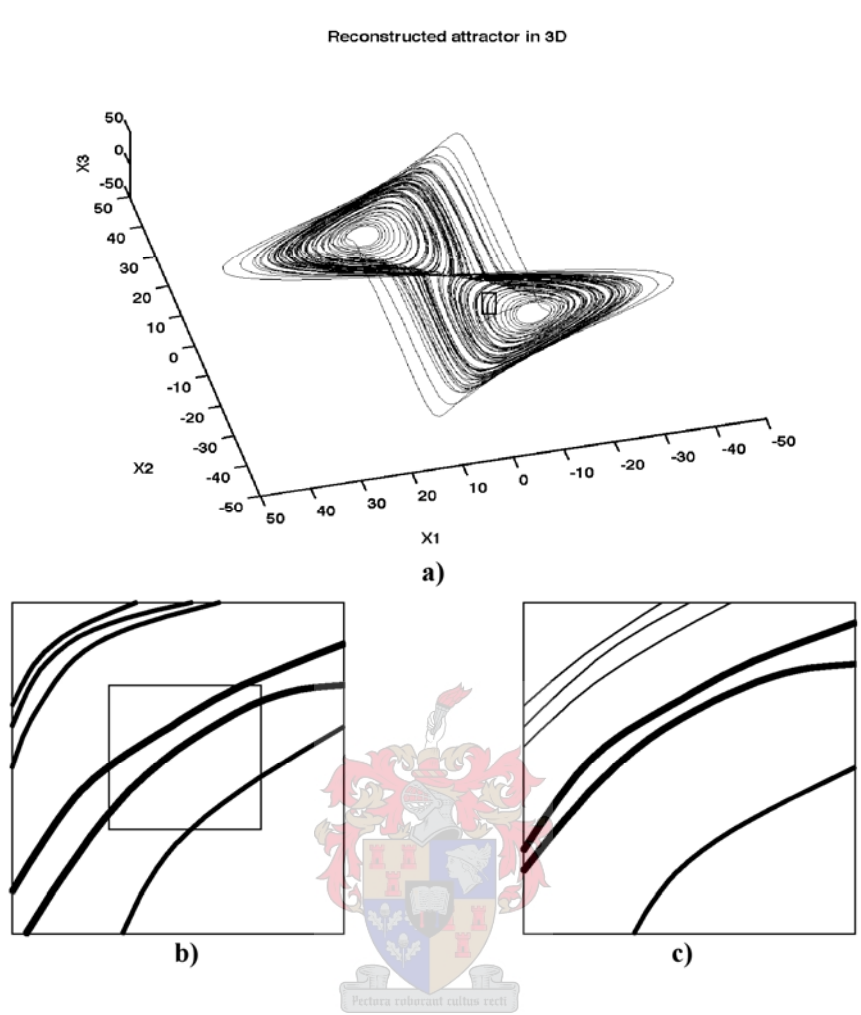


Figure 2.4: Construction of a self-similar set a) chaotic attractor b) enlargement of region defined by rectangle in (a) c) enlargement of region defined by rectangle in (b).

There are several ways to quantify the self-similarity of a geometrical object. One such way is the Hausdorff dimension, which has formed the basis for other dimensions. However, owing to computational limitations the box-counting dimension d_0 is calculated instead. The box counting dimension is closely related to the Hausdorff dimension and presents the upper bound on the Hausdorff dimension (Kantz & Schreiber, 1997).

(a). Box-counting dimension (d_0)

Consider a point set in $\Re^{d_e}$. This point set is covered with hypercubes or boxes with side length ε and call $G(\varepsilon)$ the number of boxes which contain at least one point. Then for a self-similar set (Kantz & Schreiber, 1997):

$$G(\varepsilon) \propto \varepsilon^{-d_0}, \qquad \varepsilon \to 0 \tag{2.18}$$

The box-counting dimension d_0 can then be defined as:

$$d_0 = \lim_{\varepsilon \to 0} \frac{\log_e(G(\varepsilon))}{\log_e\left(\frac{1}{\varepsilon}\right)} \tag{2.19}$$

The box-counting dimension is generally calculated and recalculated for increasing embedding dimensions. For embedding dimensions less than of equal to the true box-counting dimension the estimate of the box-counting dimension will be the same as the embedding dimension ($d_0 = d_e$). This will happen until the geometric structure of the attractor is fully unfolded. Embedding the data into a higher dimension will not cause a significant increase in the box-counting dimension. This is the true value of the box-counting dimension.

Although the box-counting dimension can be calculated easily, it is not suitable as an algorithm for extracting dimensions from experimental data (Kantz & Schreiber, 1997). The edge effects due to the finite size of the attractor are severe and not easily overcome. The box-counting dimension ignores the distribution of the points on the attractor since it measure only whether a box contain at least one point or not. When more weight needs to be given to those parts visited more frequently on the attractor, there is a family of dimensions, the generalised or Renyi dimensions (Kantz & Schreiber, 1997).

(b). Generalised Dimension (d_q)

The generalised correlation integral is defined as (Kantz & Schreiber, 1997):

$$C_q(\varepsilon) = \int_x p(x)_{\varepsilon}^{q-1} dp(x) \tag{2.20}$$

For self-similar set:

$$C_q(\varepsilon) \propto \varepsilon^{(q-1)d_q}, \ \varepsilon \to 0 \tag{2.21}$$

Now the generalised dimension d_q is defined as:

$$d_q = \lim_{\varepsilon \to 0} \frac{1}{q-1} \frac{\log_e C_q(\varepsilon)}{\log_e(\varepsilon)} \tag{2.22}$$

In the case where $q = 0$ the generalised dimension d_q is equivalent to the box-counting dimension d_0. The other two fractal dimensions most commonly examined are the information (d_1) and correlation (d_2) dimensions.

(c). Information dimension (d_1)

The information dimension can be thought of as the dimension of the "core set" which is the part of the attractor that contains most of the points. It is the average amount of information needed to specify a point x with accuracy ε. The information dimension also specifies how this amount of information scales with resolution ε. It accounts for the differences in the distribution density of the points covering the attractor. By setting $q = 1$ in the generalised dimension equation 2.22 and applying l'Hospital rule yields the information dimension as (Kantz & Schreiber, 1997):

$$d_1 = \lim_{\varepsilon \to 0} \frac{\sum_i p_i \log_e p_i}{\log_e \varepsilon} \tag{2.23}$$

Where p_i is the probability of a point being in the i^{th} partition:

$$p(x_i) = \frac{n(x_i)}{\sum_i n(x_i)} \tag{2.24}$$

The generalised dimension is a non-increasing function in q, thus the correlation dimension d_2 is calculated as the lower bound of the information dimension d_1. The reason for this is the correlation dimension is much easier to calculate for a limited amount of data.

(d). Correlation dimension (d_2)

The correlation dimension as a measure to quantify the "strangeness" of an attractor has been introduced by Grassberger and Procaccia (1983). When $q = 2$ in the generalised dimension equation 2.22 it becomes:

$$d_2 = \lim_{\varepsilon \to 0} \frac{\log_e \sum_i p_i^2}{\log_e \varepsilon} \tag{2.25}$$

The $\sum_i p_i^2$ in equation 2.25 is a two point correlation function that measures the probability of finding two random points within a certain radius ε. Grassberger and Procaccia (1983) suggested a simple algorithm to calculate the correlation dimension. They defined the correlation function to estimate the $\sum_i p_i^2$ term in equation 2.25 as:

$$C_N(\varepsilon) = \binom{N}{2}^{-1} \sum_{i \neq j} \Theta(\varepsilon - \|x_i - x_j\|) \tag{2.26}$$

Where x_i and x_j are points defining the trajectory of the attractor and $\Theta(x)$ is a Heaviside step function: $\Theta(x) = 0$ if $x \leq 0$ and $\Theta(x) = 1$ if $x > 0$. The sum basically count the number of inter point distances that is smaller than ε. In the limit of an infinite amount of data we expect $C_N(\varepsilon)$ to scale like a power law:

$$C_N(\varepsilon) \propto \varepsilon^{d_2}, \ \varepsilon \to 0 \tag{2.27}$$

Thus the correlation dimension can be defined as:

$$d_2 = \lim_{\varepsilon \to 0} \lim_{N \to \infty} \frac{\log_e C_N(\varepsilon)}{\log_e \varepsilon} \tag{2.28}$$

Figure 2.5 illustrates this probing of a hypersphere of radius ε on one of the points defining the trajectory of the attractor. The correlation sum is calculated for a number of different hypersphere radii ε_i and a graph is constructed that plots $\log_e C_N(\varepsilon)$ against $\log_e(\varepsilon)$. The slope of this graph where $\varepsilon \to 0$ should approach the correlation dimension d_2.

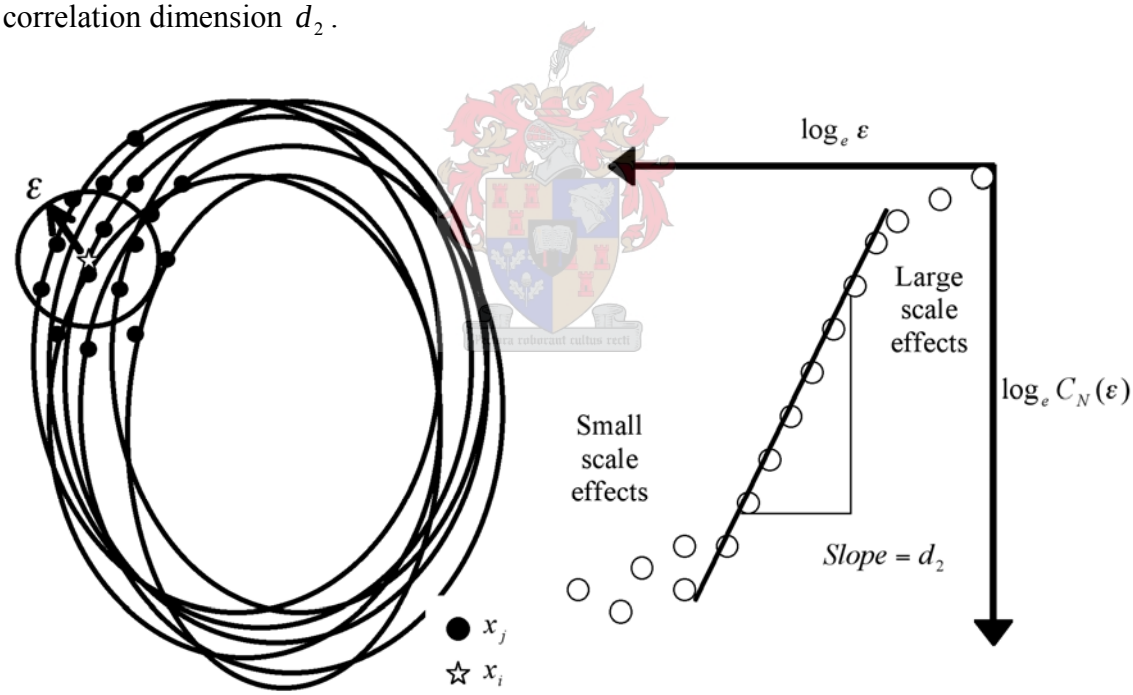


Figure 2.5: Correlation dimension calculation by Grassberger and Procaccia (1983) algorithm.

This is however not the case when limited amount of data is used. The graph will jump irregularly for small values of ε because of insufficient amount data points. This is known as small scale effects. For this reason there is looked at the intermediate, but still small values of ε where a constant slope is present. This is known as the scaling range and a reliable estimation of the correlation dimension can be drawn from this region. At large hypersphere radii ε the graph will flatten. This is

due to the finite size of the attractor and known as the large scale effects. This is all illustrated in Figure 2.5.

This algorithm wasn't however without flaws. Lai and Lerner (1998) showed that the scaling region is sensitive to the choice of embedding time delay. Judd (1994) also indicated that linear correlation in the data set misleads the algorithm to wrongly show convergence to some low dimension, which could then be misinterpreted for inherently low-dimensional dynamics. Judd further points out the deficiencies of the Grassberger and Procaccia (1983) algorithm (Judd, 1992):

1. The first problem is inherent in all methods that are based on calculations of inter point distances. From a sample trajectory of length n there are $n(n-1)/2$ interpoint distances, but these are not all independent. The triangle inequality states that the distance between two points is no more than the sum of the distances from a third point.

2. The smoothness of the correlation function is misleading because it contains a lot of statistically correlated information. The correlation function $C_N(\varepsilon)$ which is the number of interpoint distances less than ε becomes more statistically correlate as ε is increased. Thus some less weight must be given to large ε values.

3. Examples have been given where the scaling region reflects only large scale properties of an attractor and do not reflect information about the dimension of the attractor. Taking a scaling region also completely ignores information about small distances between points, which is still information about scaling. There is no reason for throwing away information if the data is not corrupted noise.

4. The algorithm also gives no estimate of the error in of the dimension estimate. This is because the error in fitting a straight line through the scaling region is not an estimate of the error in the estimation of the dimension, but rather an error in the straight line fit.

Judd (1992) proposes a modified algorithm on the Grassberger and Procaccia (1983) algorithm. The new algorithm replaces the linear scaling region by fitting a polynomial of the order of the topological dimension in the region. The correlation dimension is also expressed for inter point distances below a specific scale ε_0 as illustrated in Figure 2.6. Now not only a single value of the correlation dimension is compared but rather the clustering of correlation dimension estimation curves calculated by Judd's (1992) algorithm. This allows the examination of the micro- and

macroscale of the reconstructed dynamic attractor with the correlation dimension. Judd's (1992) proposed algorithm for the correlation dimension is valid for $\varepsilon < \varepsilon_0$:

$$C_N(\varepsilon) \propto \varepsilon^{d_2} q(\varepsilon),\ \varepsilon \to 0 \tag{2.29}$$

Where $q(\cdot)$ is a polynomial of order of the topological dimension. Judd (1992) also further proposes a form of confidence levels for the correlation dimension with his new algorithm. He also states that the algorithm is accurate for correlation dimension estimates up to four ($d_2 < 4$).

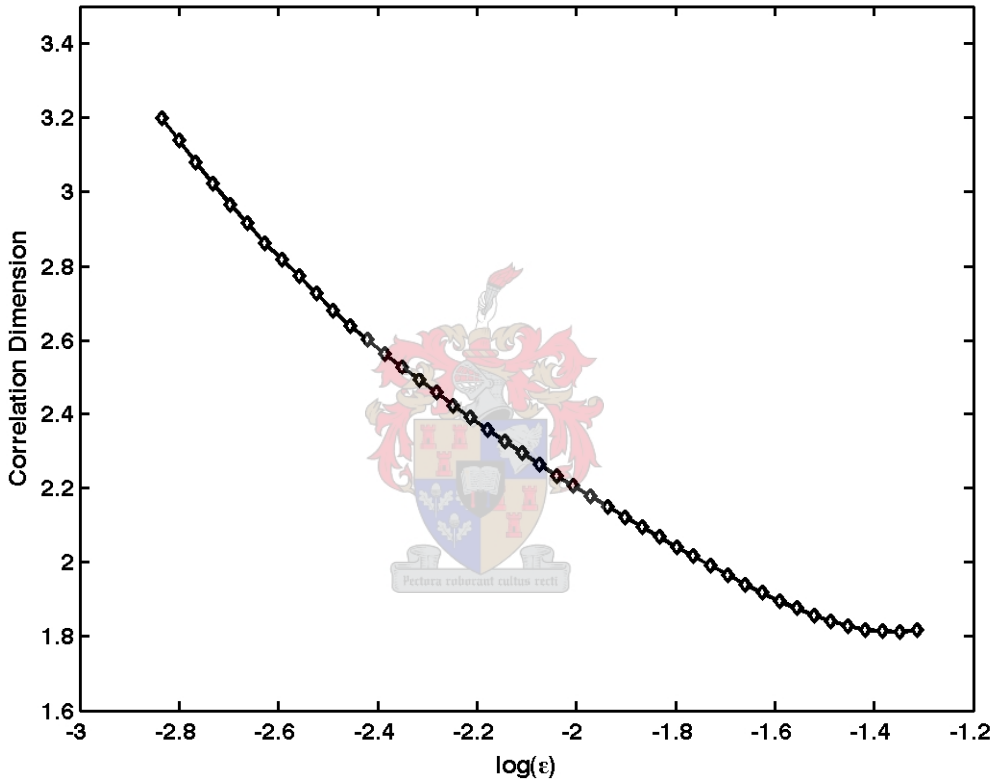


Figure 2.6: Representation of the correlation dimension estimate by Judd's (1992) algorithm.

2.2.2. Entropy

Entropy is a concept fundamental to thermodynamics. It is a thermodynamic quantity describing the amount of disorder in the system. This concept can be generalised to characterise the amount of information stored in more general probability distributions (Kantz & Schreiber, 1997). For a time series the entropy characterise the amount of information, on average, a single measurement provide about the state of the system. Thus the amount of information already possessed about the future states given the past observations. The entropy measures the divergence of pairs of nearby orbits.

Just as the case with the generalised or Renyi dimensions there exist a series of entropy measures, the order-q Renyi entropies. They characterise the amount of information which is needed in order to specify the value of an observation with a certain precision when the probability density function is know. What makes the entropy relevant for nonlinear time series analysis is the inverse of the entropy is the time scale relevant for the predictability of the system. It also supplies topological information about the folding process. Calculating the entropies are however difficult since the computation requires more data points than dimension estimates and Lyapunov exponents. The order-q entropies h_q are defined as (Kantz & Schreiber, 1997):

$$h_q = \lim_{d_e \to \infty} H_q(d_e + 1) - H_q(d_e) \tag{2.30}$$

With $H_q(m)$ being the block entropies of block size d_e. A critical problem with the computation is the limit of the embedding dimension $d_e \to \infty$. This is overcome by calculating $C_q(\varepsilon)$ for increasing embedding dimensions then:

$$C_q(d_e, \varepsilon) \propto \varepsilon^{d_q} e^{-H_q(d_e)} \tag{2.31}$$

For values of ε inside the scaling region of the dimension plots the factor ε^{d_q} is almost constant and the entropy h_q can be determine by plotting h_q versus ε for various m.

$$h_q(d_e, \varepsilon) = H_q(d_e + 1, \varepsilon) - H_q(d_e, \varepsilon) = \log_e \frac{C_q(d_e, \varepsilon)}{C_q(d_e + 1, \varepsilon)} \tag{2.32}$$

For a sufficiently large d_e the graph will converge towards a constant h_q as illustrated in Figure 2.7. As in the case of fractal dimension estimates the correlation dimension d_2 is the most robust and computable and the same goes for the correlation entropy h_2. The correlation entropy is the lower bound of the Kolomogorov-Sinai entropy h_1. The significance of this is that typically for a linear deterministic system the Kolomogorov-Sinai entropy h_1 will be zero, for a linear stochastic system it will be infinite and finite for a deterministic nonlinear system. A positive finite entropy is also typical of a chaotic system, as illustrated in Figure 2.7.

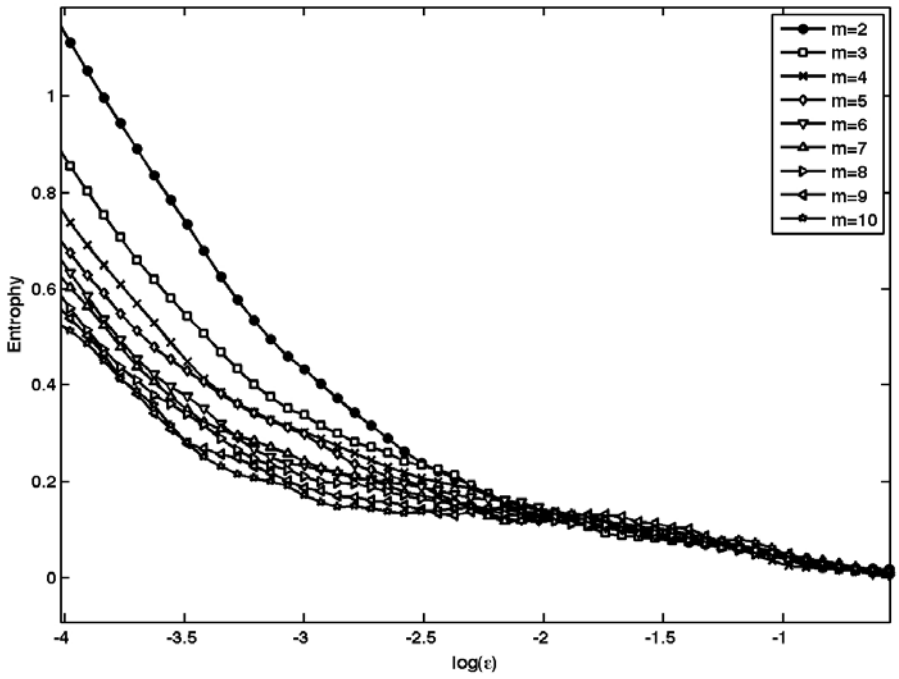


Figure 2.7: Converging of entropy h_2 as embedding dimension is increased.

2.2.3. Lyapunov exponents

The Lyapunov exponents are a measure of the divergence of the orbits or trajectories that define an attractor. This is an indication of the sensitivity to initial conditions and unpredictability of a dynamic system that characterise chaos. Predominantly periodic systems trajectories may diverge over the course of time, but the divergence will not be very dramatic. Thus we only speak of chaos when the divergence is exponentially fast. There are as many Lyapunov exponents for a dynamic system as there are phase space dimension (Kantz & Schreiber, 1997), thus defining exponential divergence in each direction of phase space. Figure 2.8 illustrates the concept of Lyapunov exponents.

The Lyapunov exponents $\overline{\lambda}_i$ are determined by following the time evolution of two initially similar points (Kantz & Schreiber, 1997). For the maximal Lyapunov exponent let a_t and b_t be two initially similar points that are separated by a distance $\delta_0 = \|a_t - b_t\|$. At a time step Δt the separation of the two point's trajectories, $a_{t+\Delta t}$ and $b_{t+\Delta t}$, are $\delta_{\Delta t} = \|a_{t+\Delta t} - b_{t+\Delta t}\|$. The divergence of the trajectories can be expressed as:

$$\delta_{\Delta t} = \delta_0 e^{\lambda.\Delta t} \tag{2.33}$$

Rearranging gives:

$$\bar{\lambda} = \frac{1}{\Delta t}\log_e\left(\frac{\delta_0}{\delta_{\Delta t}}\right) \tag{2.34}$$

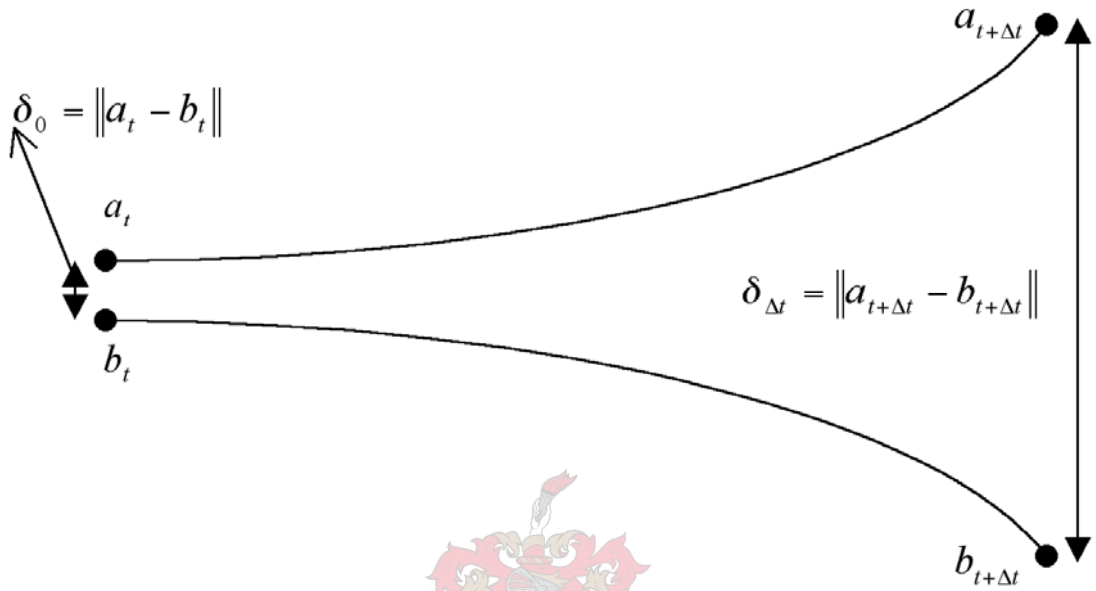


Figure 2.8: Divergence of orbits/trajectories of an attractor.

A positive finite Lyapunov exponent characterise the presence of chaos, and exponent $\bar{\lambda} \leq 0$ characterise a periodic attractor. For a random attractor (noise) the exponent will tend to infinity. Because of the finite size of an attractor two trajectories can't separate further then the size of the attractor. Thus equation 2.34 is only valid during times Δt for which $\delta_{\Delta t}$ remain small. Thus a mathematical more rigorous definition will have to involve a first limit $\delta_0 \to 0$ such that a second limit $\Delta n \to \infty$ can be performed without the trajectories separating beyond the attractor's size.

Generally experimental data are contaminated with noise and its influence can be minimised by using an appropriate averaging statistics when computing the maximal Lyapunov exponent. A method of determining the exponent by Kantz and Schreiber (1997) is to choose a point β_{n_0} of the time series in the embedding space and select all the neighbours within a distance ε. Then compute:

$$S(\Delta n) = \frac{1}{N}\sum_{n_0=1}^{N}\log_e\left(\frac{1}{\left|\bar{U}(\beta_{n_0})\right|}\sum_{\beta_n \in \bar{U}(\beta_{n_0})}\left|s_{n_0+\Delta n} - s_{n+\Delta n}\right|\right) \tag{2.35}$$

Where s_n is a point in the neighbourhood $\overline{U}(\beta_{n_0})$ with a diameter ε and Δn the time span. Since the minimum embedding dimension d_e and optimal distance ε might be unknown, $S(\Delta n)$ should be calculated for a variety of both values. The size of the neighbourhood should be as small as possible but still sufficient to have a few neighbours. If for some range of Δn the function $S(\Delta n)$ exhibits a robust linear increase, the slope is an estimate of the maximal Lyapunov exponent. If there exists no linear increase the data reflects the lack of exponential divergence of nearby trajectories (Hegger et al., 1998).

As earlier state there is a Lyapunov exponent for each dimension in state space. Determining them is unfortunately quite hard but Kantz and Schreiber (1997) proposed the following algorithm: let x_n and y_n be two nearby trajectories in a d_e dimensional phase space separated by an infinitesimal distance δ_n. The time evolution of their distance is:

$$y_{n+1} - x_{n+1} = F(y_n) - F(x_n) \tag{2.36}$$

$$y_{n+1} - x_{n+1} = J_n(y_n - x_n) + O(\|y_n - x_n\|^2) \tag{2.37}$$

Where $F(y_n)$ is expanded around x_n and $J_n = J_n(x_n)$ is the $d_e \times d_e$ Jacobian matrix of F at x. Given $\delta_n = y_n - x_n$, the modulus one step later can be computed. Let V_i and Λ_i be the eigenvector and eigenvalue of J respectively, then decompose δ_n into these vectors with coefficients $\overline{\beta}_i$:

$$\delta_{n+1} = \sum \overline{\beta}_i \Lambda_i V_i \tag{2.38}$$

Each arbitrary point of the phase space will find different eigenvectors and eigenvalues of the Jacobian since it is position dependent. The Lyapunov exponent $\overline{\lambda}_i$ is defined as the normalised logarithm of the modulus of the i^{th} eigenvalue Λ_i of the product of all Jacobians along the trajectory (in time order) in the limit of an infinitely long trajectory:

$$\overline{\lambda}_i = \lim_{N \to \infty} \frac{1}{N} \log_e \left| \Lambda_i^{(N)} \right| \tag{2.39}$$

Where Λ_i is defined by:

$$\prod_{n=1}^{N} J_n u_i^{(N)} = \Lambda_i^{(N)} u_i^{(N)} \tag{2.40}$$

This set of d_e different Lyapunov exponents is called the Lyapunov spectrum. This method was first suggested by Sano and Sawada (1985) and Eckmann and Ruelle (1985) independently. The method is later on applied to a hydrodynamic experiment by Eckmann et al. (1986) and compared to that of Sano and Sawada (1985). Sano and Sawada (1985) states that the minimum amount of points required is $N > d_e(\overline{L}/\varepsilon)^{d_1}$ where d_e is the embedding dimension, d_1 the information dimension and $\overline{L}$ the horizontal extent of the attractor. A typical value for $\varepsilon/\overline{L}$ is 3-5%.

The Lyapunov exponents are typically arranged in decreasing order, $\overline{\lambda}_1 > \overline{\lambda}_2 > ... > \overline{\lambda}_{d_e}$, with at least $\overline{\lambda}_1 > 0$ for a chaotic system. The sum of the positive Lyapunov exponents can be used to estimate the Kolomogorov-Sinai entropy (Hegger et al., 1998).

3. Methodology: detecting dynamic change with phase space methods

Whether a time series is analysed for system identification or forecasting in MPC many practical problems usually arise when parameters describing the system are subjected to changes at unknown time instances. All the invariant statistics described in chapter 2 are also under the assumption that the time series under investigation are not subjected to such changes. In this chapter methods are discussed to identify such a parameter change, also referred to as a non-stationarity. Before the algorithms can be discussed a clearer understanding of stationarity are required.

A signal is called stationary if all transition properties from one state of the system to another are independent of time within the observation period (Kantz & Schreiber, 1997). In general there are two reasons for a measured series to be non-stationarity. The first being inadequate sampling of the system to capture the total dynamics of the system and the second being a change in the deterministic rules governing the dynamics of the system, also known as a dynamic change.

The first instance is an analytical error that can be avoided by sampling at a sufficient high sampling frequency, usually two to ten times that of the highest frequency describing the dynamics of the system, and analysing a sufficiently long time series ensuring that the total dynamics of the system is captured. To ensure that a sufficiently long time series is analysed an invariant characteristic measure is calculated for a fraction of the times series. The measure is then recalculated for a larger fraction of the time series. Under constant deterministic rules governing the dynamics of the system the invariant characteristic measure should converge once a certain fraction of the time series is reach, indicating a sufficiently long time series for analysis of the measure.

The second instance of non-stationarity is what the analysts are interested in when an actual change in the dynamics of the system occur. This can be in the form of a normally constant parameter changing instantaneously or drifting slowly over time.

There exist a number of statistical tests for stationarity in data proposed in the literature (Schreiber, 1997; Manuca & Savit, 1996; Kennel 1997; Hively et al., 1999; Yu et al., 1999). Most of the tests are based on the following idea (Schreiber, 1997):

1. The time series of the system under investigation are divided into a number of segments.

2. A test statistic is calculated for each of the segments of the time series.

3. The results of the test statistic of all the segments are compared and if there exists a significant difference, variation outside statistical fluctuations, the time series is regarded as non-stationary.

The importance of the choice of test statistic is obvious. A simple statistic such as the mean or standard deviation is not particularly good. The statistic is arbitrary and not related to any geometric properties of an attractor, which captures and characterise the dynamics of a dynamic system. Unless the particular statistic estimates a parameter deemed physically or dynamically important the power of such an arbitrary choice against various sorts of non-stationarity will vary greatly. Applying such a statistic could also overestimate the significance of the difference. Observed dynamic data are correlated and classical statistical estimations of confidence rely heavily on the independence of the observations. An example is measuring empirical means of segments of a chaotic time series and performing classical t-test for their equality will quite often reject null hypothesis of stationarity, even when data comes from noise free stationary experiments such as the Lorenz attractor (Kennel, 1997). Such methods do not identify non-stationarity reliably.

Then there are a number of statistical process control (SPC) techniques such as standard Shewhart control charts, moving average control charts and CUSUM control charts discussed comprehensively by Basseville and Nikiforov (1993). Further multivariate statistical process control (MSPC) tools are also available where principal component analysis (PCA) is combined with SPC (Wise & Gallagher, 1996). PCA is one of the favourite tools used for data compression and information extraction. It finds combinations of variables and factors that describe the major trends in a data set. Once the PCA model has been developed control limits can be placed on the process scores, sum of scores T^2 and overall residual Q for fault detection.

Similarly to MSPC tools a change-point detection algorithm by Moskvina and Zhigljavsky (2000) has been developed and implemented in a software package. This algorithm is based on using SSA to obtain a subspace in $\Re^M$ and calculating the detection statistic, the Euclidean distance between the subspace and phase space vectors.

The main problem with these techniques is that they were designed to detect changes in constant mean, linear Gaussian sequences. This presents some problems when

applied to nonlinear and chaotic systems since the geometric properties of the attractor are again not considered.

One test statistic that suffers considerably from non-stationarity in the data is the correlation dimension. The correlation dimension is used extensively throughout the literature for the detection of non-stationarity (Logan & Mathew, 1996). A simple drift of parameters usually causes an increase in dimensionality, while most other types of non-stationarities and insufficient sampling yield spuriously low estimates (Kantz & Schreiber, 1997).

A change detection algorithm using the correlation dimension as test statistic is proposed and discussed together with the change-point detection algorithm and another phase space method, nonlinear cross prediction, in this section. These algorithms are applied to data in chapter 5 that was obtained from simulated systems discussed in chapter 4. The data is in the form of a single variable's observed time series with a change being induced at some point. The observed time series will be analysis as if no knowledge of the underlying dynamics of the system is known.

3.1. Change-point detection with singular spectrum analysis

The four steps of singular-spectrum analysis are described in chapter 2. In this section only the change-point detection algorithm as implemented in the software by Moskvina and Zhigljavsky (2000) will be discussed. Now if the original time series are divided into a number of segments, with the first segment having a length of N data points. Then $K = N - M + 1$ with M being the lag parameter, which is an integer value.

In the third step of the SSA algorithm a particular combination of a certain number $l < M$ of the eigenvectors determines an l-dimensional hyperplane L_I in $\Re^M$. The distance between the vectors $X_j (j = 1,...,K)$ and the l-dimensional hyperplane L_I is controlled by the choice of I and can be reduced to a rather small value. If the next segment are selected and there is no change in the deterministic structure of the system, then the distance between the l-dimensional hyperplane L_I and $X_j, j \geq K$ should stay reasonably small. But if at a certain time τ a dynamic change occur then the distance between the l-dimensional hyperplane L_I and $X_j, j \geq K + \tau$ will be expected to increase. Figure 3.1 is an illustration of the hyperplane obtained from the first part of the time series and how the attractor moves away as a dynamic change is encountered.

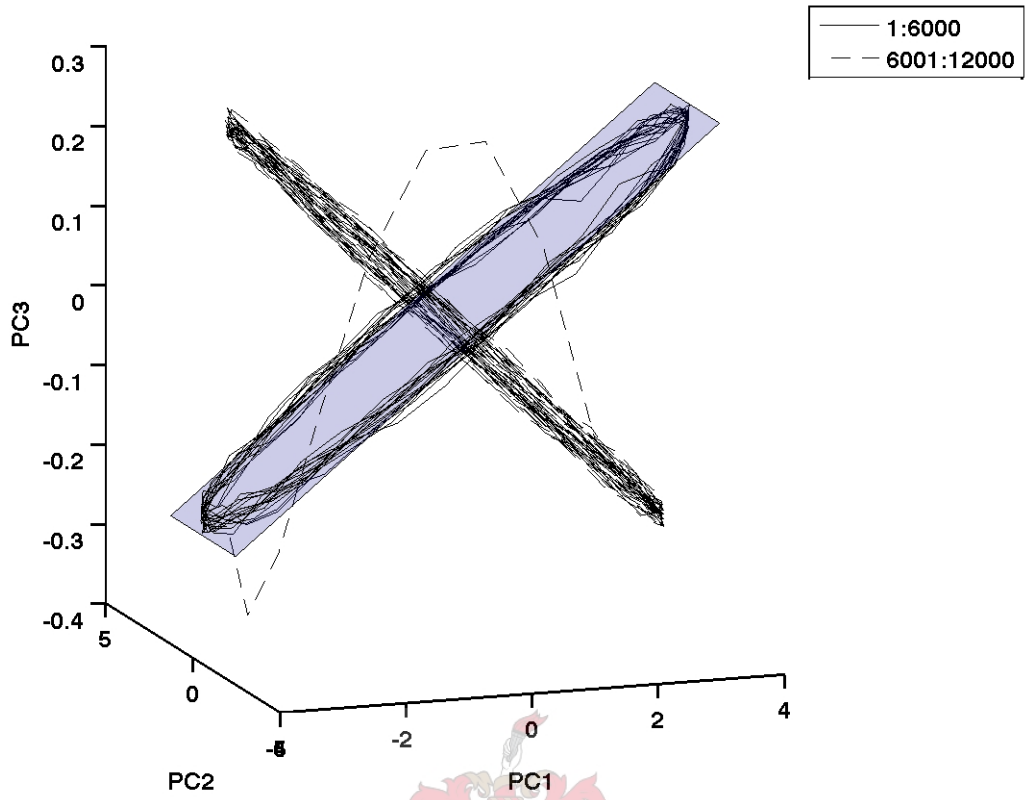


Figure 3.1: Embedded time series with l-dimensional hyperplane obtained from eigenvectors of the first 6000 data points.

In the implementation of the algorithm the singular value decomposition are applied to a sequence of time intervals $[n+1, n+m]$ where $n = 0,1,...$ is the iteration number and m is the length of the time interval the trajectory matrix is computed from. This implementation accommodates the presence to detect instantaneous as well as slow dynamic changes.

3.1.1. Description of the algorithm

The algorithm proposed by Moskvina and Zhigljavsky (2000) as implemented in the software package is described as follow:

Let $x_1, x_2, ..., x_N$ be a time series with $N \leq \infty$ and choose two integers: an even integer $m, (m < N)$ the window width and the lag parameter M. Also set $K = N - M + 1$. Now for each $n = 0,1,...,N-m$ construct the trajectory matrices over the time intervals $[n+1, n+m]$:

$$\overline{X}^{(n)} = (x_{n+i+j-1})_{i,j=1}^{M,K} = \begin{pmatrix} x_{n+1} & x_{n+2} & \dots & x_{n+K} \\ x_{n+2} & x_{n+3} & \dots & x_{n+K+1} \\ \dots & \dots & \dots & \dots \\ x_{n+M} & x_{n+M+1} & \dots & x_{n+m} \end{pmatrix} \tag{3.1}$$

The columns of $\overline{X}^{(n)}$ are the vectors $\overline{X}_j^{(n)}(j=1,...,K)$, where $\overline{X}_j^{(n)}=(x_{n+j},...,x_{n+M+j-1})^T$ for $j \geq -n+1$. The lag-covariance matrix is defined as $R_n=\overline{X}^{(n)}(\overline{X}^{(n)})^T$ for each n. The SVD of R_n gives a collection of M eigenvectors, and a particular combination I of $l<M$ of them determines an l-dimensional subspace $L_{n,I}$ of the M-dimensional space $\Re^M$ of vectors $\overline{X}_j^{(n)}$. Now $P_{i_1},...,P_{i_l}$ are the l eigenvectors that determines the subspace $L_{n,I}$ and the sum of squares of the distance between the vectors $\overline{X}_j^{(n)}=(j=p+1,...,q)$, the test matrix, and the subspace $L_{n,I}$ are denoted by $D_{n,I,p,q}$.

Since the eigenvectors are orthonormal, the squared Euclidean distance between an M-vector Z and the subspace $L_{n,I}$ spanned by the l eigenvectors $P_{i_1},...,P_{i_l}$ is:

$$\|Z\|^2-\left\|P^TZ\right\|^2=Z^TZ-Z^TPP^TZ \tag{3.2}$$

where $\|\bullet\|$ is the usual Euclidean norm and P is the $(M\times l)$-matrix with columns $P_{i_1},...,P_{i_l}$. It is also the difference between the squared norms of the vector Z and the projection of Z to the space $L_{n,I}$. The squared distance $D_{n,I,p,q}$ is the sum of these differences for the vectors $\overline{X}_j^{(n)}$ constituting the test matrix. Therefore:

$$D_{n,I,p,q}=\sum_{j=p+1}^{q}\left((\overline{X}_j^{(n)})^T\overline{X}_j^{(n)}-(\overline{X}_j^{(n)})^TPP^T\overline{X}_j^{(n)}\right) \tag{3.3}$$

When a dynamic change occurs at a certain point τ it is expected that the vectors $\overline{X}_j=\overline{X}_{j-n}^{(n)}$ with $j>\tau$ lie further away from the l-dimensional subspace $L_{n,I}$ than the vectors $\overline{X}_j$ with $j\leq\tau$. Thus when n changes, the sequence $D_{n,I,p,q}$ starts growing somewhere around $\hat{n}$ such that $\hat{n}+q+M-1=\tau$. This means that $\hat{n}=\tau-q-M+1$ is the first value of n such that the test sample $x_{n+p+1},...,x_{n+q+M-1}$ contains a point with a change. The sequence $D_{n,I,p,q}$ will continue to grow for some time depending on the signal and the relations between p, q and N. The sequence $D_{n,I,p,q}$ will reach a maximum and will decrease to its original level or perhaps a new level depending on the signal after the change-point (Moskvina & Zhigljavsky, 2000).

In order to obtain a decision rule the normalised sum of squared distances is defined as follow:

$$\tilde{D}_{n,I,p,q}=\frac{1}{M(q-p)}D_{n,I,p,q} \tag{3.4}$$

The ratio $\overline{S}_n$ is then defined as:

$$\overline{S}_n = \frac{\tilde{D}_{n,I,p,q}}{\mu_{n,I}} \tag{3.5}$$

where $\mu_{n,I}$ is an estimate of the normalised sum of squared distances $\tilde{D}_{j,l,p,q}$ at the time intervals $[j+1, j+m]$ where the hypothesis of no change can be accepted. Here the largest value of $m \le n$ is selected so that the hypothesis of no change is accepted. A CUSUM-type statistic is the defined as:

$$W_1 = \overline{S}_1,\ W_{n+1} = \left(W_n + \overline{S}_{n+1} - \overline{S}_n - \frac{\kappa}{\sqrt{M(q-p)}} \right)^+, \text{ for } n \ge 1 \tag{3.6}$$

where $(a)^+ = \max\{0, a\}$ for any $a \in \Re$ and $\kappa = \frac{1}{3\sqrt{M(q-p)}}$. The algorithm announces a structural change of the time series once the statistic $W_n > \overline{h}$ with the threshold $\overline{h}$ defined as:

$$\overline{h} = \frac{2t_\alpha}{M(q-p)} \sqrt{\frac{1}{3}(q-p)(3M(q-p) - (q-p)^2 + 1)} \tag{3.7}$$

with t_α the $(1-\alpha)$-quantile of the standard normal distribution.

Figure 3.2 and Figure 3.3 illustrate the results obtained from a numerical example analysed with the software implementing the algorithm.

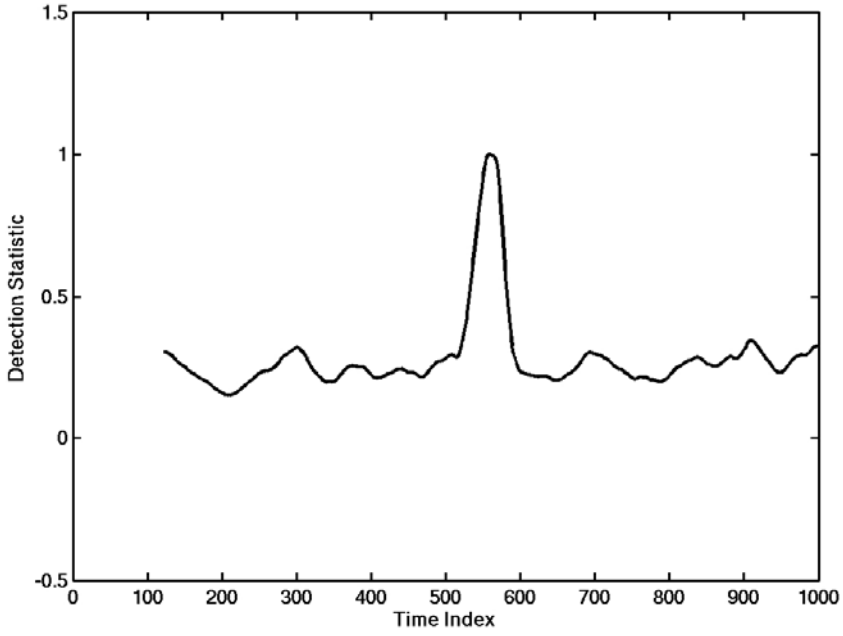


Figure 3.2: The detection statistic $\tilde{D}_{n,I,p,q}$ obtained for a dynamic change.

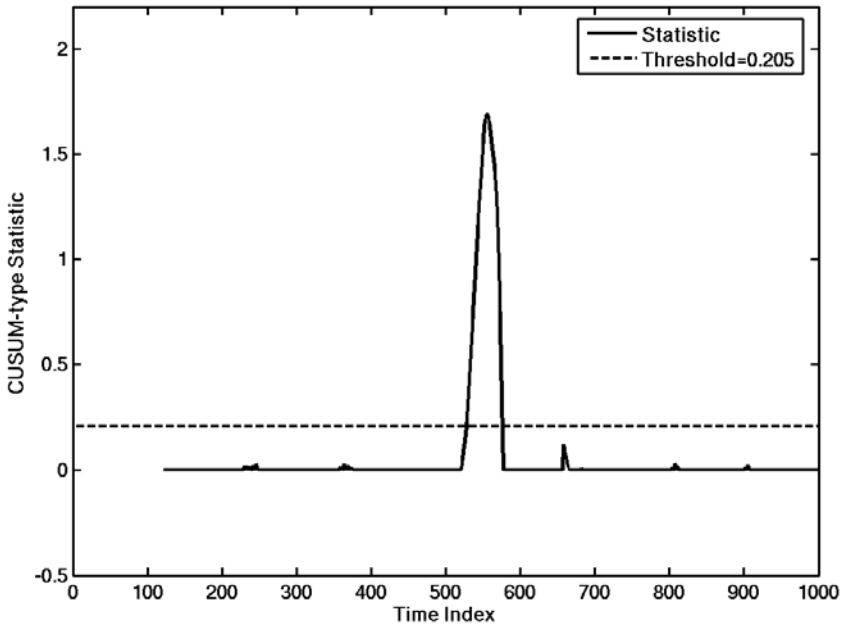


Figure 3.3: The CUSUM statistic W_n obtained for a dynamic change.

3.1.2. Choice of parameters

The first parameter that has to be specified is the window width m. This parameter specifies the size of the base sample, which determines the l-dimensional subspace $L_{n,I}$. The size of m depends on the kind of dynamic changes that should be detected. A general rule is that m should be reasonably large. However choosing m too large might smooth out all changes in the time series. Then m should also be smaller than τ and all other distances between sequential change-points. Choosing m to small cause a haphazard behaviour of the moving squared distance $D_{n,I,p,q}$, which may lead to a high frequency of false alarms because outliers might be recognized as dynamic changes.

The next two parameters are the length Q and location p of the test sample. A general recommendation is to choose $p \geq K$, this ensures that the columns of the base and test matrices are different. Numerical simulations have shown that $Q = q - p = 1$ is often very reasonable and even optimal (Moskvina & Zhigljavsky, 2000). To obtain a smoother behaviour of the test statistic $D_{n,I,p,q}$ the length Q of the test sample should be increased so that $Q > 1$. Choosing Q too large might again cause the test statistic $D_{n,I,p,q}$ to become too smooth for any dynamic changes to be observed.

3.2. Mutual cross prediction to detect change

Mutual cross prediction is a test for non-stationarity proposed by Schreiber (1997). The test is based on the similarity between different parts of the time series itself, rather than the similarity of parameters derived from the time series. In this approach the cross-prediction error, which is the predictability of one segment using another segment as database, is evaluated.

Let $x_1, x_2, ..., x_n$ be a time series that is divided into r continues segments of length l where the ith segment is denoted by S_i^l. Now traditionally a statistic γ_i is calculated for each segment and compared with other segments or the whole time series. Should a statistical significant difference exists the series is classified as non-stationary. Now Schreiber (1997) defines a statistic on pairs of segments, $\gamma_{ij} = (S_i^l, S_j^l)$, in particular the cross-prediction error. By using the statistic γ_{ij} on pairs of segments the number of parameters calculated increase from $r = n/l$ to $r^2 = (n/l)^2$. Although the information gained are largely redundant for the purpose of statistical testing since the γ_{ij} for different i, j are not expected to be independent, different and more hidden kinds of non-stationarity can be detected (Schreiber, 1997). In general a more detailed picture about the nature of the changes are created to locate the segments where non-stationarity exist. The main purpose of the test is to compare the information captured in the paired statistic $\gamma_{ij} = (S_i^l, S_j^l)$ relative to the diagonal terms $\gamma_{ii} = (S_i^l, S_i^l)$.

In principle $\gamma_{ij} = (S_i^l, S_j^l)$ can be any quantity which is sensitive to differences in the dynamics in S_i^l and S_j^l respectively. In this study the statistic used is the mean square prediction error obtained by fitting a multilayer perceptron (MLP) neural network model to S_i^l and predicting S_j^l one step into the future with the model. Refer to appendix A for a description of the models used in this study.

Now the time series $s_1, s_2, ..., s_n$ is divided into r continues segments of length l so that the i^{th} segment is denoted by S_i^l. Select two segments so that $S_i^l = (x_1, ..., x_n)$ and $S_j^l = (y_1, ..., y_n)$. Now choose two integers, an embedding dimension d_e and time delay T, so that embedding vectors $X_t = (x_t, x_{t-T}, x_{t-2T}, ..., x_{t-(d_e-1)T})$ and $Y_t = (y_t, y_{t-T}, y_{t-2T}, ..., y_{t-(d_e-1)T})$ in the same d_e dimensional phase space are formed.

The model f, a MLP neural network, is trained to predict x_i one step into the future, so that $x_{t+1} = f(X_t)$. The model f, which was built on the segment $S_i^l = (x_1, ..., x_n)$, is now used to predict y_i one step into the future, thus $\hat{y}_{t+1} = f(Y_t)$. The test statistic $\gamma_{ij} = (S_i^l, S_j^l)$, the mean square prediction error, can now be evaluated by:

$$\gamma_{ij} = (S_i^l, S_j^l) = MSE = \frac{1}{n - T(d_e - 1) + 1} \sum_{t=T(d_e-1)+1}^{n} (y_t - \hat{y}_t)^2 \tag{3.8}$$

When diagonal segments are selected, the model is simulated with the data it was trained on and the test statistic $\gamma_{ii} = (S_i^l, S_i^l)$ should be expected to produce its lowest values. This is illustrated in the results obtained from an example in Figure 3.4 and Figure 3.5.

From these figures it is also clear that a change in dynamics exists in segment 6, since the model build on segments 1 to 5 are able to predict the first 5 segments quite well, but fail to do so there after. Again when the model are build on segments 7 to 11 it predicts the last 5 segments quite well but fail to do so with the first 5 segments. Thus the dynamics of the system has changed in the 6th segment and stayed constant there after.

Now a decision rule is created to evaluate whether a prediction errors differs significantly to be associated with a dynamic change. A 95% confidence limit on the MSE is created for the stationary parts of the time series by assuming a normal distribution of the MSE and using the following equation:

$$MSE_{95\% \lim it} = \overline{MSE} + t_{df,\alpha} S \tag{3.9}$$

Where $\overline{MSE}$ is the mean of the MSE, S^2 is the variance of MSE and $t_{df,\alpha}$ is the value from the t-table that corresponds to a right-hand tail area of α for a t-distribution with df degrees of freedom. For example in Figure 3.4 a 95% confidence limit is created on segments 1 to 5. Should the prediction MSE of a segment by a model from the stationary part of the time series be higher than the confidence limit, a change in the dynamics of the system is identified for that segment. The same holds for the prediction MSE of a segment from the stationary part of the time series. Should the prediction MSE be higher than that of the confidence limit the segment from which the model was built, is identified as having different dynamics as that from the stationary part.

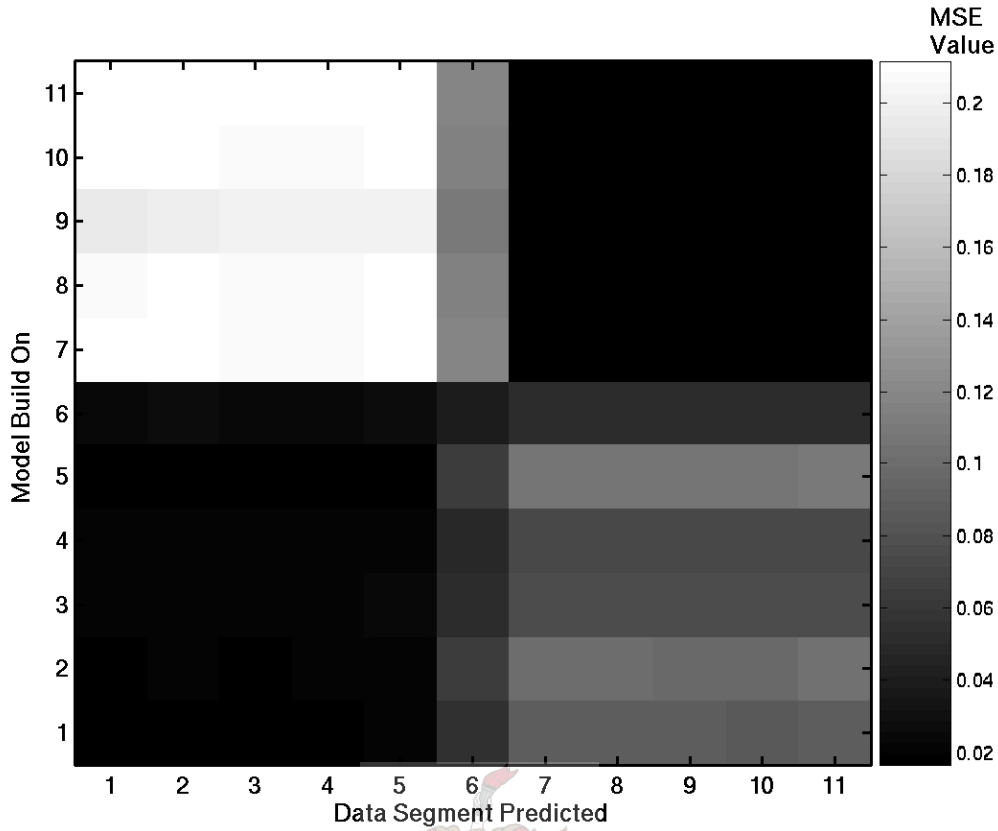


Figure 3.4: Mutual cross prediction errors expressed in 2 dimensions.

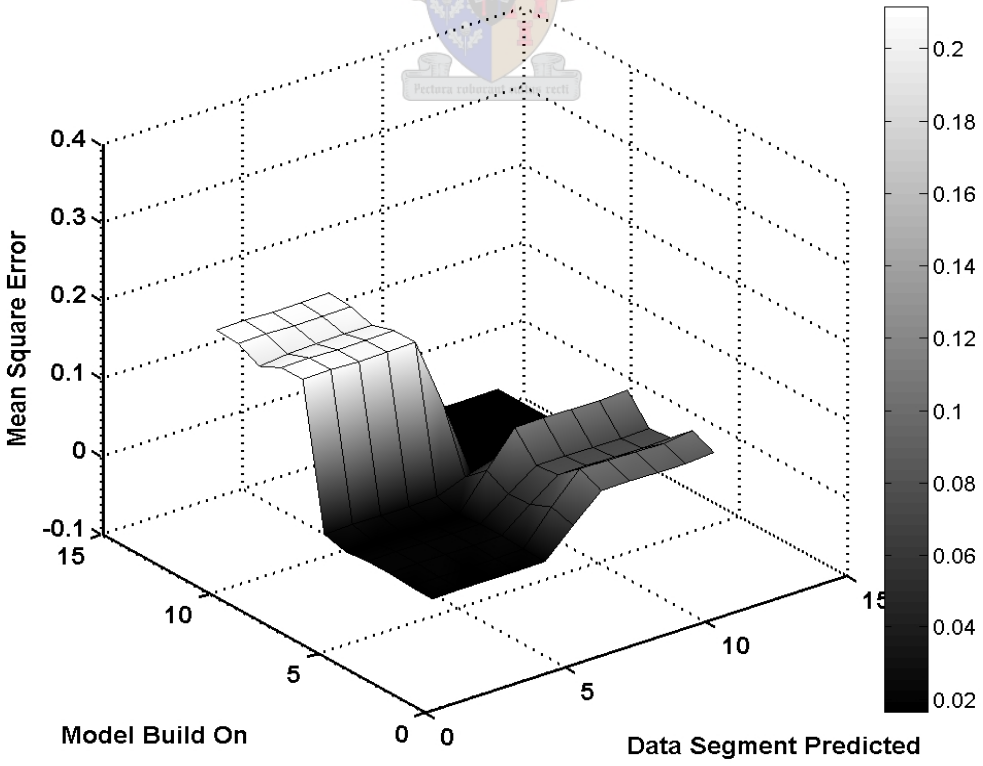


Figure 3.5: Mutual cross prediction errors expressed in 3 dimensions.

3.3. Correlation dimension as test statistic to detect change

The presence of non-stationarities in a time series will have an influence on the geometrical structure of a system's attractor. As discussed in chapter 2 the correlation dimension is an invariant measure of the topological structure of an attractor and should lose its invariant property once a non-stationarity is encountered in a time series. The correlation dimension can be estimated by a number of methods, but in this study the Judd (1992) algorithm is used. What makes the Judd (1992) algorithm so attractive is unlike the Grassberger and Procaccia (1983) algorithm, which produces a binary value for the correlation dimension, the Judd (1992) algorithm express the correlation dimension over a range of inter-point distances. This should enable the test to detect dynamic changes over a range of scales, which will not be averaged out as in the case of a single binary value.

Now there are several ways in which the test statistics can be applied to an observed time series. An obvious method will be to divide the time series into a number of segments and estimate the test statistic for each of these segments. Should the test statistic estimations differ significantly from each other the series will be regarded as non-stationary, indicating that a change has occurred.

However, there are some limitations to this approach. The estimation of the correlation dimension requires sufficiently long data sets, which will lead to few estimations when only a short time series is available. For example: there are only enough data available to divide the time series into just three segments and not one of the estimations of the different segments give the same result, as illustrated in Figure 3.6. The only information obtained from this result is that some change occurred in the second segment, but it is still unclear whether another change occurred in the third segment. It is also still unclear whether the dynamic change was in the form of an instantaneous change or a slow drift. Real time monitoring will also have a slow response time since long time interval will be required to collect sufficient data for the next estimation.

An improved approach will be to use fixed size segments that are overlapping, a moving window approach. This eliminates the problem of short time series that can't be divided into many segments. This approach will be able to determine when the change is induced and whether it is an instantaneous change or slow drift. This approach also presents the possibility of real time monitoring, since only a small number of new observations are required. The new data are combined with previous data to produce a sufficiently long data set for estimation of the test statistic. This approach is illustrated in Figure 3.7.

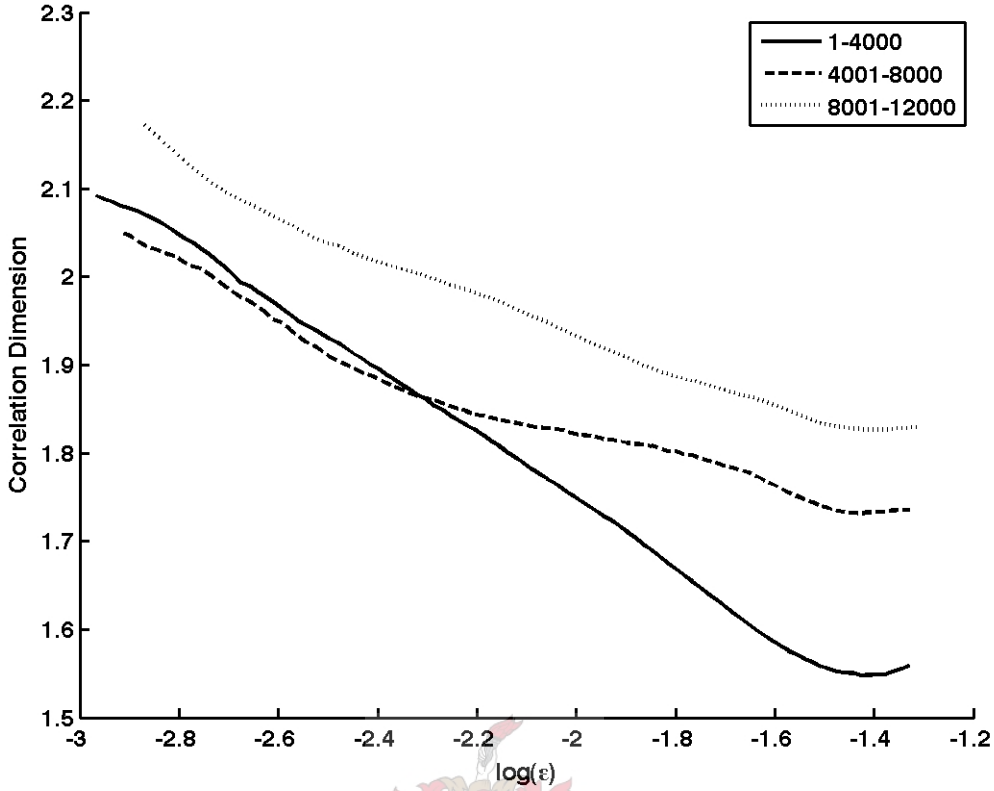


Figure 3.6: Correlation dimension estimated for three parts of a time series.

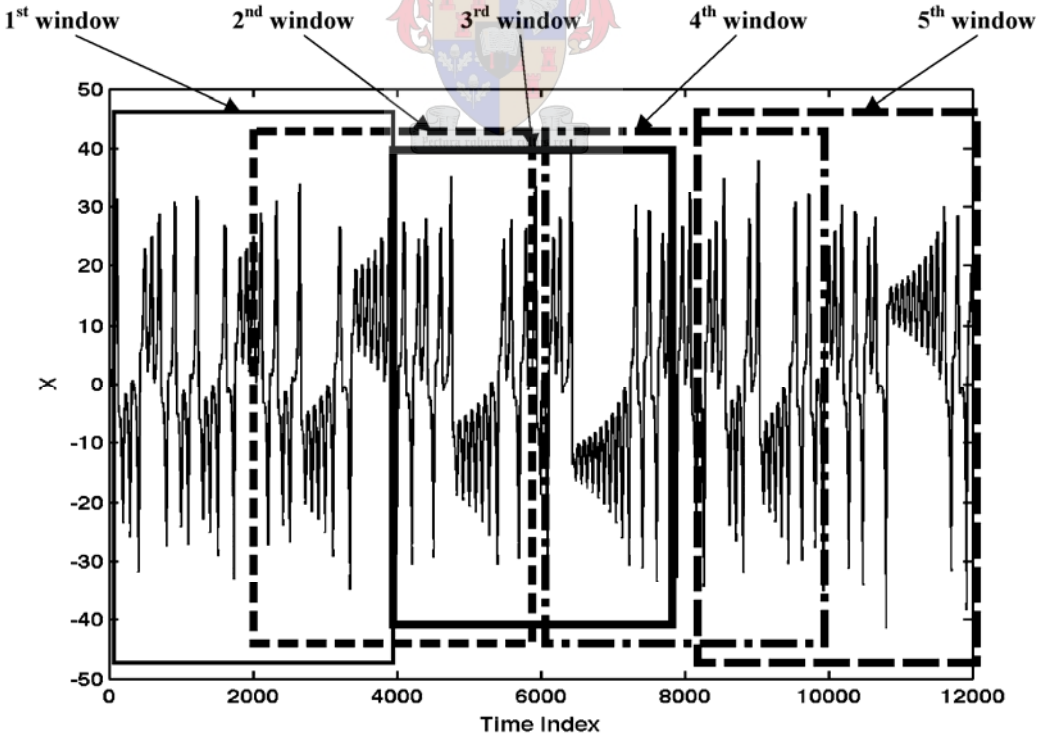


Figure 3.7: Moving window approach.

The result obtained from this approach is illustrated in Figure 3.8. No dynamic change is observed within the first two segments, since their correlation dimension curves are quite similar, a more formal approach to test similarity will be introduced later. In segment three a change in the dynamics of the system has occurred, not only causing the correlation dimension curve to increase but also to change its shape. When the last two segments are studied, the correlation dimension curves do not differ that much again, but they are different from those related to the first two segments. With closer inspection it is also clear that the change was instantaneous at the 6000th time interval since the two segments leading up to and following the 6000th time interval are similar respectively.

The moving window approach proofed to be superior and will be used for analysis of the simulated case studies in chapter 5.

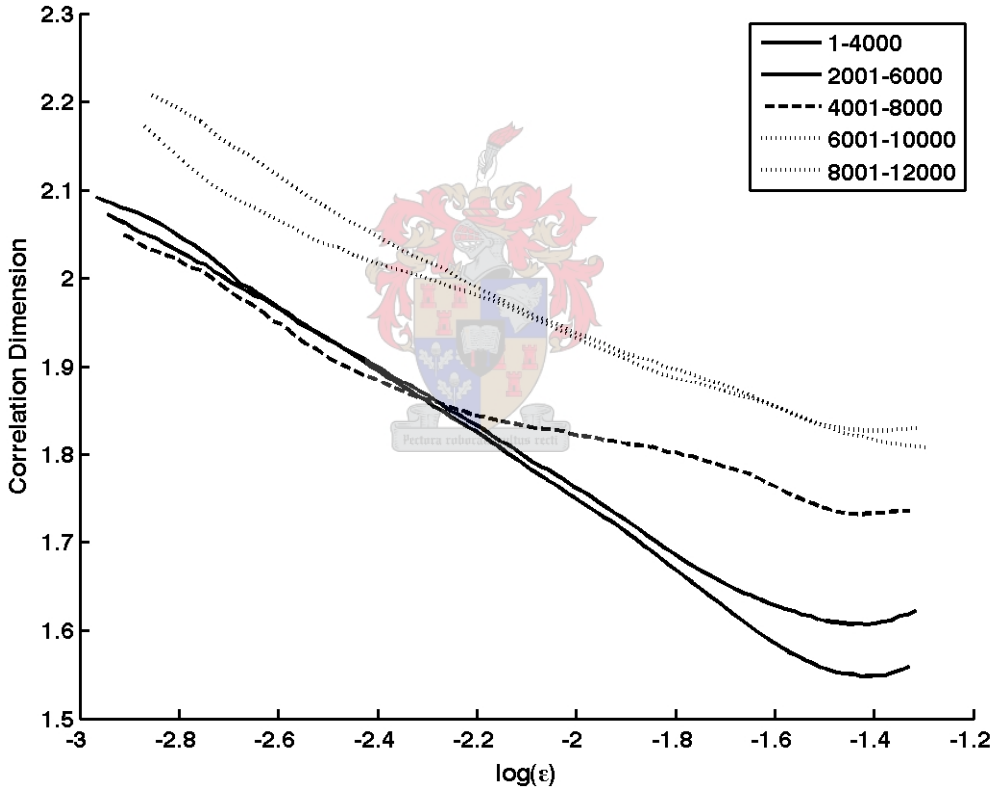


Figure 3.8: Result obtained from moving window approach.

In order to quantify the similarity between correlation dimension curves a decision rule is created. This is done by dividing the scaled length axis (x-axis) into a number of segments and creating 95% confidence intervals on the correlation dimension estimates for each segment with the following equation:

$$d_2 = \overline{d_2} \pm t_{df,\alpha/2} S \tag{3.10}$$

Where $\overline{d_2}$ is the mean of the correlation dimension, S^2 is the variance of correlation dimension and $t_{df,\alpha/2}$ is the value from the t-table that corresponds to a right-hand tail area of $\alpha/2$ for a t-distribution with df degrees of freedom. For equation 3.10 to be valid a normal distribution of the correlation dimension estimates is assumed for stationary data.

Now to obtain sufficient data to create the 95% confidence interval a calibration set is generated (simulated) in this study. This calibration set has the same parameters as the starting conditions of the simulated case studies and is sufficiently long to obtain 20 correlation dimension estimates from non-overlapping segments.

The procedure to specify normal process with confidence limits:

1. Simulate time series data (calibration set), with parameters the same as starting conditions of system, sufficiently long to obtain 20 correlation dimension estimates from (non-overlapping segments).

2. Determine suitable embedding parameters for phase space reconstruction. The time delay T is estimated with either the autocorrelation function or average mutual information and the embedding dimension d_e with the FNN algorithm.

3. Divide the data into a number of non-overlapping segments of length N, sufficiently long to capture the full dynamics of the system, and embed each segment using the above embedding parameters (time delay T and embedding dimension d_e).

4. Estimate the correlation dimension of the segments characterising the topology of the attractor that represents the normal process behaviour (dynamics of the system). Also estimate the mean and 95% confidence intervals from these estimates.

The procedure to monitor nonlinear time series for changes:

1. Simulate new data until a sufficient long time series, N samples, has been obtained. The new data can be combined with previous data to produce time series of sufficient length N, as described by the moving window approach, for more rapid monitoring.

2. Embed the data with the same embedding parameters, time delay T and embedding dimension d_e, determined for the normal process.

3. Estimate the correlation dimension for the new time series data and compare it with the normal process's estimates and limits. If the correlation dimension

falls outside the confidence limits of the normal process data, a change in the dynamics of the system has occurred.

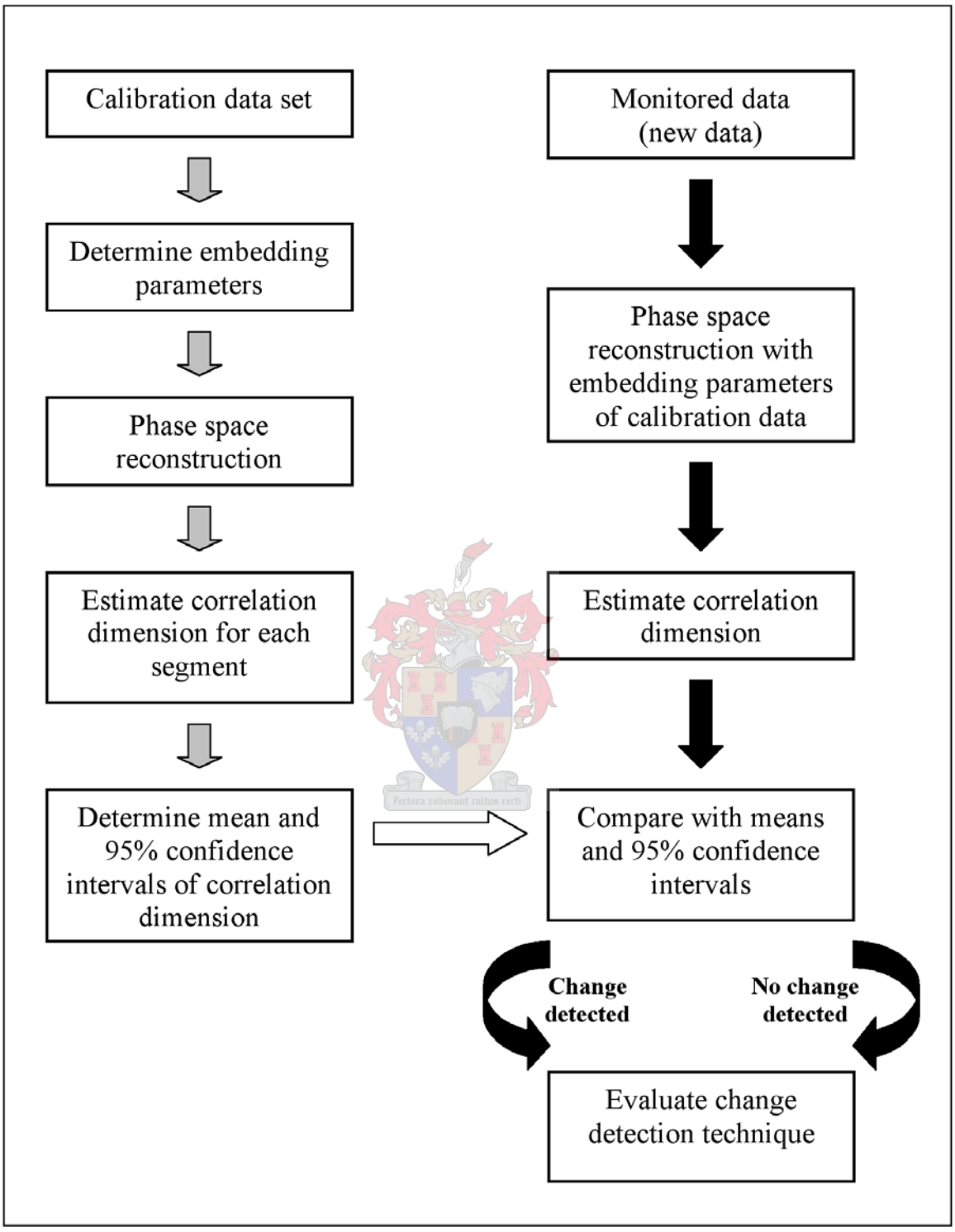


Figure 3.9: Schematic diagram of calibration and monitoring procedure using the correlation dimension as test statistic.

3.4. Summary of methodologies

Table 3.1 : Summary of parameters required for different change detection methods.

<table>
<tr><th></th><th>Change-Point Algorithm</th><th colspan="2">Mutual Cross Prediction</th><th>Correlation Dimension</th></tr>
<tr><td rowspan="5">Parameters to be Specified</td><td>Lag parameter M</td><td colspan="2">Time delay T</td><td rowspan="3">Time delay T</td></tr>
<tr><td>Set of eigenvectors retained I</td><td colspan="2">Embedding dimension d_e</td></tr>
<tr><td>Length of base set m</td><td colspan="2">Model parameters:</td></tr>
<tr><td>Length of test set Q</td><td>Weights</td><td rowspan="2">Embedding dimension d_e</td></tr>
<tr><td>Position of test set p</td><td>Number of neurons</td></tr>
<tr><td>Decision Rule</td><td>CUSUM-type statistic ≥ Threshold</td><td colspan="2">MSE of MLP ≥ 95% confidence limit</td><td>Correlation dimensions outside 95% confidence limits</td></tr>
</table>

4. Description of simulated case studies

Since the stationarity of real world experimental results is usually unknown, all three case studies considered in this chapter are simulated systems. A dynamic change is also induced in the simulation but the starting location, duration and size of the change are known for evaluation of the change detection algorithms. In chapter 5 the change detection algorithms are applied to the simulated systems discussed in this chapter.

4.1. Belousov-Zhabotinsky reaction (BZ reaction)

The Belousov-Zhabotinsky reaction is basically an unstable chemical reaction that maintains self-oscillations and propagating waves, which may display chaos under certain conditions. The reaction is the transition-metal-ion catalyzed oxidation and bromination of an organic dicarboxylic acid by bromate ions in an acidic aqueous medium (Zhang et al., 1993). The reaction was first founded by Boris Belousov in 1951 and was proven true later on by Anatol Zhabotinsky (Lu & Teulilo, 2001). Field et al. (1972) proposed the first detailed chemical mechanism for the BZ reaction. They also included a set of reaction rate constants based on thermodynamic data and experimental results that were available at the time. The chemical mechanism contains 15 chemical species and 11 reaction steps.

Györgyi et al. (1991) introduced a model consisting of 11 dynamic variables and 19 reactions that are able to exhibit deterministic chaos. They further reduced the model to 4 and 3-variable models for deterministic chaos (Györgyi & Field, 1991). The model used for the simulation in this case study, Model N, which was proposed by Györgyi & Field (1992) has the following chemical scheme (Zhang et al., 1993):

$$(1)\ X + Y + H \rightarrow 2V \tag{4.1}$$

$$(2)\ Y + A + 2H \rightarrow V + X \tag{4.2}$$

$$(3)\ 2X \rightarrow V \tag{4.3}$$

$$(4)\ \frac{1}{2}X + A + H \rightarrow X + Z \tag{4.4}$$

$$(5)\ X + Z \rightarrow \frac{1}{2} X \tag{4.5}$$

$$(6)\ V + Z \rightarrow Y \tag{4.6}$$

$$(7)\ Z + M \rightarrow products \tag{4.7}$$

Where $A = BrO_3^-$, $H = H^+$ and $M = MA$ are chemicals of constant concentrations. The concentrations of $Y = Br^-$, $X = HBrO_2$, $Z = Ce(IV)$ and $V = BrMA$ are variables of the model. The rates and constants are:

$$r_1 = k_1[H][X][Y],\ k_1 = 4.0 \times 10^6 \tag{4.8}$$

$$r_2 = k_2[A][H]^2[Y],\ k_2 = 2 \tag{4.9}$$

$$r_3 = k_3[X]^2\ , k_3 = 3000 \tag{4.10}$$

$$r_4 = k_4[A]^{0.5}[H]^{1.5}(C - [Z])[X]^{0.5},\ k_4 = 55.2 \tag{4.11}$$

$$r_5 = k_5[X][Z],\ k_5 = 7000 \tag{4.12}$$

$$r_6 = \alpha k_6[Z][V],\ k_6 = 0.09 \tag{4.13}$$

$$r_7 = \beta k_7[M][Z],\ k_7 = 0.23 \tag{4.14}$$

By assuming that Y is a fast variable the system of scaled differential equations are:

$$\frac{dx}{d\tau} = T_0\left[-k_1 H Y_0 x\breve{y} + \frac{k_2 A H^2 Y_0}{X_0}\breve{y} - 2k_3 X_0 x^2 + \frac{1}{2}k_4 A^{0.5} H^{1.5} X_0^{-0.5}(C - Z_0 z)x^{0.5} - \frac{1}{2}k_5 Z_0 xz - k_f x\right] \tag{4.15}$$

$$\frac{dz}{d\tau} = T_0\left[k_4 A^{0.5} H^{1.5} X_0^{0.5}\left(\frac{C}{Z_0} - z\right)x^{0.5} - k_5 X_0 xz - \alpha k_6 V_0 zv - \beta k_7 Mz - k_f x\right] \tag{4.16}$$

$$\frac{dv}{d\tau} = T_0\left[\frac{2k_1 H X_0 Y_0}{V_0} x\breve{y} + \frac{k_2 A H^2 Y_0}{V_0}\breve{y} + \frac{k_3 X_0^2}{V_0}x^2 - \alpha k_6 Z_0 zv - k_f v\right] \tag{4.17}$$

Where $\tau = time / T_0$ and $T_0 = (10 k_2 AHC)^{-1}$ are the scaled time with scaling factors: $x = X / X_0$, $X_0 = k_2 AH^2 / k_5$, $Y_0 = 4k_2 AH^2 / k_5$, $z = Z / Z_0$, $Z_0 = CA/(40M)$, $v = V / V_0$ and $V_0 = 4AHC / M^2$ which are the scaled concentration variables. The approximation of the fast variable Y is:

$$\breve{y} = \frac{\left[\dfrac{\alpha k_6 Z_0 V_0 zv}{k_1 H X_0 x + k_2 A H^2 + k_f}\right]}{Y_0} \tag{4.18}$$

Where C is the total cerium ion concentration and α, β, and k_f are adjustable parameters. Zhang et al. (1993) have shown chaotic behaviour for these chemical conditions at: $A = 0.1\,\text{M}$, $M = 0.25\,\text{M}$, $H = 0.26\,\text{M}$, $C = 0.000833\,\text{M}$, $\alpha = 6000/9$ and $\beta = 8/23$ for different windows of k_f, which is the flow rate.

The reaction model is simulated with the ODE45 subroutine in MATLAB 7.2 at a sampling rate of 0.00025 with initial values as follow:

$$\begin{aligned} x_0 &= 0.0099 \\ z_0 &= 2.2001 \\ v_0 &= 0.4582 \end{aligned} \tag{4.19}$$

After the first 10 000 stationary data points are obtained a dynamic change is introduced in the form of a slow drift of the constant parameter k_f, the flow rate. The parameter k_f slowly drifts from $4.5\times10^{-4}\,s^{-1}$ to $5.0\times10^{-4}\,s^{-1}$ over the next 10 000 data points. This drift is in the form of a step function, thus k_f increase with linear increments every 1 000 data points, with the first increase at the 10 000$^{\text{th}}$ data point and the last increase at the 19 000$^{\text{th}}$ data points. For the last 10 000 data points the parameter k_f is again kept constant at its final value. Thus a time series of 30 000 observations for each variable, x, z and v, are obtained. The x-variable, as used for modelling by Györgyi & Field (1991), is considered as the only measurable variable and is used as the time series data obtained from the system under investigation. The result obtained from the simulation is illustrated in Figure 4.1.

To ensure that all the internal dynamics of the system are captured, a discrete Fourier transform is performed that produced a continuous spectrum, as can be expected for a chaotic system, with the highest frequency of about 1000 Hz[8]. Thus the sampling period of 4000Hz should be sufficient to capture all the dynamics of the system.

The dynamic behaviour of the system is reconstructed by embedding the time series data into phase space. The first step in phase space reconstruction is to determine the embedding parameters. First the time delay is estimated with the autocorrelation function or the average mutual information statistic as illustrated in Figure 4.2 and Figure 4.3.

[8] No calculations shown here.

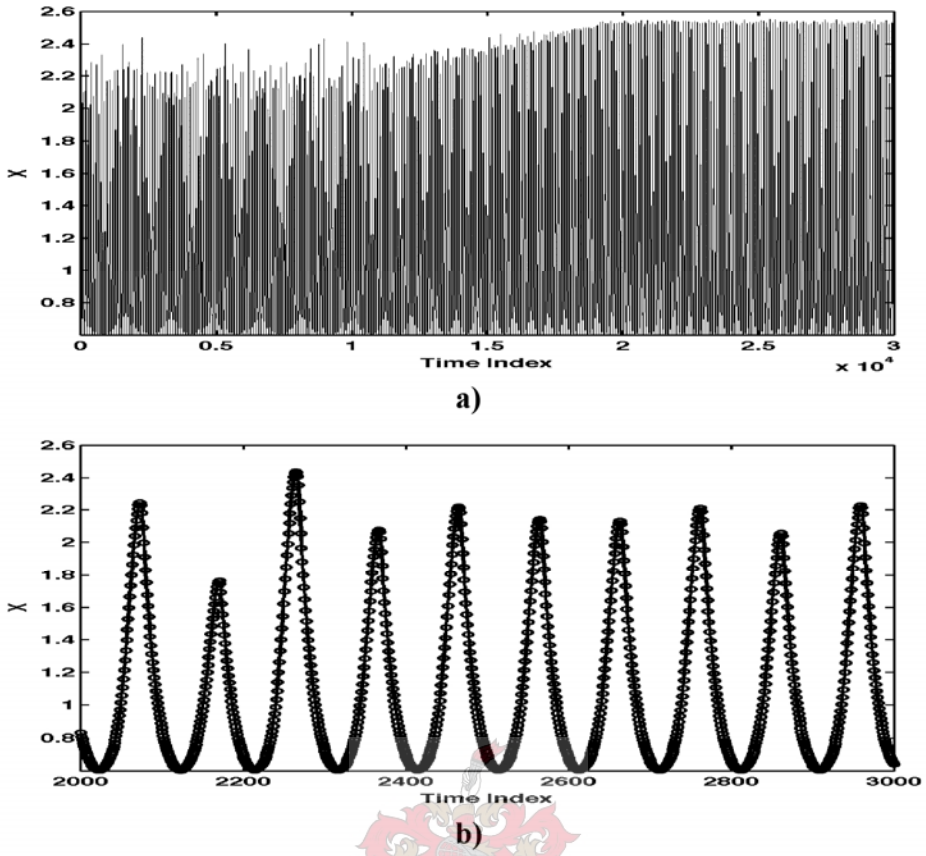


Figure 4.1: Time series obtained from BZ reaction model simulation **a)** all 30 000 data points **b)** 1000 data points.

The first zero value of the autocorrelation is at 24 time steps, where the first minimum value of the average mutual information is obtained at 41 time steps. Since the autocorrelation is a linear statistic and the average mutual information is seen as a superior choice as discussed in chapter 2, a time delay of 41 time steps is selected for the embedding.

Next the embedding dimension is determined by the FNN algorithm. In Figure 4.4 it is evident that there are no more FNN after three dimensions. This is as expected since the system is described by three deterministic equations, equation 4.15 to 4.17, and thus all the states of the system can be represented in three dimensions.

Now that all the embedding parameters have been evaluated for time delay embedding, the reconstruction is performed with equation 2.6 and the results illustrated in Figure 4.5. A typical strange attractor, filling the embedding space but still bounded, characteristic of chaotic systems, is obtained in Figure 4.5. To further investigate the possibility of chaos the maximal Lyapunov exponent was estimated at $\overline{\lambda}_{\max} \approx 0.0066$ for the first 10 000 data points. The positive Lyapunov exponent suggests chaotic behaviour.

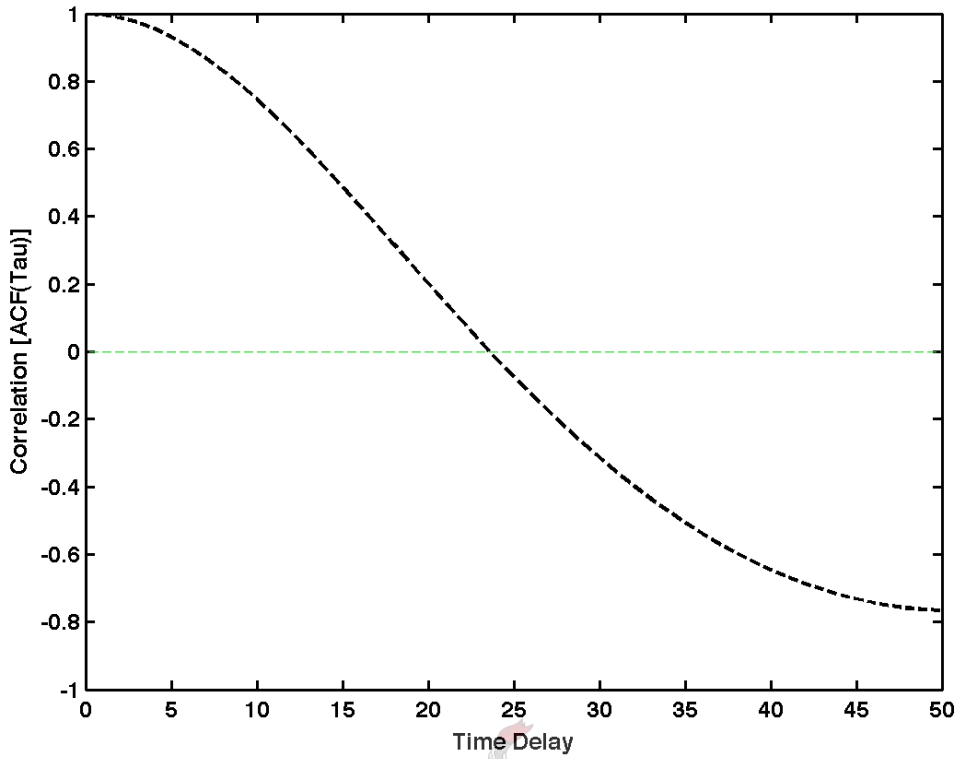


Figure 4.2: Autocorrelation function of BZ reaction data with first zero value at $T = 24$.

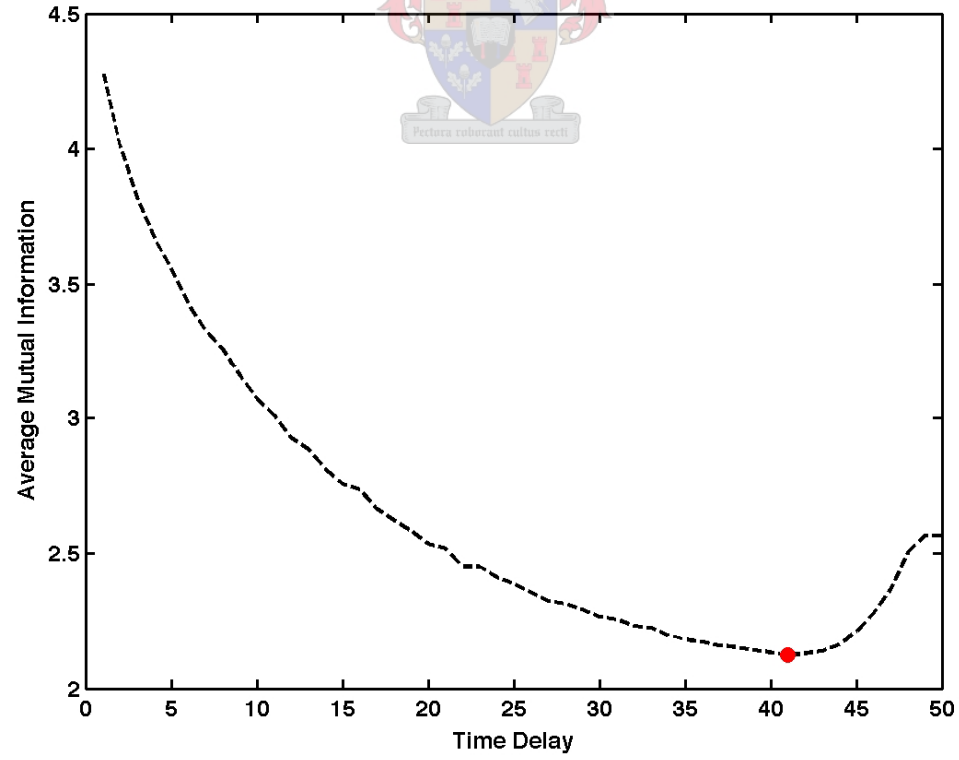


Figure 4.3: Average mutual information of BZ reaction data with first minimum indicated at $T = 41$.

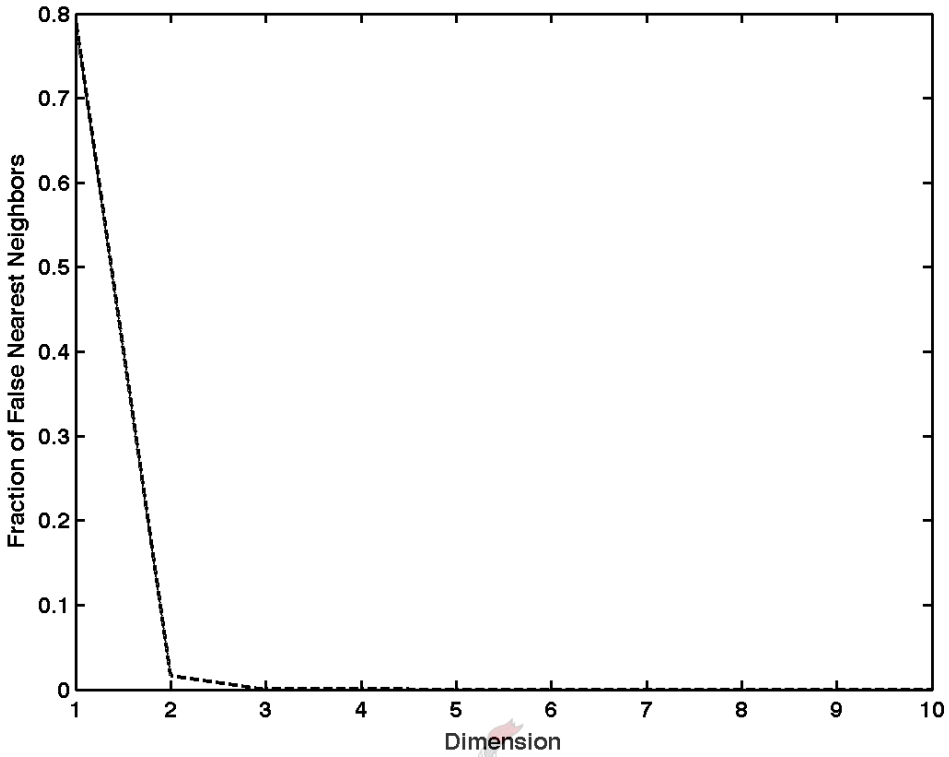


Figure 4.4: False nearest neighbours of BZ reaction data ($d_e = 3$).

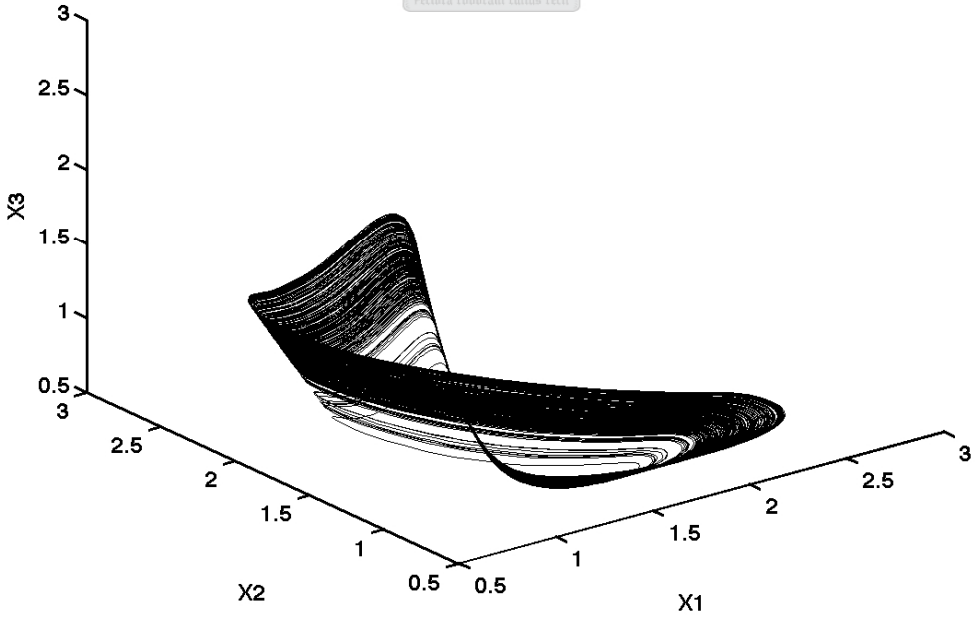


Figure 4.5: Reconstructed state space of BZ reaction data with time delay of $T = 41$ and embedding dimension of $d_e = 3$.

4.2. Autocatalytic process

The autocatalytic process considered in this simulation consists of two parallel, isothermal autocatalytic reactions taking place in a continuous stirred tank reactor (CSTR) (Lee & Chang, 1996). The system is capable of producing self-sustained oscillations based on cubic autocatalysis with catalyst decay at certain parameters. The system proceeds mechanically as follow:

$$A + 2B \rightarrow 3B \tag{4.20}$$

$$B \rightarrow C \tag{4.21}$$

$$D + 2B \rightarrow 3B \tag{4.22}$$

Where A, B and C are participating chemical species. The reaction rates are governed by the following rate equations:

$$-r_A = k_1 c_A c_B^2 \tag{4.23}$$

$$r_C = k_2 c_B \tag{4.24}$$

$$-r_D = k_3 c_D c_B^2 \tag{4.25}$$

Where k_1, k_2 and k_3 are the rate constants for the chemical reactions and c_A, c_B and c_D are concentrations of species A, B and D respectively. To describe the system by three ordinary differential equations the dimensionless concentration, ratios of species in feed and dimensionless time is described as follow:

$$X = \frac{C_A}{C_{A,0}}, \qquad Y = \frac{C_D}{C_{D,0}}, \qquad Z = \frac{C_B}{C_{B,0}} \tag{4.26}$$

$$a = \frac{k_1 C_{B,0}^2 V}{Q}, \quad b = \frac{k_3 C_{B,0} V}{Q}, \quad c = \frac{k_2 V}{Q} \tag{4.27}$$

$$\gamma_1 = \frac{C_{A,0}}{C_{B,0}}, \qquad \gamma_2 = \frac{C_{D,0}}{C_{B,0}} \tag{4.28}$$

$$\tau = \frac{t.Q}{V} \tag{4.29}$$

The ordinary differential equations governing the system is then defined by:

$$\frac{dX}{dt} = 1 - X - aXZ^2 \tag{4.30}$$

$$\frac{dY}{dt} = 1 - Y - bYZ^2 \tag{4.31}$$

$$\frac{dZ}{dt} = 1 - (1+c)Z + \gamma_1 aXZ^2 + \gamma_2 bYZ^2 \tag{4.32}$$

Lee and Chang (1996) have shown that for $a = 18000$; $b = 400$; $c = 80$; $\gamma_1 = 1.5$; $\gamma_2 = 4.2$ and with initial conditions $X_0 = 0$; $Y_0 = 0$; $Z_0 = 0$, the system exhibits chaotic behaviour.

The system is simulated with the ODE45 subroutine in MATLAB 7.2 at a sampling rate of 0.005. For the first 10 000 observations the same parameters as specified by Lee and Chang (1996) are used. A dynamic change is then introduced by creating a hypothetical case where the feed concentrations of A and D increase slightly. Although the exact increase of A and D are not considered, it can be reasoned from equations 4.28 that this will cause an increase in the dimensionless parameters γ_1 and γ_2. Thus over the next 10 000 observations the parameters, γ_1 and γ_2, drift slowly from their starting values, $\gamma_1 = 1.5$ and $\gamma_2 = 4.2$, to $\gamma_1 = 1.55$ and $\gamma_2 = 4.25$. The drift is again in the form of a step function, thus γ_1 and γ_2 increase with linear increments every 1 000 data points, with the first increase at the 10 000th data point and the last increase at the 19 000th data points. The parameters are again kept constant at their new values, $\gamma_1 = 1.55$ and $\gamma_2 = 4.25$, for the final 10 000 observations. The 30 000 observations of the X variable, illustrated in Figure 4.6, is considered as the only data available from the system, as used by Bezuidenhout (2004).

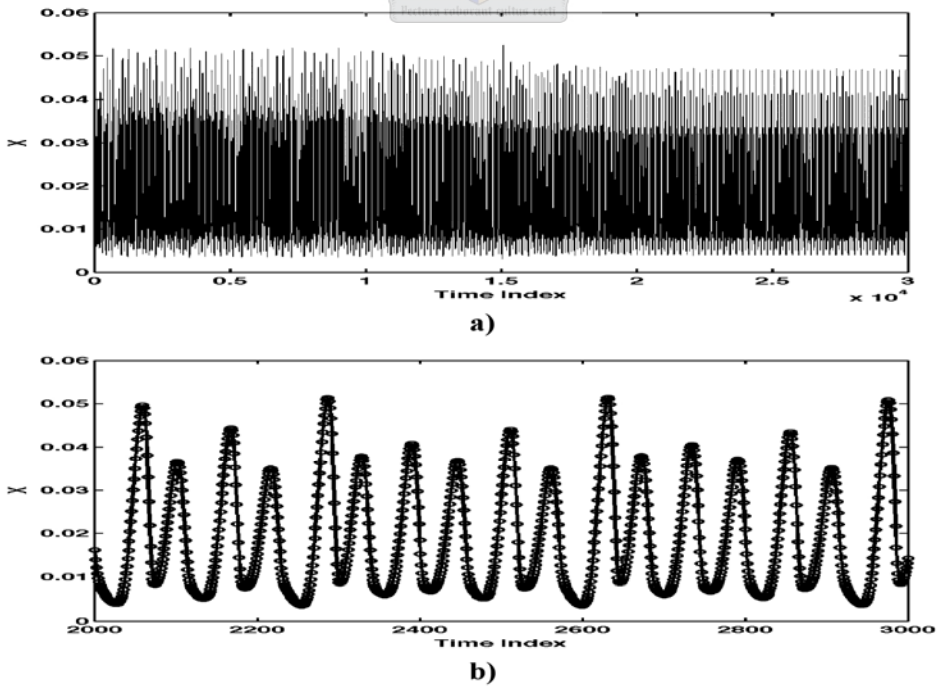


Figure 4.6: Time series obtained from autocatalytic process model simulation **a)** all 30 000 data points **b)** 1000 data points.

The sampling period of 200Hz also seems reasonable since the highest frequency component obtained from the Fourier spectrum is about 20Hz. Reconstructing the phase space is again done by time delay embedding which is summarized in Figure 4.7. A typical strange attractor is again obtained with a maximal Lyapunov exponent estimated at $\overline{\lambda}_{\max} \approx 0.0052$ for the first 10 000 observations, suggesting chaotic behaviour of the system.

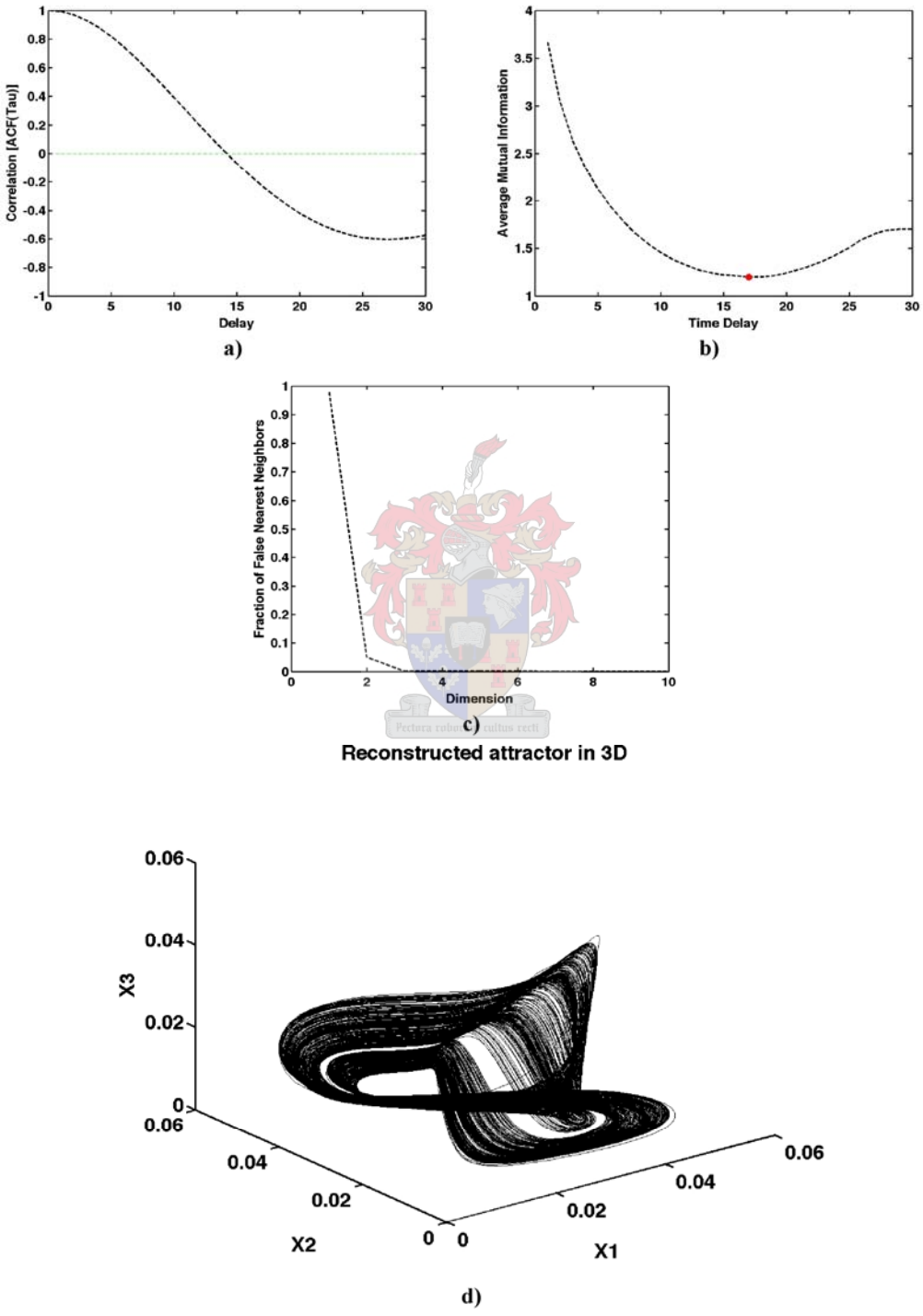


Figure 4.7: Summary of state space reconstruction by time delay embedding of the autocatalytic process data **a)** autocorrelation function ($T = 14$) **b)** average mutual information ($T = 17$) **c)** FNN ($d_e = 3$) **d)** reconstructed state space with $T = 17$ and $d_e = 3$.

4.3. Lotka-Volterra predator-prey model

Some of the most interesting dynamics in the biological world are the interaction between species. Recently there have been many developments in use of biological processes in process engineering. Thus the relevance of studying such a predator-prey model is evident.

The Lotka-Volterra was one of the first models introduced to describe the interactions between predator and prey species. It was first proposed by an American biophysicist Alfred Lotka and an Italian mathematician Vito Volterra in 1925. These types of models have broader relevance to other fields of study as well, such as a bio-physical coastal ecosystem model by Dowd (2005), a atmospheric chemistry model by Wang et al. (2002), a juvenile salmon migration model by Anderson (2005) and a model on plankton predation rates by Lewis and Bala (2006). A predator-prey model proposed by Lindfield and Penny (2000) is evaluated in this study. It is based on the following two differential equations:

$$\frac{dX}{dt} = k_1 X - CXY \tag{4.33}$$

$$\frac{dY}{dt} = k_2 XY - DY \tag{4.34}$$

Where k_1 is the rate of prey population growth, C the encounter rate (reaction rate), k_2 the predator's efficiency to turn food into offspring (conversion efficiency) and D the predator death rate. The model is a nonlinear system that is not able to exhibit chaotic behaviour since only two variables are present. An oscillating behaviour are established when the constant parameters are set at $k_1 = 2$, $k_2 = 10$, $D = 0.002$ and $C = 0.001$. Image a hypothetical situation where the environment is changed to favour the conversion efficiency k_2, i.e. temperature of reactor increases to favour conversion in a biological reactor. This will increase the rate at which the prey population grow k_1. Also an increase in encounter rate C and predator death rate D can now be expected. Thus the parameters changing as follow:

Start		End
$k_1 = 2$	⟶	$k_1 = 3$
$k_2 = 10$	⟶	$k_2 = 11$
$C = 0.001$	⟶	$C = 0.0011$
$D = 0.002$	⟶	$D = 0.0021$

Now the system is simulated with the ODE45 subroutine in MATLAB 7.2 with a sampling rate of 0.02 seconds and starting values of the two variables, X and Y, at 100 and 5000 respectively. The parameters are kept constant at their starting values for the first 10 000 observations where after a dynamic change, the hypothetical situation, is introduced. Over the next 10 000 observations the parameters drift slowly as described in the hypothetical situation .The drift is again in the form of a step function, thus the parameters increase with linear increments every 1 000 data points, with the first increase at the 10 000th data point and the last increase at the 19 000th data point. Thereafter the parameters are kept constant at the end values described in the hypothetical situation for the final 10 000 observations. The X-variable, 30 000 observations, is considered as the only data available from the system under investigation. The result obtained from the simulation is illustrated in Figure 4.8. From a discrete Fourier transform the highest frequency component present in data is 10 Hz. Thus the sampling frequency of 50 Hz should be adequate to capture all the dynamics of the system.

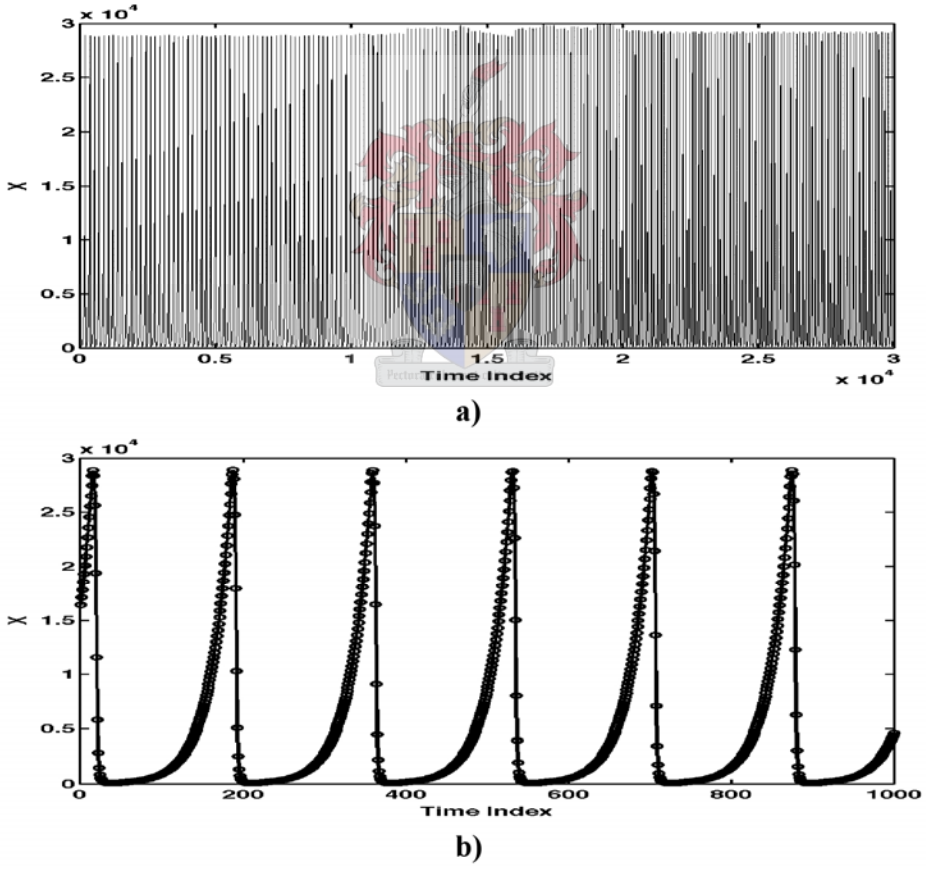


Figure 4.8: Time series obtained from the predator-prey model simulation **a)** all 30 000 data points **b)** first 1000 data points.

Reconstructing the phase space is again done by time delay embedding which is summarized in Figure 4.9. Although a very small number of FNN are still present in two dimensions, it should be borne in mind that t is a purely deterministic system, and

therefore three dimensions were selected for embedding, where the number of FNN was zero.

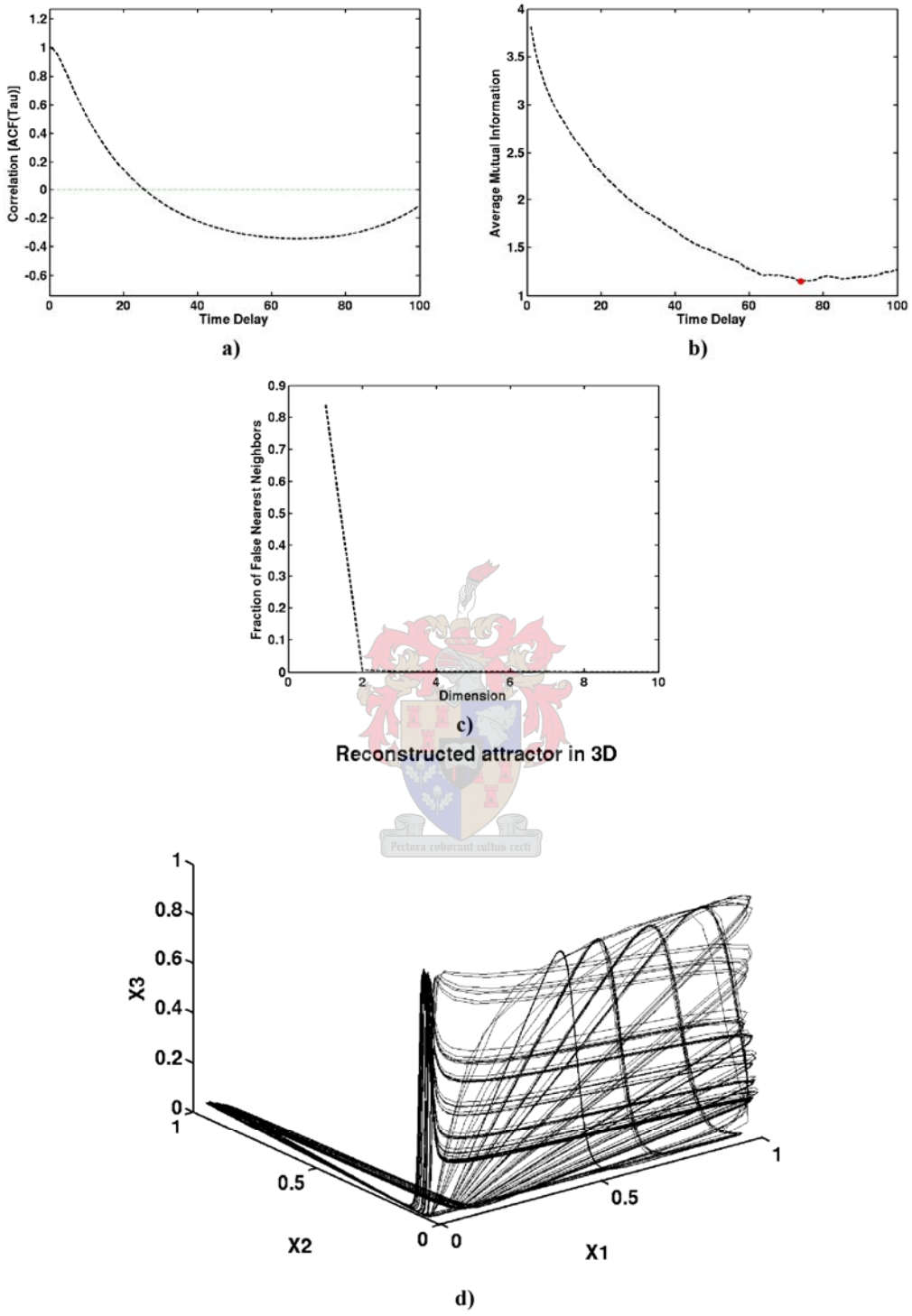


Figure 4.9: Summary of state space reconstruction by time delay embedding of the predator-prey data **a)** autocorrelation function ($T = 26$) **b)** average mutual information ($T = 74$) **c)** FNN ($d_e = 3$) **d)** reconstructed state space with $T = 74$ and $d_e = 3$.

5. Detecting dynamic change in simulated case studies

In this chapter the dynamic change detection algorithms discussed in chapter 3 are applied to the three simulated case studies described in chapter 4.The methods are:

- Change-point detection based on SSA (section 3.1).
- Nonlinear cross prediction of forecasting error from MLP model (section 3.2).
- Using the correlation dimension as detection statistic (section 3.3).

It is important to note that in this section onwards, reference to the correlation dimension will imply the Judd (1992) variant of the correlation dimension, i.e. the correlation dimension as a function of scale, as opposed to the binary value obtained from the Grassberger and Procaccia (1983) implementation.

5.1. Detecting dynamic change with the change-point detection algorithm

The change-point detection algorithm is based on determining an l-dimensional hyperplane by SSA for a base set and determining the sum of squares of the distance between the l-dimensional hyperplane and a test set. Should a change in dynamics be encounter an increase in the detection statistic is expected.

5.1.1. Belousov-Zhabotinsky reaction

The first step in the algorithm is to perform the reconstruction of the time series with SSA. Two parameters are required for the SSA, the lag parameter M and the first l principal components that describe the main trends of the signal. The lag parameter $M = 24$ is obtained from the autocorrelation function in Figure 4.2. The trajectory matrix is then obtained by time delay embedding with a time delay $T = 1$ and an embedding dimension $d_e = M = 24$.

The second step, the singular value decomposition of the trajectory matrix, is then performed and the eigenvalues obtained are shown in Figure 5.1. The first 2 principal components are selected to describe the signal of the BZ reaction data. The time series

is then reconstructed with the 2 principal components and the result is illustrated in Figure 5.2.

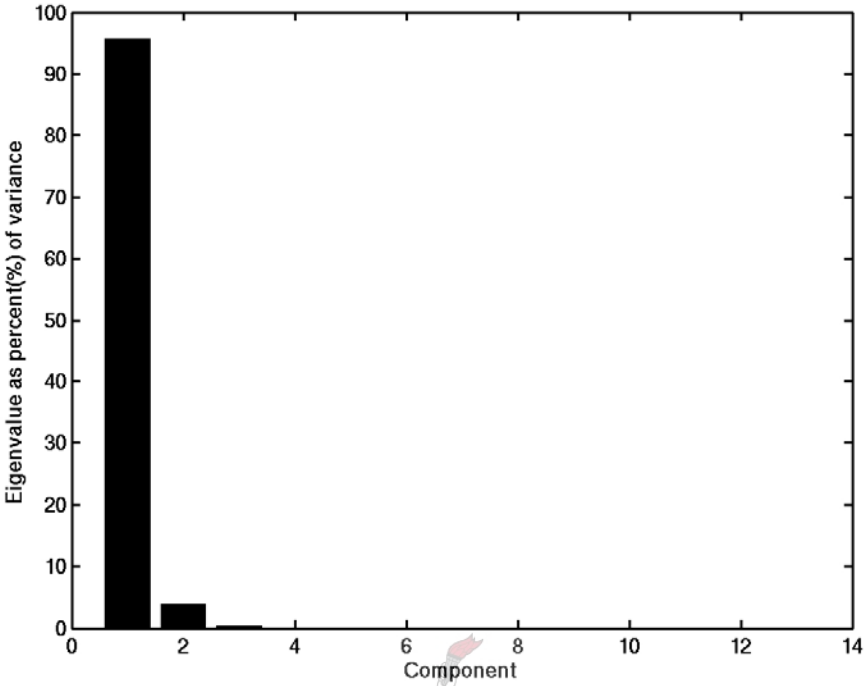


Figure 5.1: Eigenvalues of lag-covariance matrix of BZ reaction data ($M = 24$).

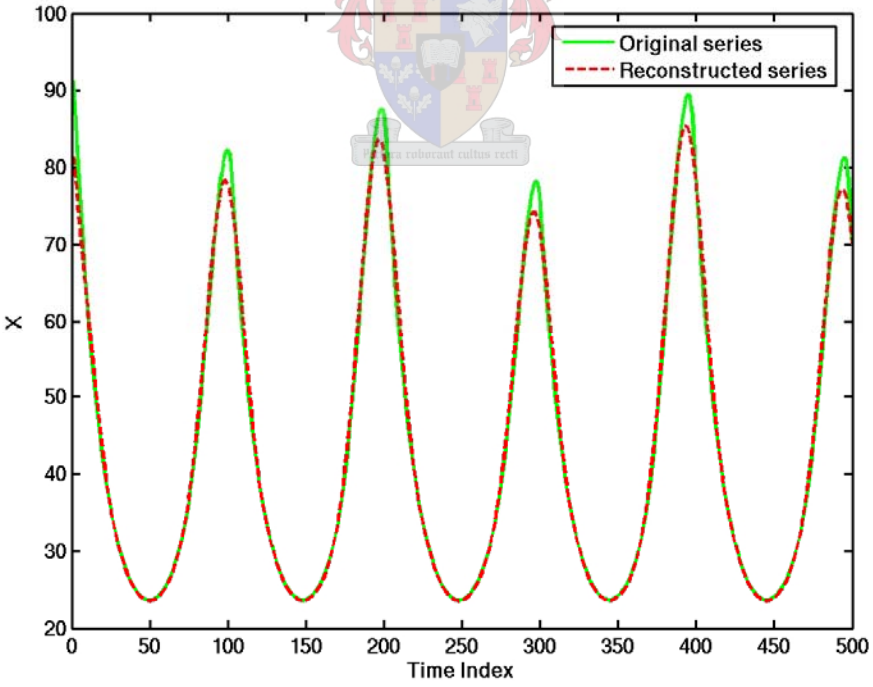


Figure 5.2: Reconstructed time series of BZ reaction data ($M = 24$ and $I = 2$).

Finally the parameters for the detection statistic $D_{n,I,p,q}$ are selected. A reasonably large window width for the base set, $m = 400$, is selected in order to avoid a haphazard behaviour of the detection statistic. Then a large test set, $Q = 300$ with

$p = 124$, is chosen to ensure smooth behaviour of the detection statistic. The results obtained from the test statistics are illustrated in Figure 5.3 and Figure 5.4.

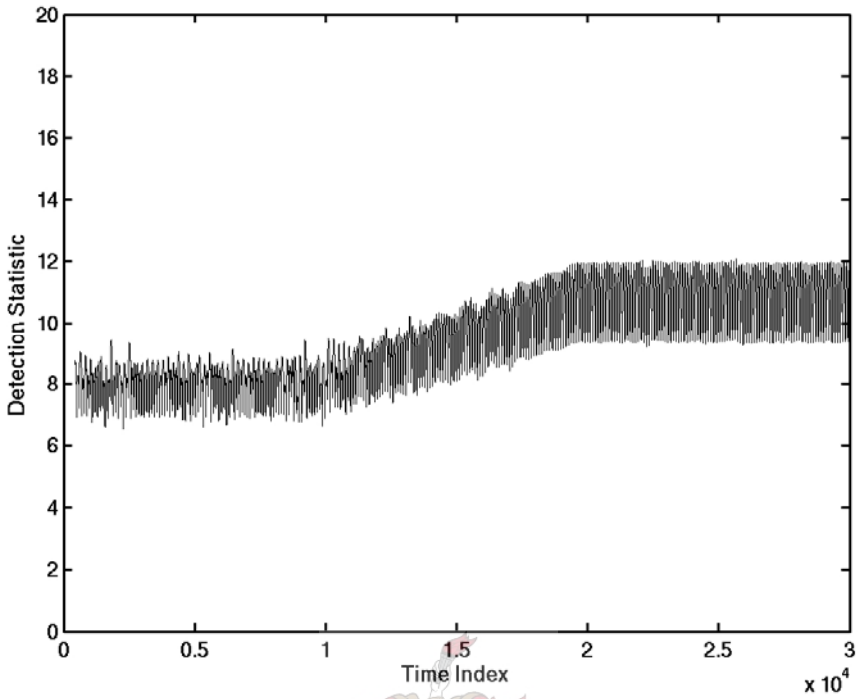


Figure 5.3: Detection statistic $D_{n,I,p,q}$ obtained for BZ reaction data ($M = 24$, $I = 2$, $m = 400$, $Q = 300$ and $p = 124$).

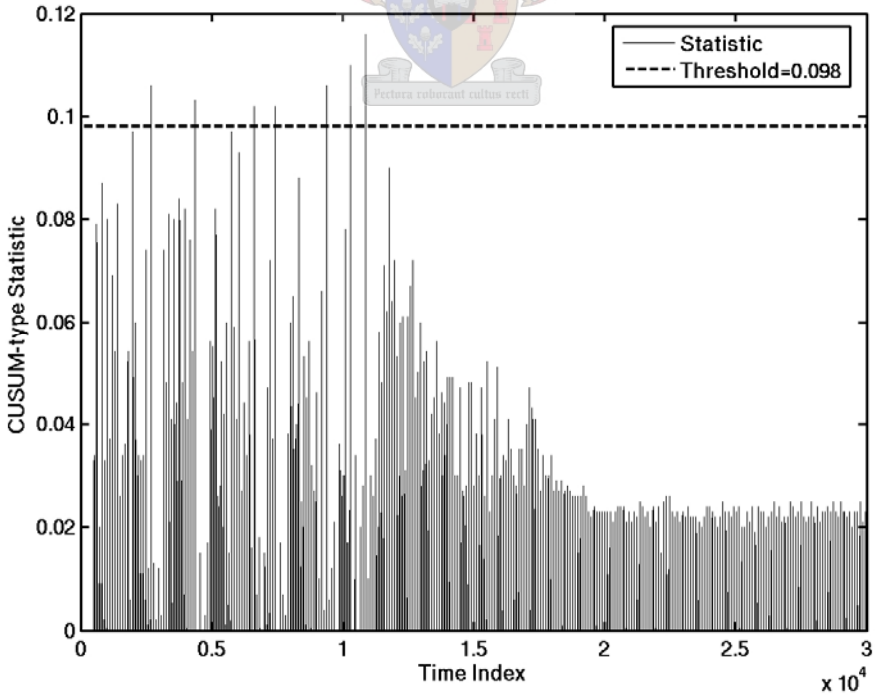


Figure 5.4: CUSUM statistic W_n obtained for BZ reaction data.

5.1.2. Autocatalytic process

The same procedure was followed as in section 5.1.1 with modified parameters summarized in the caption of Figure 5.5.

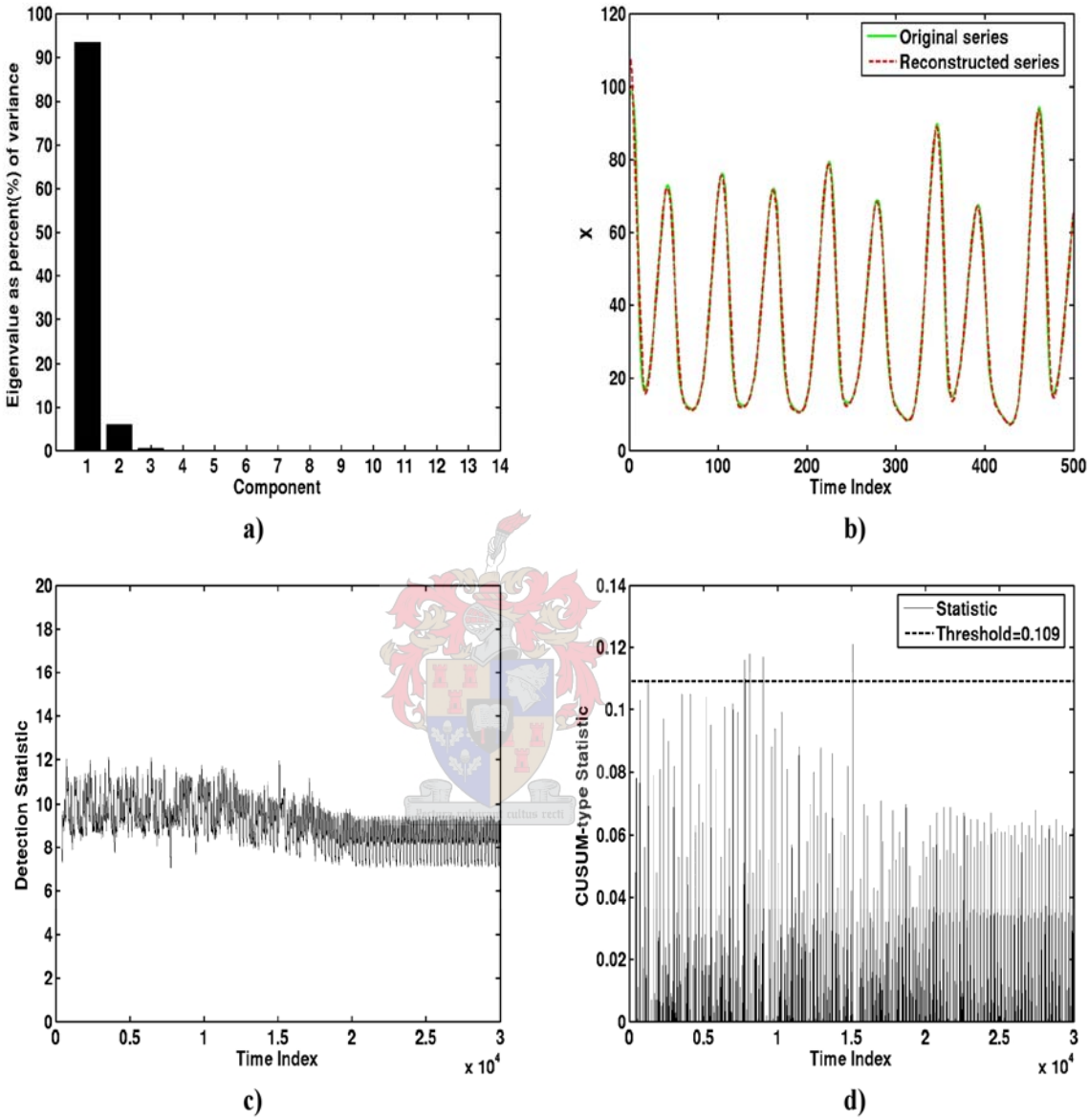


Figure 5.5: Summary of results from change-point detection algorithm for autocatalytic process data **a)** eigenvalues ($M = 14$) **b)** reconstructed time series ($I = 2$) **c)** detection statistic $D_{n,I,p,q}$ ($m = 400$, $Q = 400$ and $p = 14$) **d)** CUSUM statistic W_n.

5.1.3. Lotka-Volterra predator prey model

The same procedure was followed as in section 5.1.1 with modified parameters summarized in the caption of Figure 5.6.

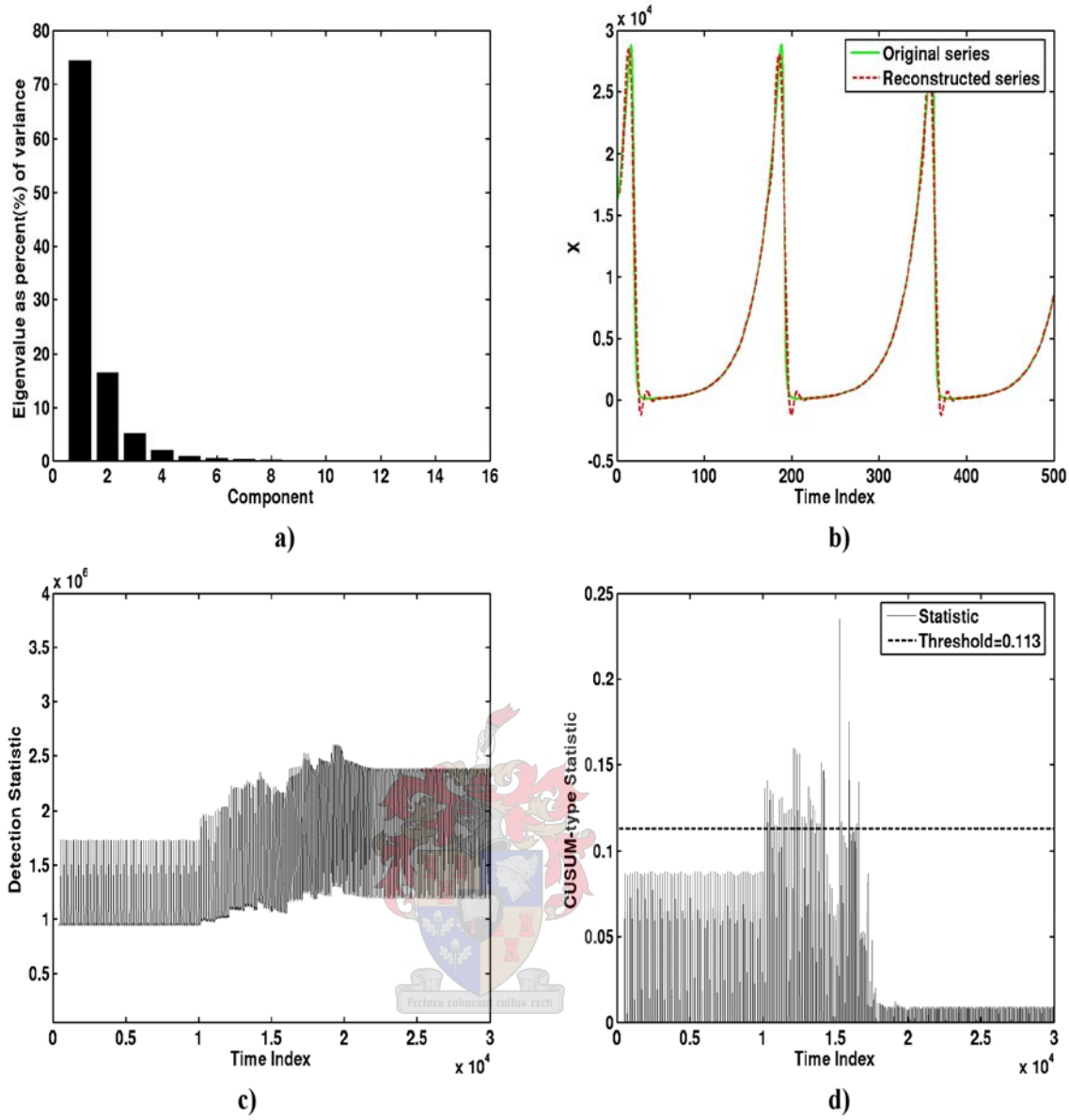


Figure 5.6: Summary of results from change-point detection algorithm predator-prey data **a)** eigenvalues ($M = 26$) **b)** reconstructed time series ($I = 4$) **c)** detection statistic $D_{n,I,p,q}$ ($m = 400$, $Q = 220$ and $p = 206$) **d)** CUSUM statistic W_n.

5.1.4. Discussion of results obtained with change-point detection algorithm

The results obtained from all three simulated case studies indicated that dynamic change is present in the time series data. A dynamic change is identified when a value of the CUSUM statistic W_n above the threshold is obtained. The increase of the CUSUM statistic W_n is due to an increase of the mean level of detection statistic $D_{n,I,p,q}$, as observed in Figure 5.3 and Figure 5.6c. Once the non-stationarity is encountered the mean level of the detection statistic $D_{n,I,p,q}$ changes due to the changing shaped of the attractor. In the cases of the BZ reaction and predator-prey

model the attractors change in such a way that it moves further away from the base set's l-dimensional subspace, thus causing the increase in the detection statistic $D_{n,I,p,q}$. Refer to Figure 5.7a for an illustration of the last 1000 data points (green points) of the BZ reaction projected onto the essentially two-dimensional ($2D$) subspace (red points) defined by the first two principal components of the first 1000 data points (black points) of the BZ reaction, as indicated by the variance explained by each PC. Note that the black and green points do not lie in this $2D$-subspace. Clearly the sum of the squared distances between the red and green points are larger than the sum of squared distances between the black and blue points (projection of the black points onto its first two principal components). Hence the initial increase of the test statistic. Figure 5.7b illustrates each segment (first 1000 and last 1000 data points) projected onto its own $2D$-subspace (defined by first two principal components), but there is only a small difference between the two planes. Thus the sum of the squared distances between the red and green points (last 1000 data points) are still larger than the sum of squared distances between the black and blue points (first 1000 data points) explaining the higher average level of the test statistic for the final 10 000 data point.

The predator-prey data produced the best behaviour of the detection statistic $D_{n,I,p,q}$ and the CUSUM statistic W_n (Figure 5.6c and Figure 5.6d). An increase to above the threshold of the CUSUM statistic W_n are observed due to the increasing mean level of the detection statistic $D_{n,I,p,q}$ cause by non-stationary data points being included in the base and test stets. A final peak of the detection statistic $D_{n,I,p,q}$ can be observed before it settles down into its final mean level. The decrease after the peak is due to less and less non-stationarity data being included in the base set and thus the base set's l-dimensional subspace and test set's vectors coming closer to one another in phase space. The decrease is terminated once no more non-stationarity data is included in the base set and a new level is reach since for any further iterations the base set and test set are from the same stationary series. The CUSUM statistic W_n also decreases to below the threshold as the detection statistic $D_{n,I,p,q}$ reaches its new level, indicating that no more dynamic changes are present.

The haphazard behaviour of the detection statistic $D_{n,I,p,q}$ in all three cases is due to the data not having a Gaussian normal distribution and using only a limited number (base window width m and test window width Q) of data points to calculate the detection statistic $D_{n,I,p,q}$. The chaotic behaviour of the autocatalytic process and BZ reaction aggravates the irregular behaviour of the detection statistic $D_{n,I,p,q}$ even more because of the irregular shape of the attractor since limited data is considered. The aggravated irregular behaviour causes multiple false alarms to be detected by the CUSUM statistic W_n. Increasing the two parameters m and Q have a smoothing

effect on the detection statistic $D_{n,I,p,q}$, but the effect of the change in dynamics is also smoothed out.

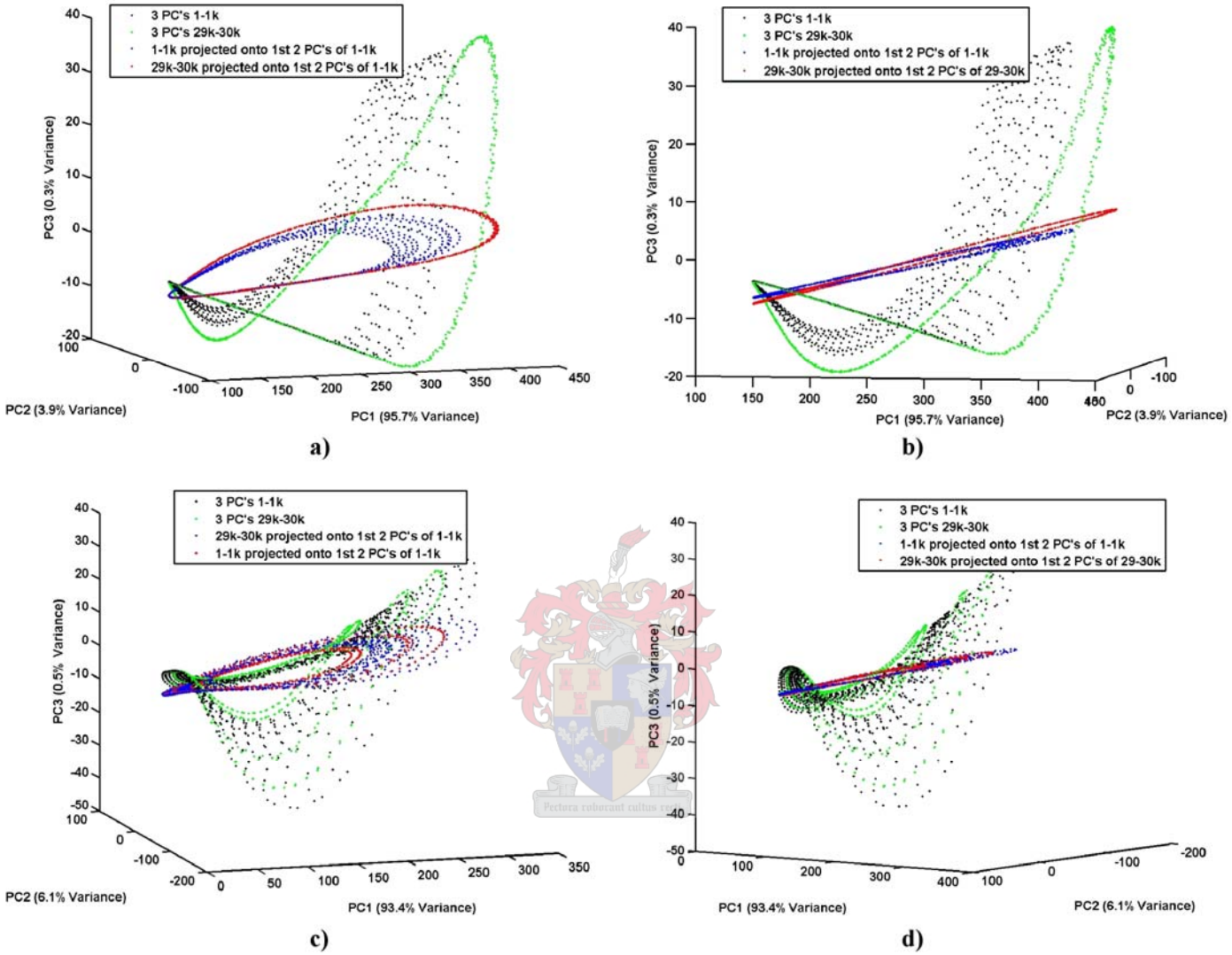


Figure 5.7: Projection of data segments onto lower subspaces **a)** BZ reaction data, both segments onto first segments 1st 2PCs **b)** BZ reaction data, each segment onto their own 1st 2PCs **c)** Autocatalytic process data, both segments onto first segments 1st 2PCs **d)** Autocatalytic process data, each segments onto their own 1st 2 PCs.

In the case of the autocatalytic process the attractor changes from having chaotic properties that fill the subspace to a nonlinear "thin" attractor that are embedded in the chaotic attractor. This causes the sum of the squared distances between the final 1000 data points and its projection onto its own $2D$-subspace to be less than that of the first 1000 data points projection onto its own $2D$-subspace, as illustrated in Figure 5.7d. This explains the decrease in average level of detection statistic $D_{n,I,p,q}$ as illustrated in Figure 5.5c. It can thus be argued that the alarm for dynamic change by the CUSUM statistic W_n is actually caused by the chaotic behaviour rather than the structural change of the attractor.

In the cases of the predator-prey data and BZ reaction data the algorithm is quick to detect structural changes, less than 1000 non-stationary data points required, but multiple false alarms are encountered in the presence of chaotic behaviour.

A major disadvantage of the algorithm is that in the case where it fails to detect a slow dynamic change, the change is integrated into the "normal" state of the system. Thus the dynamics of the system can continue to change until a system with total different dynamics is obtained, without the algorithm ever detecting a change. To avoid this problem perhaps a better implementation of the algorithm will be to select one base sample against which all test samples will be tested. This will prevent the algorithm from detecting multiple changes and identifying a new steady state after a dynamic change.

5.1.5. Effect of noise on change-point detection algorithm

The ability of the algorithm to detect dynamic changes is influenced by the presence of noise (Figure 5.8 and Figure 5.9). The presence of noise aggravates the irregular behaviour of the detection statistic $D_{n,I,p,q}$, which may lead to an increase in false alarms by the CUSUM statistic W_n as in the case of the autocatalytic process. As can be expected there is an increase in the overall mean level of detection statistic $D_{n,I,p,q}$ as the noise level increases. This is due to the vectors being scattered more in higher dimensions and are thus further away from the l-dimensional subspace, increasing their squared distances. This scattering effect of the noise has the same effect on the detection statistic $D_{n,I,p,q}$ as a dynamic change has on it. Thus decreasing the ability of the algorithm to resolve dynamic changes. Even changing parameters to obtain smoother behaviour of the detection statistic $D_{n,I,p,q}$ will also cause the dynamic change to be smoothed out.

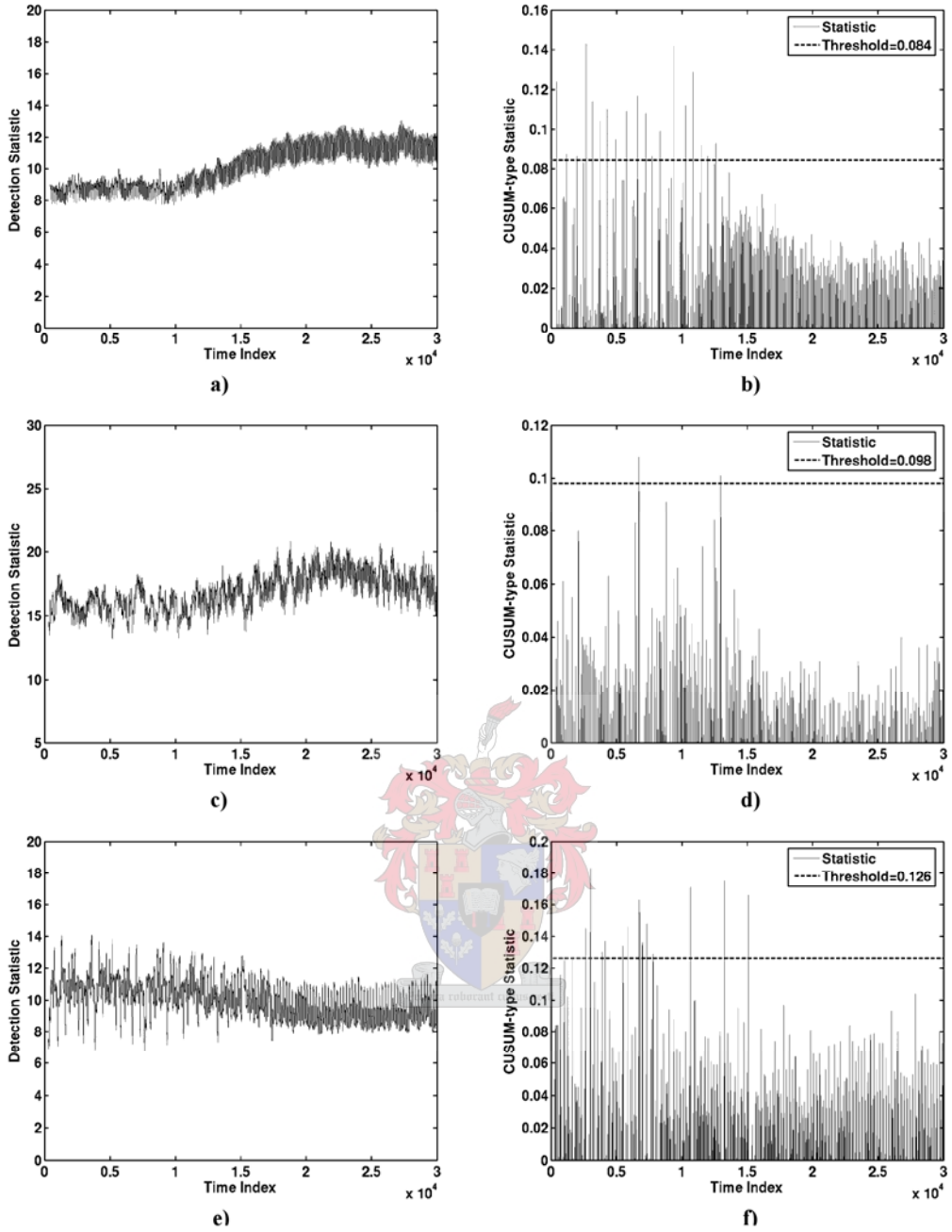


Figure 5.8: Results of change-point detection algorithm on noisy data **a)** detection statistic $D_{n,I,p,q}$ of BZ reaction data (5% noise) **b)** CUSUM statistic W_n of BZ reaction data (5% noise) **c)** detection statistic $D_{n,I,p,q}$ of BZ reaction data (15% noise) **d)** CUSUM statistic W_n of BZ reaction data (15% noise) **e)** detection statistic $D_{n,I,p,q}$ of autocatalytic process data (5% noise) **f)** CUSUM statistic W_n of autocatalytic process data (5% noise).

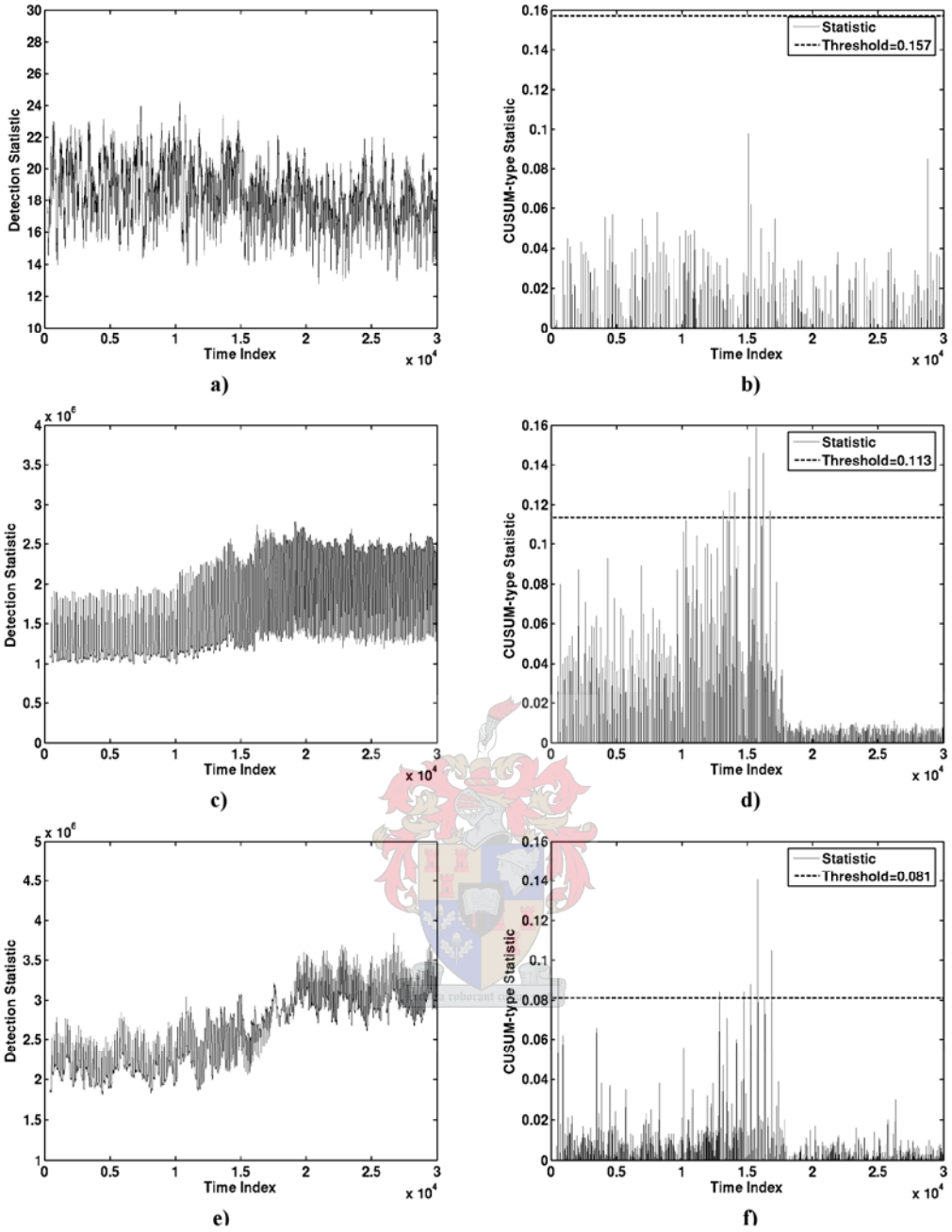


Figure 5.9: Results of change-point detection algorithm on noisy data **a)** detection statistic $D_{n,I,p,q}$ of autocatalytic process data (15% noise) **b)** CUSUM statistic W_n of autocatalytic process data (15% noise) **c)** detection statistic $D_{n,I,p,q}$ of predator-prey data (5% noise) **d)** CUSUM statistic W_n of predator-prey data (5% noise) **e)** detection statistic $D_{n,I,p,q}$ of predator-prey data (15% noise) **f)** CUSUM statistic W_n of predator-prey data (15% noise).

5.2. Detecting dynamic change with mutual cross prediction

Mutual cross prediction is a test for non-stationarity based on predicting one part of a time series by using another part as database. In this particular case a multilayer perceptron network (MLP) is trained to predict one step into the future of a certain part of the time series. The same MLP is then used to predict one step ahead for a different part of the time series. The prediction errors of the two parts are then compared. The results are displayed on a two-dimensional map. Should one part be selected before a dynamic change and the second part after a significant increase in the prediction error is expected, no matter whether the first part is used to predict the second part or the other way around. In this section the mean square error (MSE), squared difference between the predicted values and actual values, is used as the prediction error. A decision rule is created by calculating a 95% confidence limit of the prediction MSE for the stationary parts of the time series. A dynamic change is identified once a prediction MSE is higher than the confidence limit.

The time series of each simulation is divided into several parts using the moving window approach. Note that Table 5.1 also indicate the labels representing the different segments as is used in the proceeding figures.

Table 5.1 : Division of data into different segments.

Segment	Data range	Label	Segment	Data range	Label
1	1-5000	1-5k	14	13000-18000	13k-18k
2	1000-6000	1k-6k	15	14000-19000	14k-19k
3	2000-7000	2k-7k	16	15000-20000	15k-20-
4	3000-8000	3k-8k	17	16000-21000	16k-21k
5	4000-9000	4k-9k	18	17000-22000	17k-22k
6	5000-10000	5k-10k	19	18000-23000	18k-23k
7	6000-11000	6k-11k	20	19000-24000	19k-24k
8	7000-12000	7k-12k	21	20000-25000	20k-25k
9	8000-13000	8k-13k	22	21000-26000	21k-26k
10	9000-14000	9k-14k	23	22000-27000	22k-27k
11	10000-15000	10k-15k	24	23000-28000	23k-28k
12	11000-16000	11k-16k	25	24000-29000	24k-29k
13	12000-17000	12k-17k	26	25000-30000	25k-30k

5.2.1. Belousov-Zhabotinsky reaction

A MLP model predicting one step into the future is build for each segment specified in Table 5.1. Before the model is trained the architectural structure of the MLP is specified. Since the reconstructed phase space vectors of the system are used as inputs to the network the number of input nodes are equal to the embedding dimension, in this case $d_e = 3$. Since the time series considered in the simulated case studies consists of observations from only one variable the output layer also consists of only

one node to produce one output. To ensure that training and prediction time are reduced to a minimum only one layer of hidden nodes is selected. The number of nodes in this hidden layer is obtained by minimizing the Schwartz Information Criterion (SIC), see appendix A. The network is first trained for 3 hidden nodes for which the SIC is calculated. The number of hidden nodes is them increased by 1 and the network is retrained with a recalculation of the SIC. As soon as there is three consecutive iterations without an improvement of the SIC the number of hidden nodes that produced the last improved SIC is considered for the model. Figure 5.10 is a graphic illustration of the use of the SIC to determine the number of hidden nodes (*NHN*).

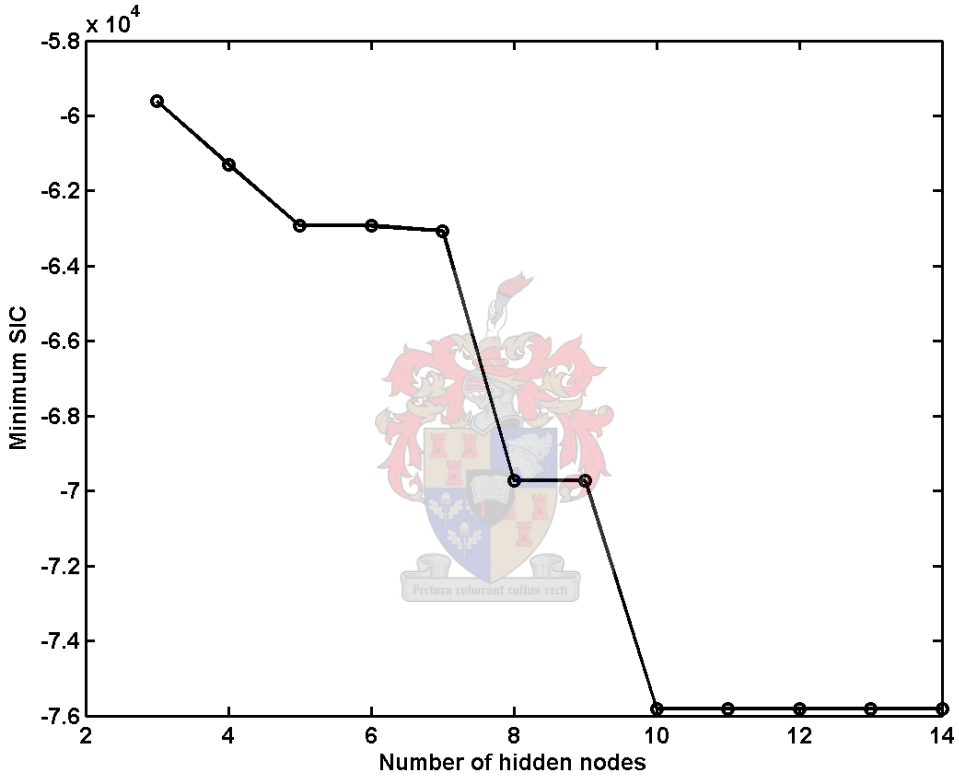


Figure 5.10: SIC for number of hidden nodes for the 26th segment of the BZ reaction data.

The same number of hidden nodes (10) is used for all 26 models build to reduce the variability of the models. Each of the 26 segments is divided into a training and validation set for training of the MLP to ensure good generalisation and that overfitting does not occur.

Now all the models are used to predict each of the 26 segments. Some of the results obtained from the 26th segment's model are illustrated in Figure 5.11 to Figure 5.16. There is a definite deterioration in the performance of this model when segments from before or during the non-stationarity period are predicted.

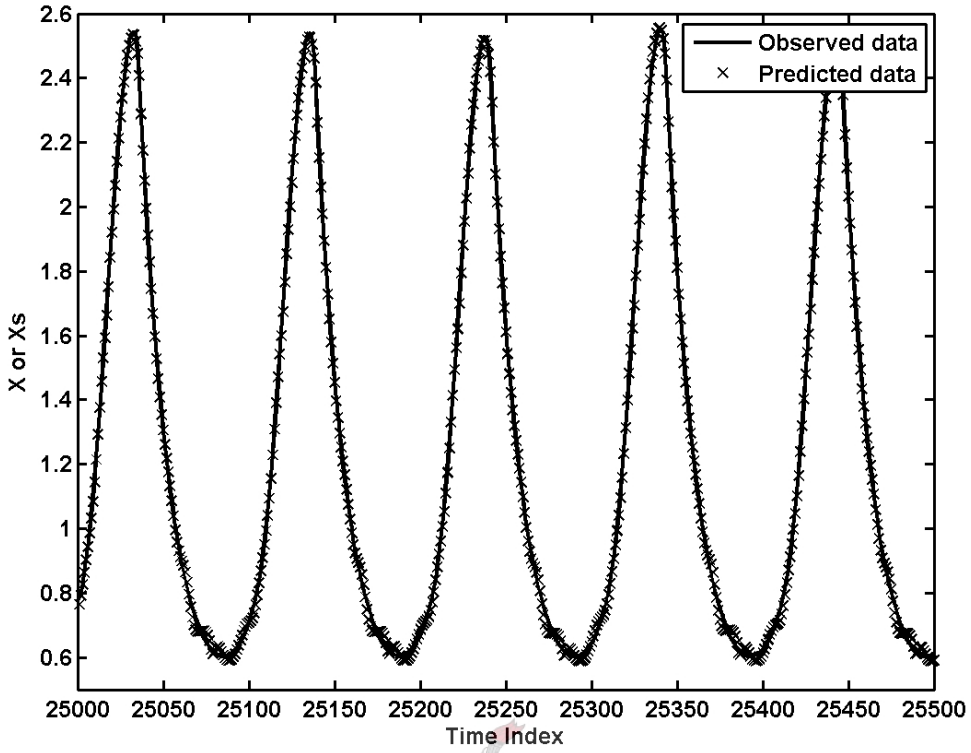


Figure 5.11: Acquired and actual time series obtained from one step prediction of 26th segment with 26th segment's model.

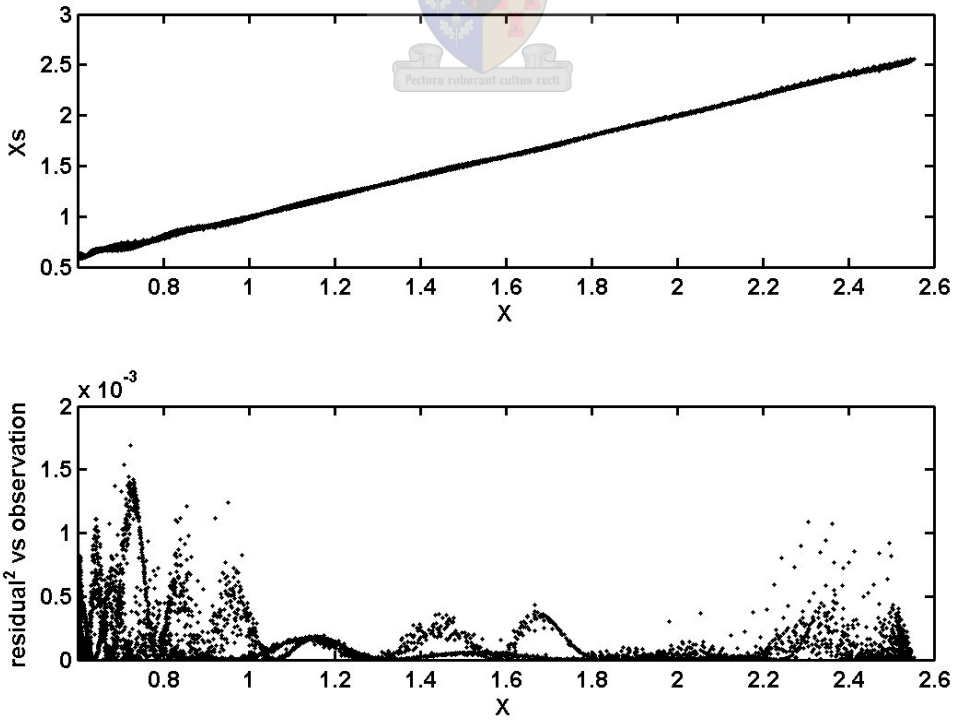


Figure 5.12: Performance of one step prediction of 26th segment with 26th segment's model.

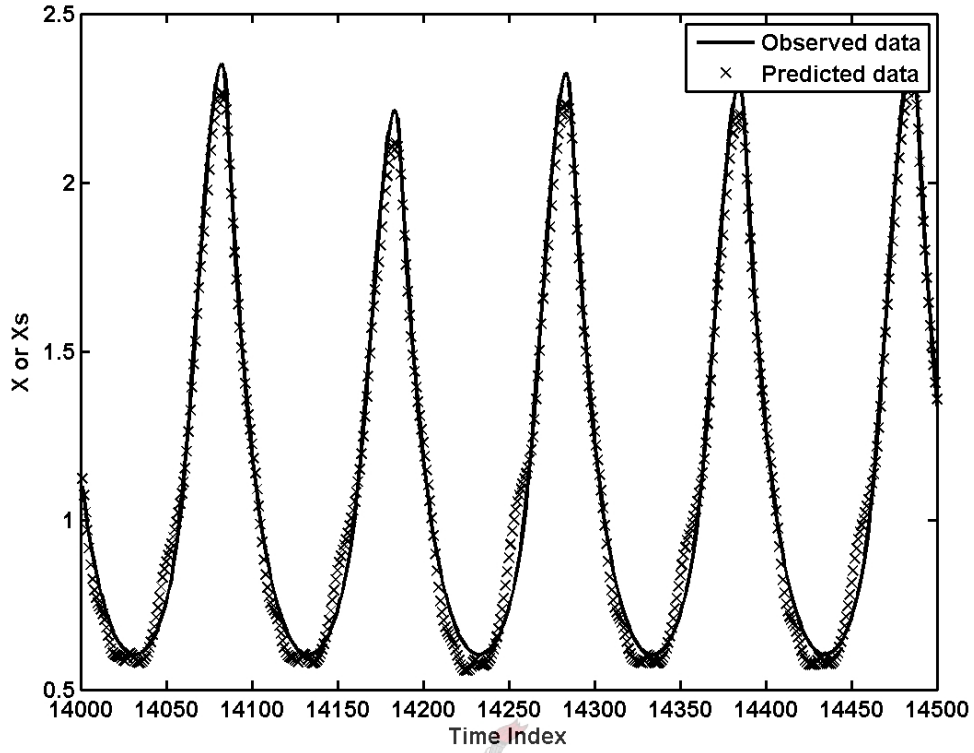


Figure 5.13: Acquired and actual time series obtained from one step prediction of 14th segment with 26th segment's model.

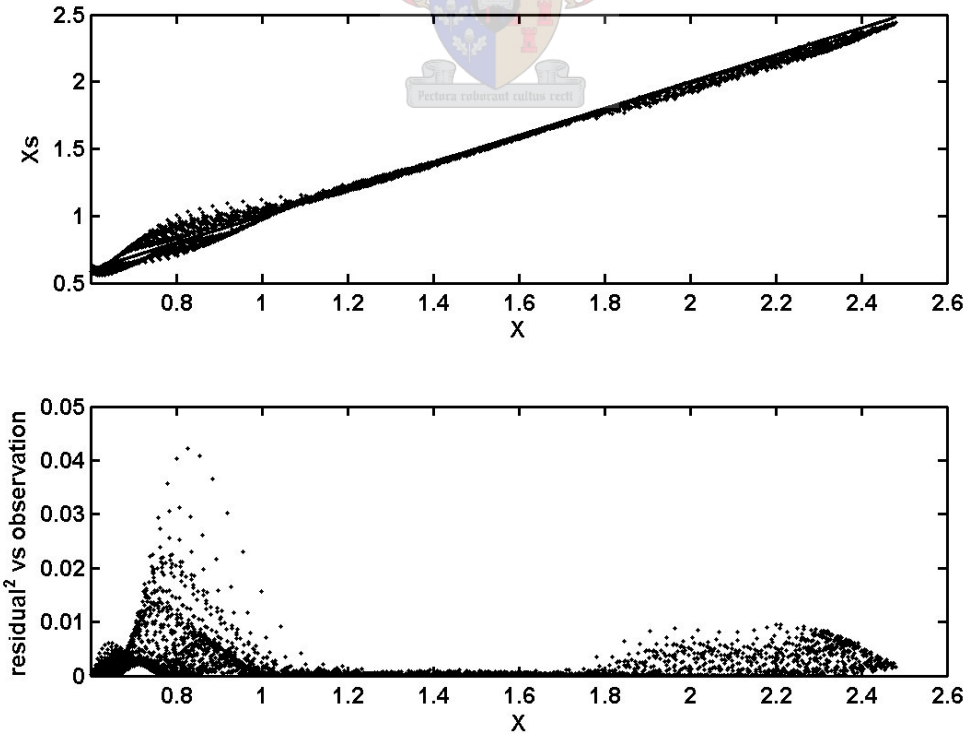


Figure 5.14: Performance of one step prediction of 14th segment with 26th segment's model.

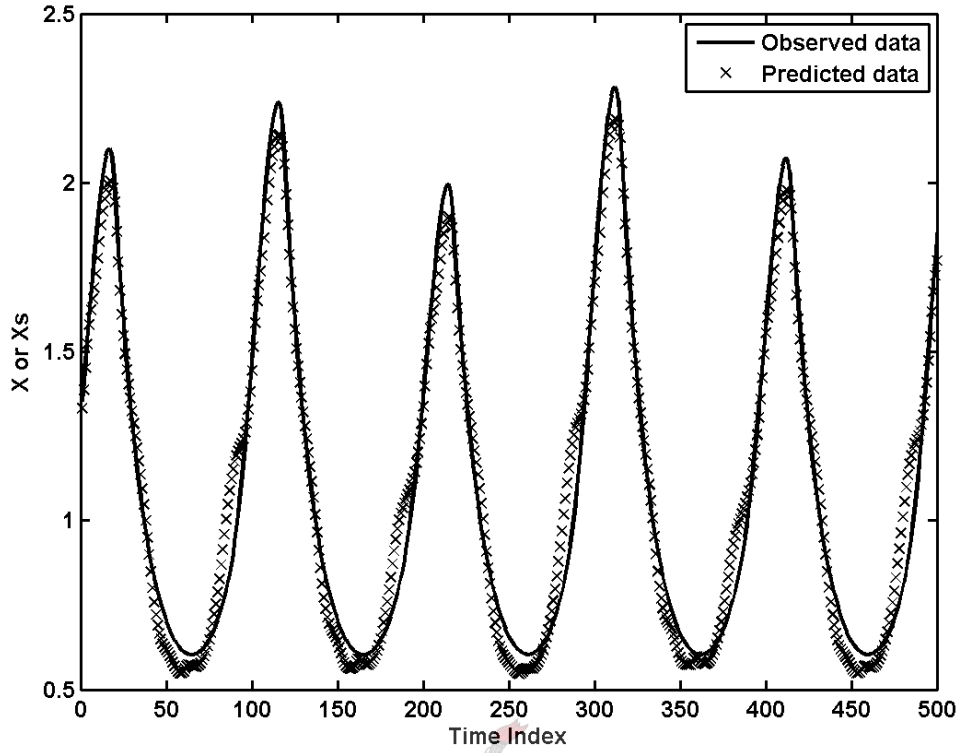


Figure 5.15: Acquired and actual time series obtained from one step prediction of 1st segment with 26th segment's model.

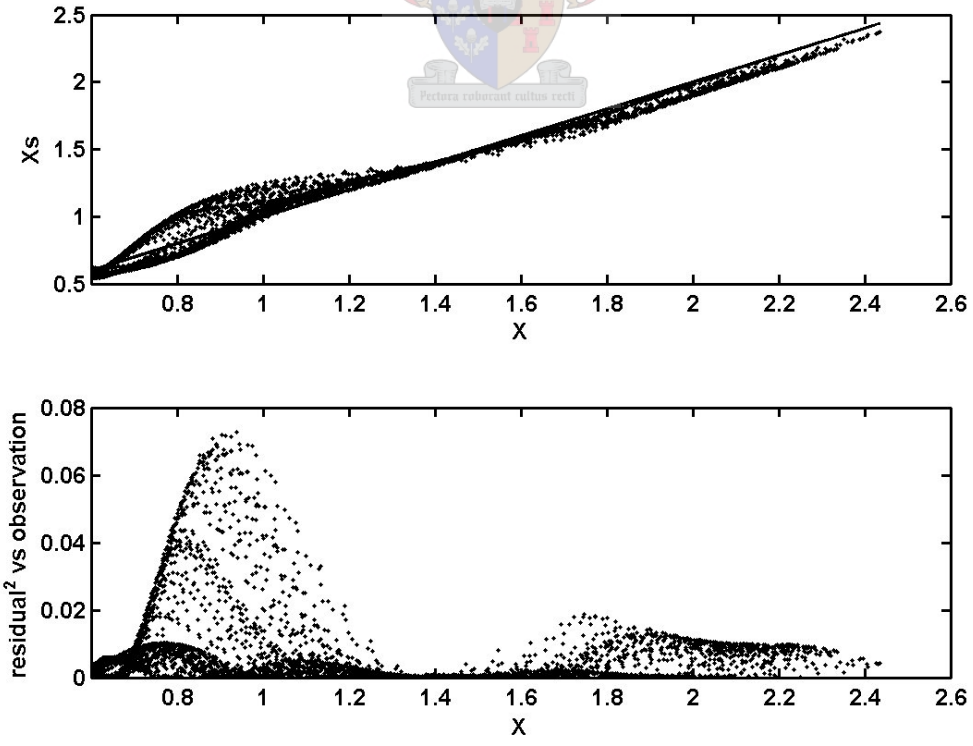


Figure 5.16: Performance of one step prediction of 1st segment with 26th segment's model.

Perhaps a better illustration of the performance of the 26th segment's model is Figure 5.17 that illustrates the MSE obtained for the whole range of segments. The 95% confidence limit is also indicated on Figure 5.17. The prediction MSE obtained for all the segments up to the 18th segment are above the 95% confidence limit. These segments (segment 1 to 18) are thus identified as having different dynamics from that of the last 8 segments.

A 2-dimensional colour map (Figure 5.18) is used to illustrate the results obtained when the whole range of segment models are used to predict the whole range of data segments, with the MSE as the measure to evaluate the prediction error.

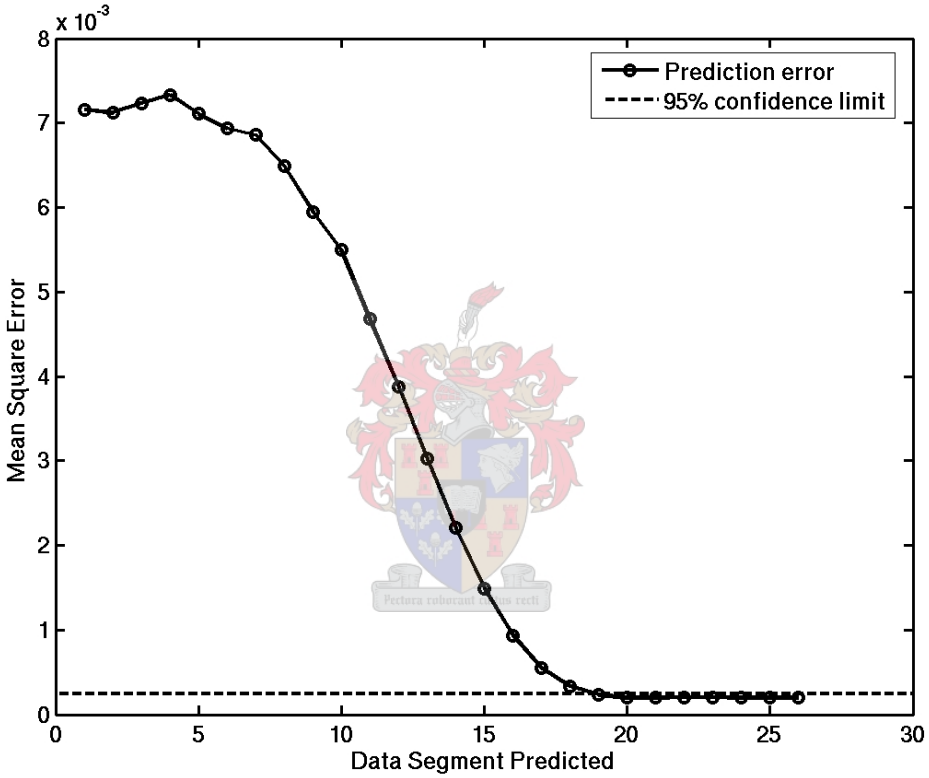


Figure 5.17: Prediction MSE of all 26 segments with 26th segment's model, the 95% confidence limit obtained from segments 21 to 26 (stationary part) is also indicated.

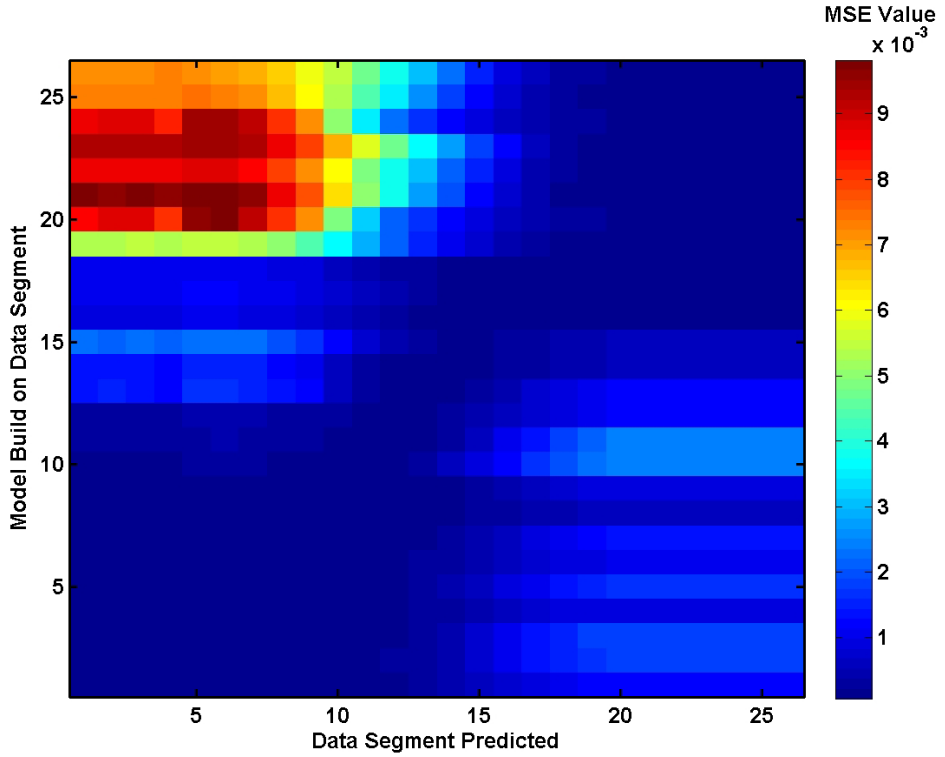


Figure 5.18: Mutual cross prediction with MSE for BZ reaction data ($T = 41,\ d_e = 3,\ NHN = 10$).

5.2.2. Autocatalytic process

The same procedure was followed as in section 5.2.1 with modified parameters summarized in the caption of Figure 5.19.

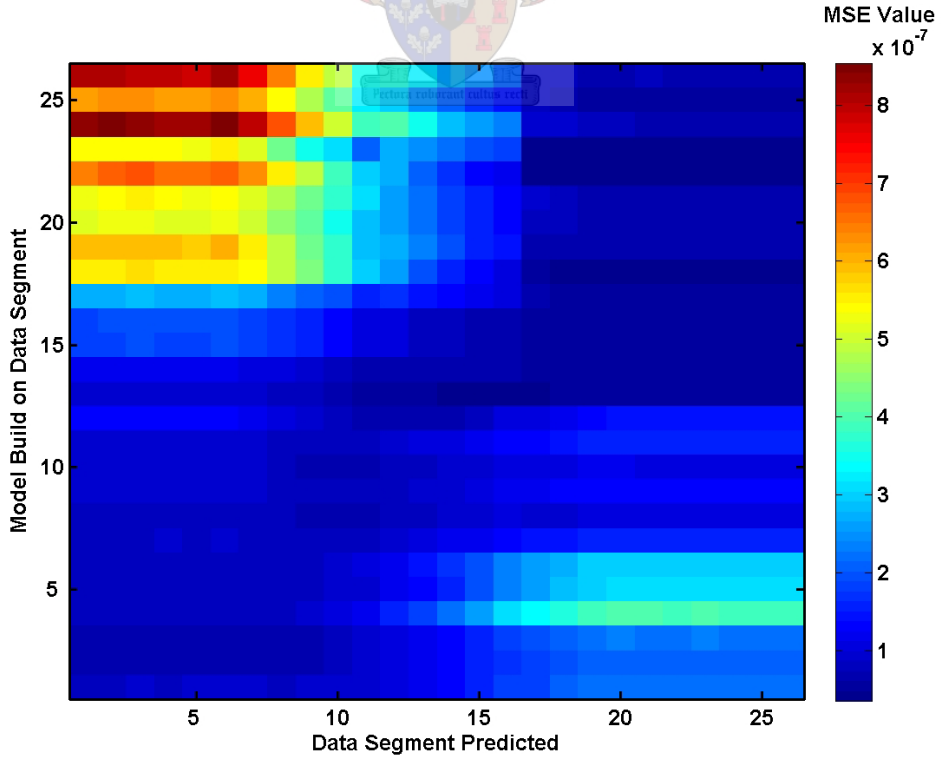


Figure 5.19: Mutual cross prediction with MSE for autocatalytic process data ($T = 17,\ d_e = 3,\ NHN = 10$).

5.2.3. Lotka-Volterra predator prey model

The same procedure was followed as in section 5.2.1 with modified parameters summarized in the caption of Figure 5.20.

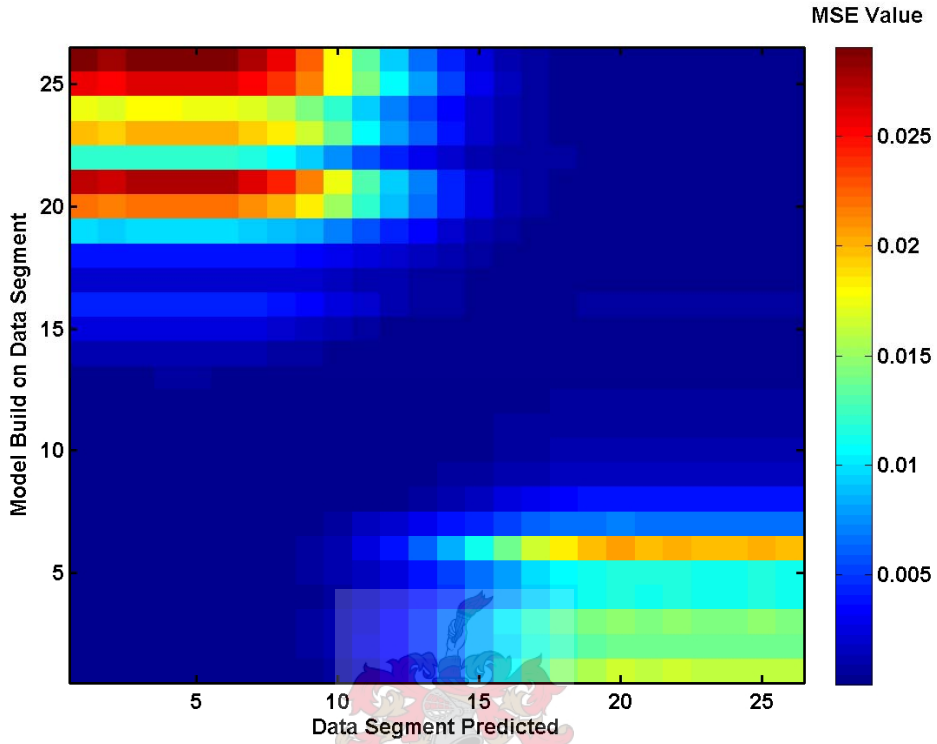


Figure 5.20: Mutual cross prediction with MSE for predator-prey data ($T = 74$, $d_e = 3$, $NHN = 10$).

5.2.4. Discussion of results obtained with mutual cross prediction

When examining the results from all three case studies, Figure 5.18 to Figure 5.20, a definite diagonal area where the lowest prediction errors are obtained can be identified. This is expected since the diagonal is the prediction of a segment on which the model was trained. The segments close to the diagonal also produce low prediction errors since these segments are still much similar to the segment the model was trained on. The performance of the models build from segments within the non-stationarity period, segments 11 to 16, does however deteriorate as segments further away from the diagonals are considered for prediction. This is due to the changing shape of the attractor in phase space as the dynamics of the system changes causing the MLP to generalise poorly.

Two definite square blocks, from segments 1 to 10 and 17 to 26 with similar low prediction errors, can also be identified in Figure 5.18 to Figure 5.20. The first 6 segments from the block defined by segments 1 to 10 are from the initial stationary part of the time series thus containing only stationarity data, hence the similar low prediction errors. In the remaining 3 segments of that block, segments 7 to 10, only a

small amount of non-stationary data are included and the parameters have only changed slightly by the time of this segment. Thus no real change in the prediction errors is observed for these segments predicted by the previous 6 segments' models or their models used to predict the previous 6 segments. A similar trend is observed for the block defined by segments 17 to 26, this time only the last 6 segments being that of the stationary part of the time series after the induced dynamic change.

The performance of a model continues to deteriorate when a non-stationarity in the form of a slow drift is encountered. This can be observed by studying the prediction errors of the first 6 segments in Figure 5.18 to Figure 5.20. An increase in the prediction error is observed from around the 11^{th} segment up to the 17^{th} segment from where on the prediction error stay reasonably constant again. This plateau is a result of a stationary part of the time series being reached, but with total different dynamics than that of the segment the model was build from. This explains the large prediction errors that are still present in predicting the last 6 segments with the first 6 segments' models even though they are all from the stationary parts of the time series.

In Figure 5.18 and Figure 5.19 it is also noticeable that the models from the first 6 segments predict the last 6 segments better than the last 6 segments predicting the first 6 segments. This can be explained by some parts of the attractor of the nonlinear last 6 segments being embedded into that of the chaotic attractor of the first 6 segments, but not the other way around as illustrated in Figure 5.21.

When the prediction errors of the first stationary part, segments 1 to 6 predicted by models from segments 1 to 6, is compared to that of the stationary part after the dynamic change, segments 21 to 26 predicted by models from segments 21 to 26, an decrease is observed (Figure 5.19). This is a good illustration of the limited predictability of a chaotic system, the first stationary part, compared to that of a nonlinear system, the second stationary part.

Generally the method of nonlinear cross prediction using the MSE of a MLP network model is a very successful method to test for dynamic changes. The decision rule was implemented by using the average MSE value obtained over the stationary parts (first 6 or last 6 segments) for a particular segment. The number of non-stationary data points required to detect a dynamic change were 3000 data points for the predator prey system, 5000 data points for the autocatalytic process and 4000 data points for the BZ reaction. The method does however require long computational periods due to model training and cross prediction, which is calculations on additional data sets, and isn't ideally suited for online monitoring.

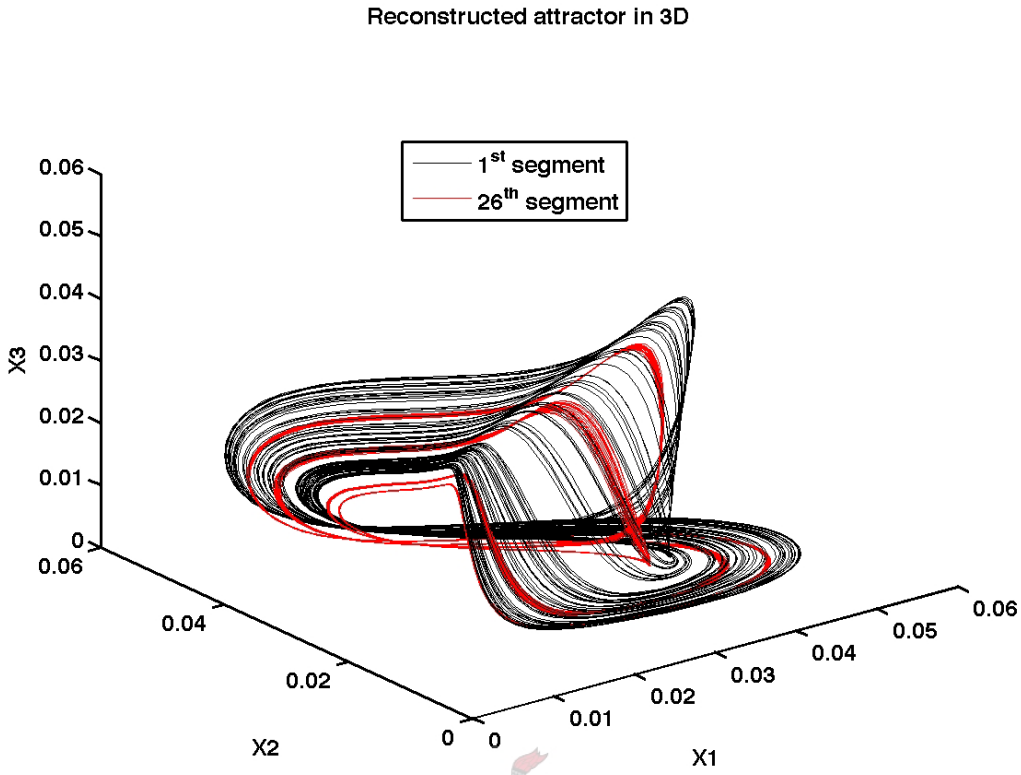


Figure 5.21: Some parts of 26th segment's attractor embedded in 1st segment's attractor.

5.2.5. Effect of noise on nonlinear cross prediction

Even with the added noise levels the method of nonlinear cross prediction is still able to identify the dynamic changes in the observed time series from the three simulated systems (Figure 5.22). Only in the case of the autocatalytic process is additional non-stationary data points (now 6000 data points) required to detect a dynamic change. The MSE does however increase for the prediction of all the segments as the level of noise increases, as can be expected since noise adds a degree of unpredictability.

The method is however not noise proof and at some arbitrary high level of noise it will be impossible to extract useful information from the attractors of the systems. This will cause only random noise to be modelled preventing any changes to be resolved.

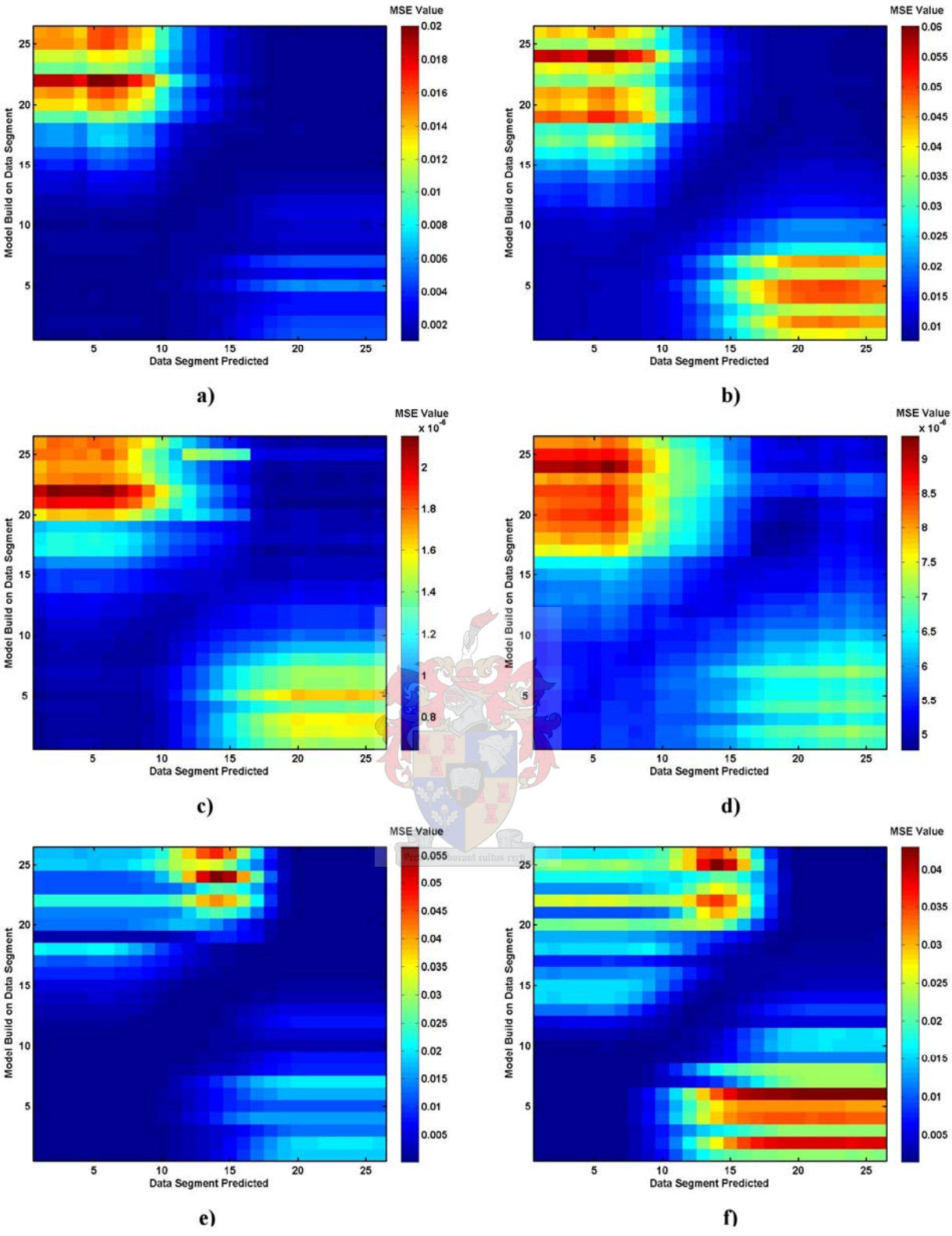


Figure 5.22: Summary of mutual cross prediction with MSE **a)** BZ reaction data (5% noise) **b)** BZ reaction data (15% noise) **c)** autocatalytic process data (5% noise) **d)** autocatalytic process data (15% noise) **e)** predator-prey data (5% noise) **f)** predator-prey data (15% noise).

5.3. Detecting dynamic change with correlation dimension as test statistic

The correlation dimension is a statistic defined in phase space which characterises the topological structure of an attractor. It characterises the topological structure of the attractor by the inter point distance distribution of the points defining an attractor over a range of length scales. Under constant system specifications it is also an invariant measure and is expected to change once a dynamic change in the system parameters is present. The algorithm proposed by Judd (1992) is used to estimate the correlation dimension in this section and is expressed over a range of length scales. The shape and region of the correlation dimension curve is considered when evaluating change in the correlation dimension. The observed time series from the simulated case studies is again divided into different segments as specified in Table 5.1.

5.3.1. Belousov-Zhabotinsky reaction

It is important to note that the same phase space reconstruction is used as for the classification of the time series in section 4.1. In the case of the BZ reaction the data is embedded with a time delay $T = 41$ and an embedding dimension of $d_e = 3$.

Now the first step in estimating the correlation dimension is evaluating the correlation function, equation 2.26. This function evaluates the number of data points within a certain distance ε from each other for a range of inter point distances. This correlation sum is expressed on a log-log axis as a function of inter point distances ε as illustrated in Figure 5.23a. When examining the slopes of the three curves enlarged in Figure 5.23b a steeper gradient is observed for the 1-10 000 data points segment than the other two segments. These differences in the gradients of the correlation sum curves already indicate the possibility of a change in dynamics. A higher correlation dimension is thus expected for the 1-1000 data points segment which is obtained in Figure 5.23c by fitting a polynomial of the order of the topological dimension to the curves, as specified by the Judd (1992) algorithm.

Now in Figure 5.23c three correlation dimensions are obtained, and there is a definite difference between the first two curves, 1-10 000 and 10 001-20 000 data points, and the last curve, 20 001-30 000 data points. The dynamics of the system defined by the third curve, 20 001-30 000 data points, are thus different from those of the first two curves due to some parametric change. The problem now is whether the difference between the first two curves is of significant level to deduce that the dynamics of the systems defining these curves are different. To overcome this problem a decision rule is created by providing 95% confidence limits on the correlation dimension for the stationary parts of the monitored time series. This is done by first simulating a calibration data set (Figure 3.9), with the same parameters as the initial conditions of

the monitored time series, sufficiently long to obtain 20 correlation dimension estimates from non-overlapping segments. A similar calibration data set is also simulated for the parameters at the end conditions of the monitored time series. The 95% confidence limits are then determined for the correlation dimension estimates from these 20 segments, as illustrated in Figure 5.24.

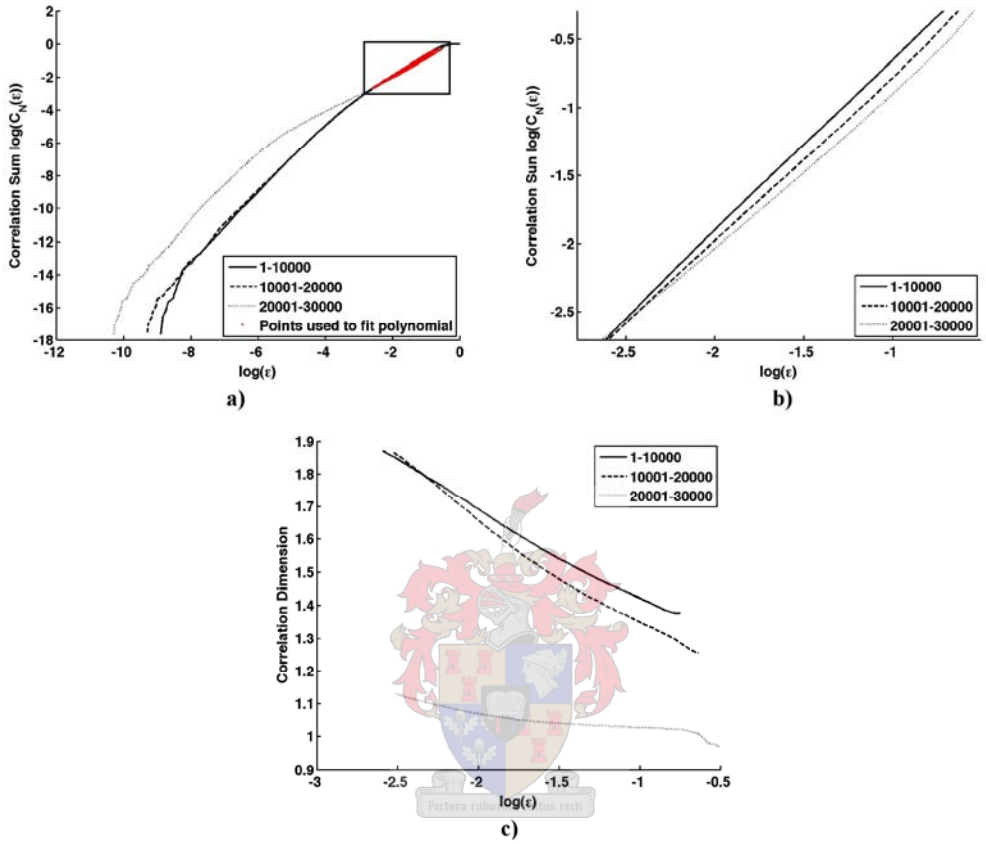


Figure 5.23: Summary correlation dimension estimate **a)** Correlation sum of the BZ reaction data **b)** Enlargement of region defined by rectangle in (a) **c)** Correlation dimension for correlation sum curves in (a).

The monitored time series is now analysed by determining the correlation dimension for each of the segments defined in Table 5.1. The results are plotted together with the mean and the 95% confidence interval limits of the two stationary parts of the time series, as illustrated in Figure 5.25.

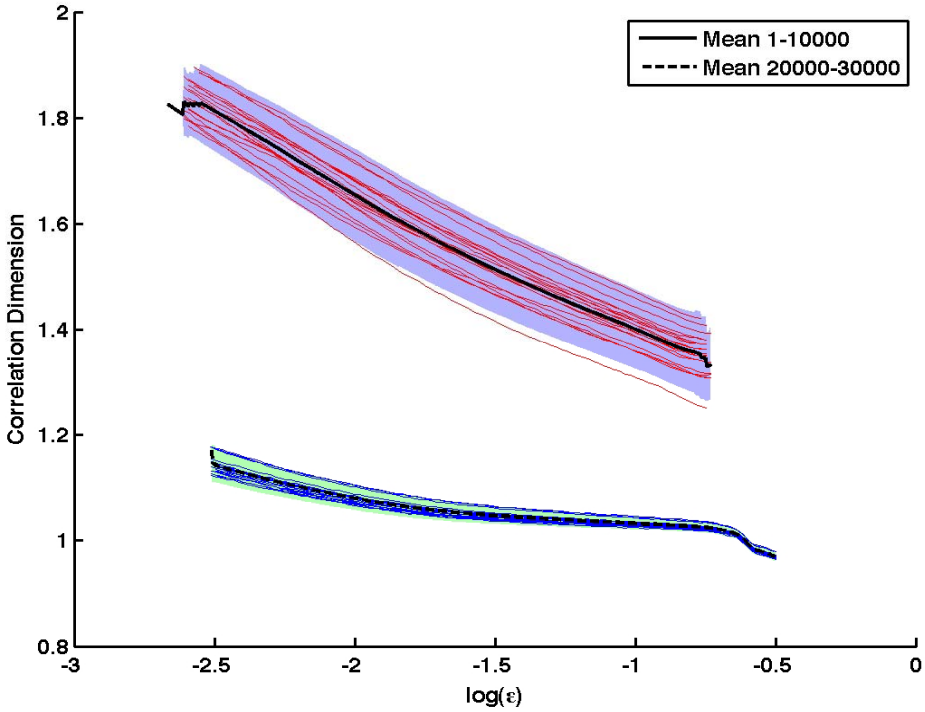


Figure 5.24: 95% limits of the correlation dimension for the stationary parts of the BZ reaction data.

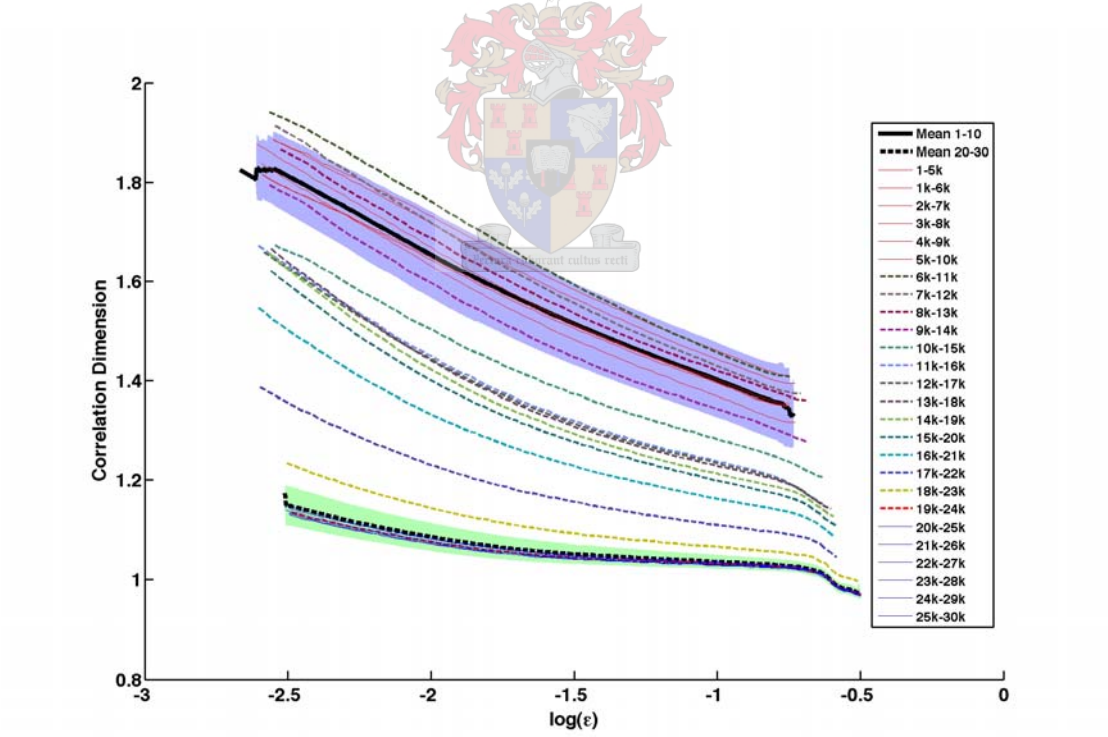


Figure 5.25: Correlation dimension estimates of the BZ reaction data by the moving widow approach with the mean and 95% confidence intervals of the stationary parts ($T = 41,\ d_e = 3$).

5.3.2. Autocatalytic process

The same procedure was followed as in section 5.3.1 with modified parameters summarized in the caption of Figure 5.26.

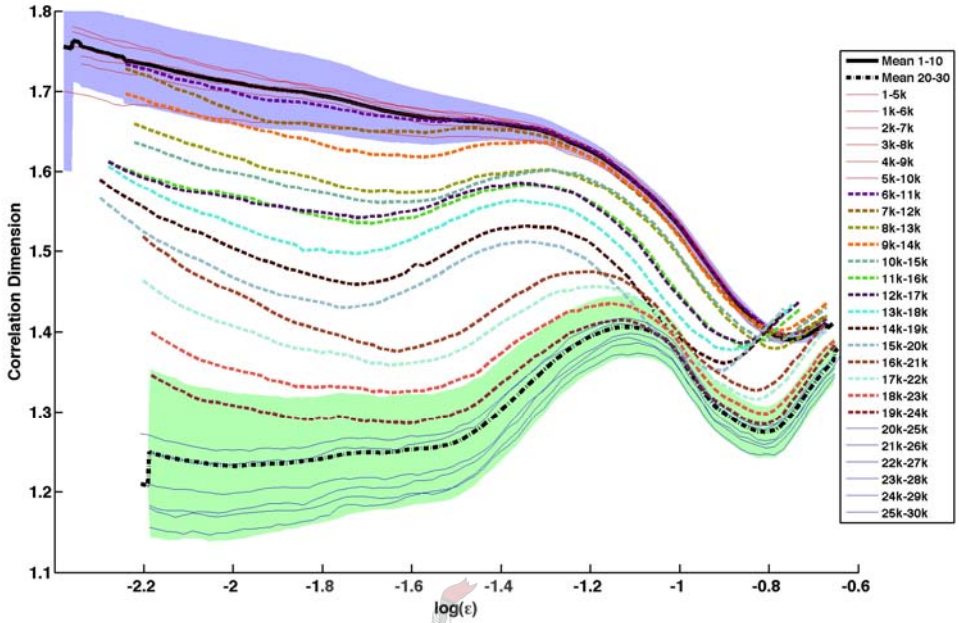


Figure 5.26: Correlation dimension estimates of the autocatalytic process data by the moving widow approach with the mean and 95% confidence intervals of the stationary parts ($T = 17,\ d_e = 3$).

5.3.3. Lotka-Volterra predator prey model

The same procedure was followed as in section 5.3.1 with modified parameters summarized in the caption of Figure 5.27.

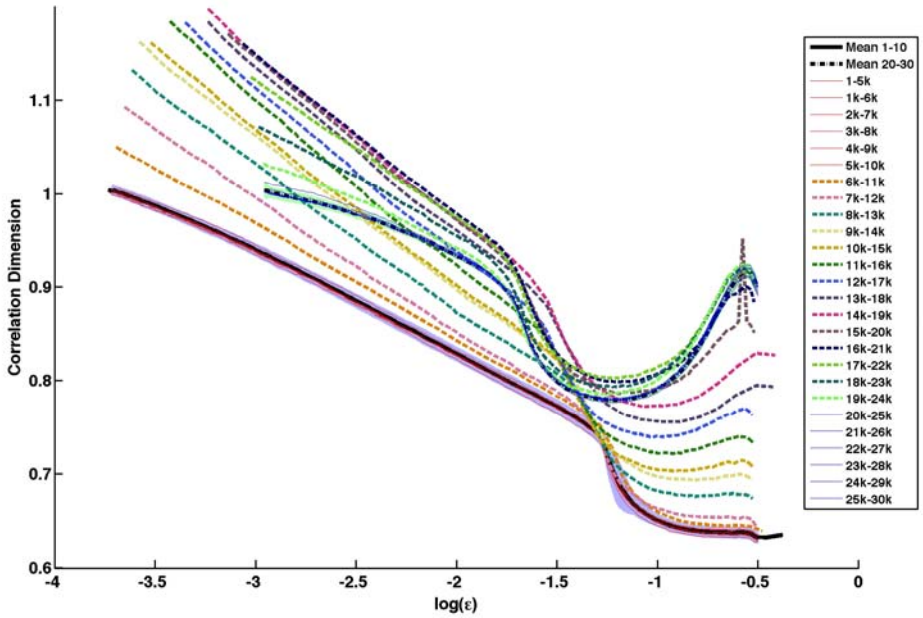


Figure 5.27: Correlation dimension estimates of the predator-prey data by the moving widow approach with the mean and 95% confidence intervals of the stationary parts ($T = 74,\ d_e = 3$).

5.3.4. Discussion of results obtained with correlation dimension as test statistic

The correlation dimension is very successful in identifying the dynamic changes in all three the simulated case studies. Generally there is an increase in the correlation dimension as a change is encountered. This is due the extra variability of the system because of the changing parameters effectively becoming "variables" and increasing the dimensionality of the attractor. This effect is probably best observed in the result from predator-prey data, Figure 5.27. A significant increase, to outside the 95% confidence limits, of the correlation dimension is observed for the very first segment containing non-stationary data (segment 6k-11k). Then a decrease in the correlation dimension can also be observed as the system becomes stable again and more stationary data are included in the segments (segments 17k-22k to 20k-25k).

The increase in the correlation dimension might not be observed in the autocatalytic process, Figure 5.26, due to the over shadowing effect of moving from a chaotic state of the system to a normal nonlinear state. This causes the tremendous decrease in the correlation dimension observed in the cases of the BZ reaction (Figure 5.25) and autocatalytic process (Figure 5.26). The chaotic behaviour observed at the initial states of the BZ reaction and autocatalytic process cause larger variations of the correlation dimension estimates due to limited amount of data used per estimation. Larger confidence limits are therefore required explaining why the change in dynamics can only be identified after about 3000 to 5000 non-stationarity data points.

In all three cases a difference in the correlation dimension between the stationary parts of the time series can be also observed. This enables the correlation dimension to identify instantaneous dynamic changes as well, even if non-overlapping windows were used to divide the data into segments.

5.3.5. Effect of noise on the correlation dimension as test statistic

The correlation dimension does suffer in the presence noise. A general increase in the statistic can be expected as the noise level increases (Figure 5.28 to Figure 5.33). Noise increases the complexity of an attractor, which explains the increase of the correlation dimension. This however does not prevent the correlation dimension from recovering some structure of the attractors and resolving the dynamic changes in the cases of the BZ reaction and autocatalytic process (Figure 5.28 to Figure 5.31). The response time increases slightly and ranges from about 1000 to 5000 non-stationary data points required to resolve a dynamic change.

The predator-prey system suffers more severely from the added noise. At both levels of added noise the stationary parts are overlain and the statistic won't be able to distinguish between the different stationary parts. The correlation dimension does

however show a significant increase for estimations of the non-stationary part of the data and will still be able to resolve some dynamic changes (Figure 5.32 and Figure 5.33).

Eventually the presence of an arbitrary high level of noise will cause the statistic to be unable to identify any structure or geometric properties of the attractor. Thus no changes will be detected and similar results for time series and random (surrogate) data can be expected.

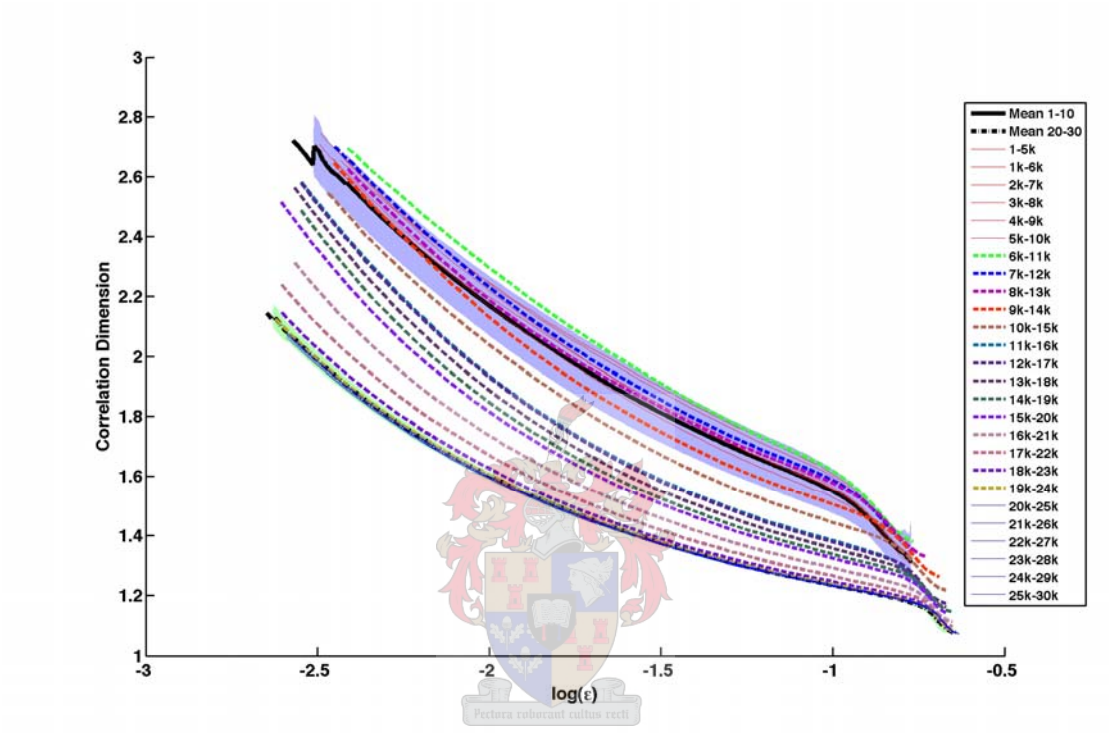


Figure 5.28: Correlation dimension estimates of the noisy BZ reaction data (5% noise) by the moving window approach with the mean and 95% confidence intervals of the stationary parts.

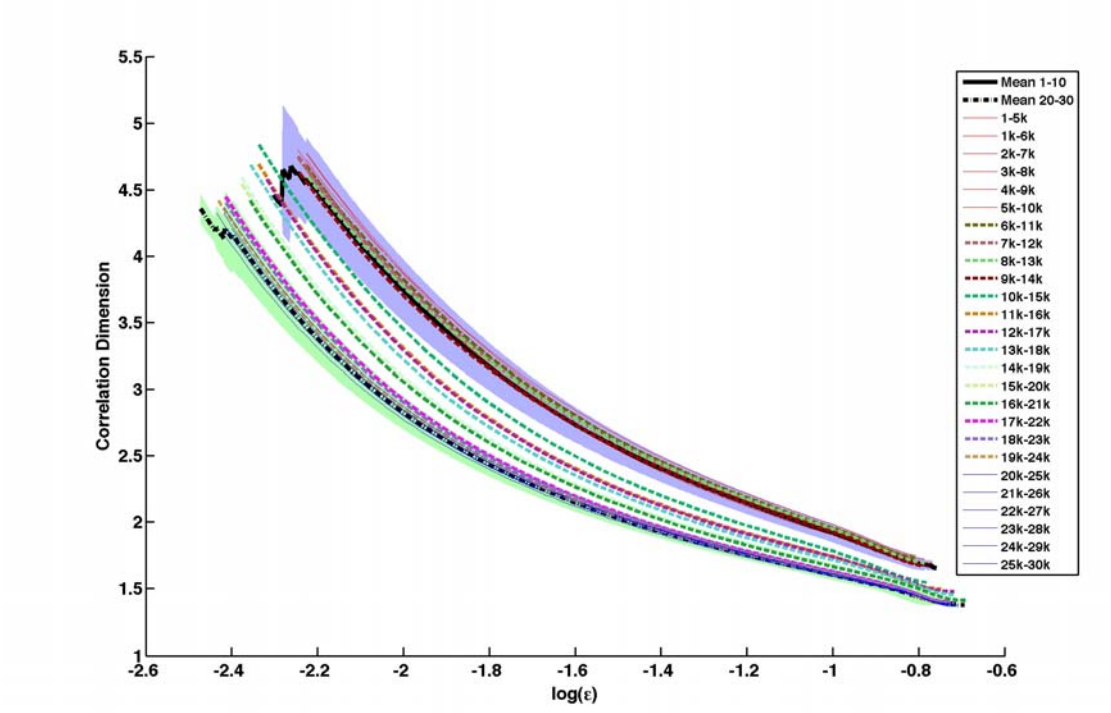


Figure 5.29: Correlation dimension estimates of the noisy BZ reaction data (15% noise) by the moving window approach with the mean and 95% confidence intervals of the stationary parts.

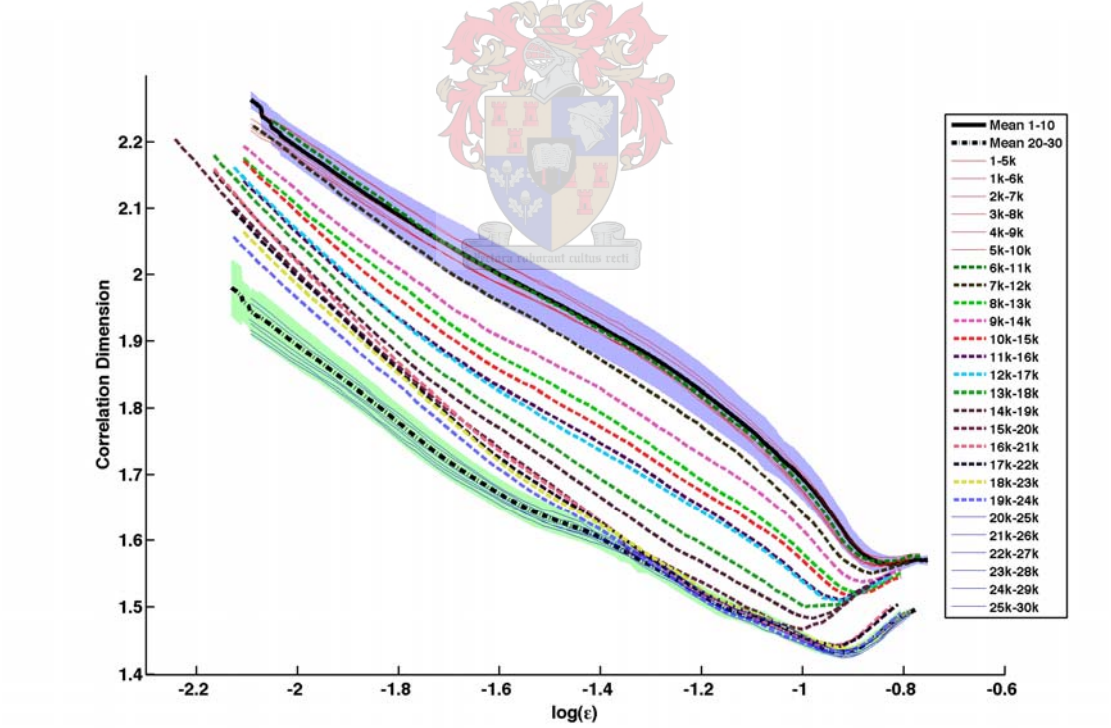


Figure 5.30: Correlation dimension estimates of the noisy autocatalytic process data (5% noise) by the moving window approach with the mean and 95% confidence intervals of the stationary parts.

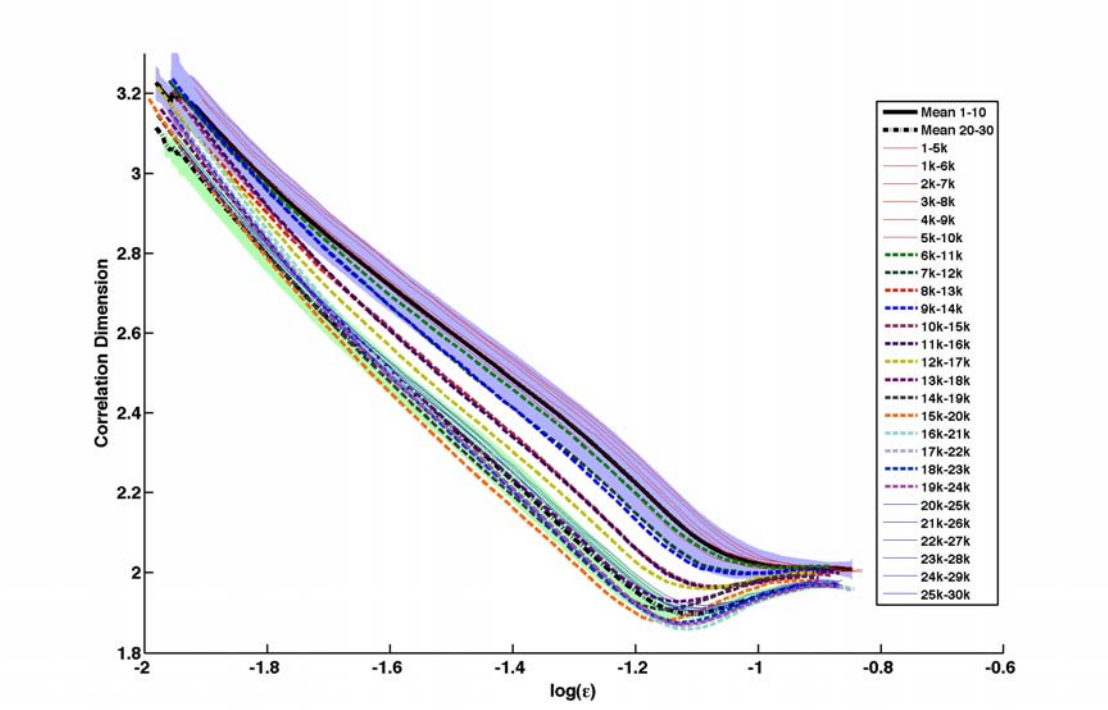


Figure 5.31: Correlation dimension estimates of the noisy autocatalytic process data (15% noise) by the moving window approach with the mean and 95% confidence intervals of the stationary parts.

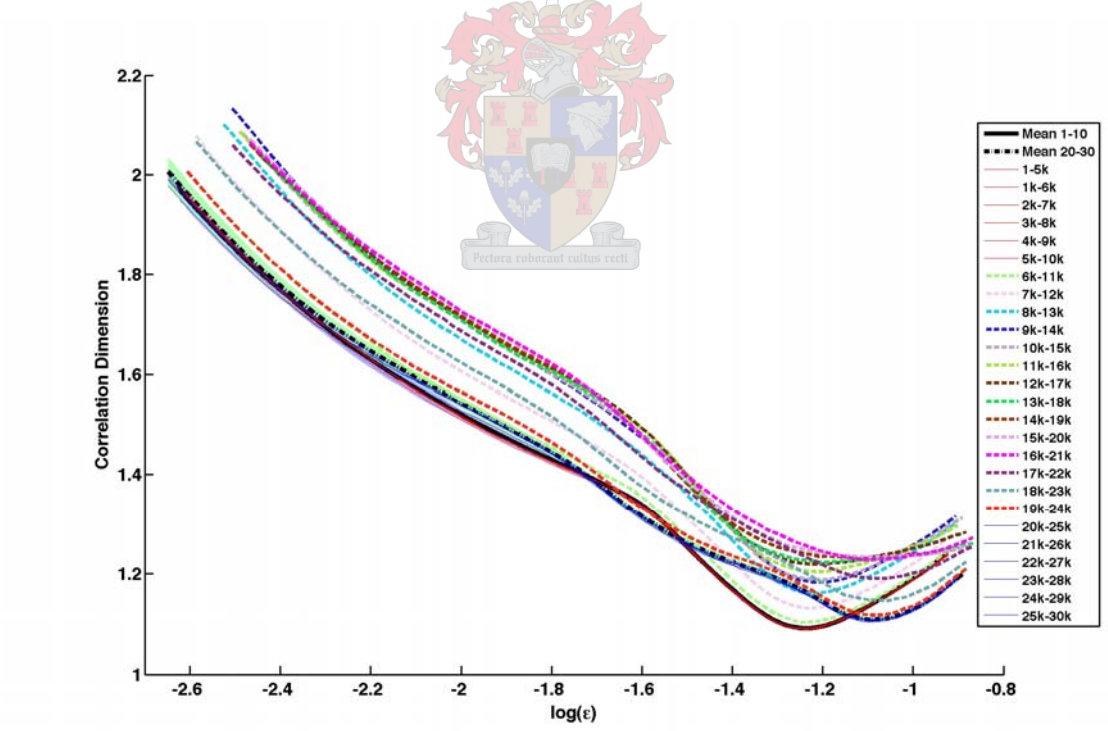


Figure 5.32: Correlation dimension estimates of the noisy predator-prey data (5% noise) by the moving window approach with the mean and 95% confidence intervals of the stationary parts.

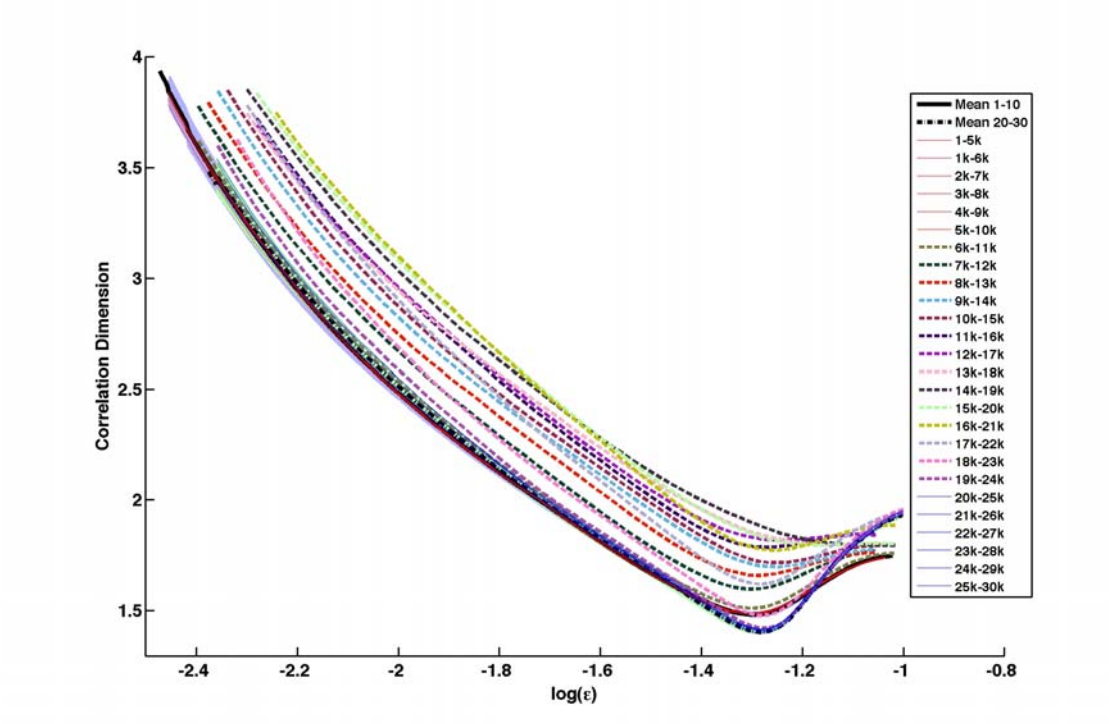


Figure 5.33: Correlation dimension estimates of the noisy predator-prey data (15% noise) by the moving window approach with the mean and 95% confidence intervals of the stationary parts.

5.4. Summary of results obtained from simulated case studies

Table 5.2 : Summary of results obtained.

System	Amount of Noise	Change-Point Algorithm		Mutual Cross Prediction		Correlation Dimension	
		Points*	False Alarms	Points*	False Alarms	Points*	False Alarms
BZ Reaction	No Noise	1000	5	4000	-	4000	-
	5% Noise	1000	14	4000	-	4000	-
	15% Noise	3000	1	4000	-	5000	-
Autocatalytic Process	No Noise	6000	4	5000	-	3000	-
	5% Noise	1000	11	5000	-	2000	-
	15% Noise	no change detected		6000	-	5000	-
Predator-prey Model	No Noise	1000	-	3000	-	1000	-
	5% Noise	4000	-	3000	-	1000	-
	15% Noise	3000	1	2000	-	1000	-

Points* - Number of points after the change required by the algorithm to detect a change in the system.

6. Detecting dynamic change in electrochemical noise data

In this chapter the spontaneously generated current and potential signals produced by a freely corroding system are investigated. These current and potential fluctuations are known as electrochemical noise. The data are obtained from a laboratory experiment but the details of the experiment and procedure are not considered in this study, only the analysis of the signals.

Since the data obtained from the experiments are in the form of three separate signals only the correlation dimension as test statistic is used to detect dynamic changes between the different signals. The implementation of the change point detection algorithm that was used doesn't allow three separate data sets as inputs. Combining the three data sets will cause unnatural jumps at the points where the signals were joined, which will easily be recognised by the algorithm, but won't be a true representation of the algorithm's performance. Although mutual cross prediction is not considered in this study, it might be interesting to test the performance of the algorithm with similar data in future studies. In this section referring to the correlation dimension will again refer to the correlation dimension curve as a function of scale, the shape and the space it occupies, produced by Judd's (1992) algorithm.

6.1. Background and experimental setup

Electrochemical noise has long been studied as a mean to detect localized corrosion phenomena such as pitting, cervice corrosion and cavity attack. The spontaneous apparently random corrosion current is related to the kinetics of the reaction, while the corrosion potential is related to the process thermodynamics.

Legat et al. (1998) did a study on various types of corrosion processes and analysed the electrochemical noise by means of the chaos theory. They found that the chaotic analysis was very useful for determining the type of corrosion. García et al. (2003) also concluded that during pitting corrosion of steel in a NaCl solution the electrochemical noise signals are of chaotic nature and that chaotic analysis provided more data concerning the type of corrosion.

In this study the aqueous corrosion of austenitic stainless steel is considered. The experimental setup (Figure 6.1) consists of a corrosion cell containing 500 *ml* of the corrosive solution. Connected to the cell is a voltammograph (CV-27) connected to a Hewlett-Packard 34970A data acquisition unit for output to a computer. The electrochemical current is measured at 500 ms intervals with a three-sensor configuration. The current is measured between the two identical working electrodes consisting of austenitic stainless steel 304 strips, which were polished and degreased in hexane prior to use.

The two corrosive solutions considered in this study are distilled water (H_2O) and hydrochloric acid (HCl). The current measurements obtained from the experiments are illustrated in Figure 6.2 to Figure 6.4.

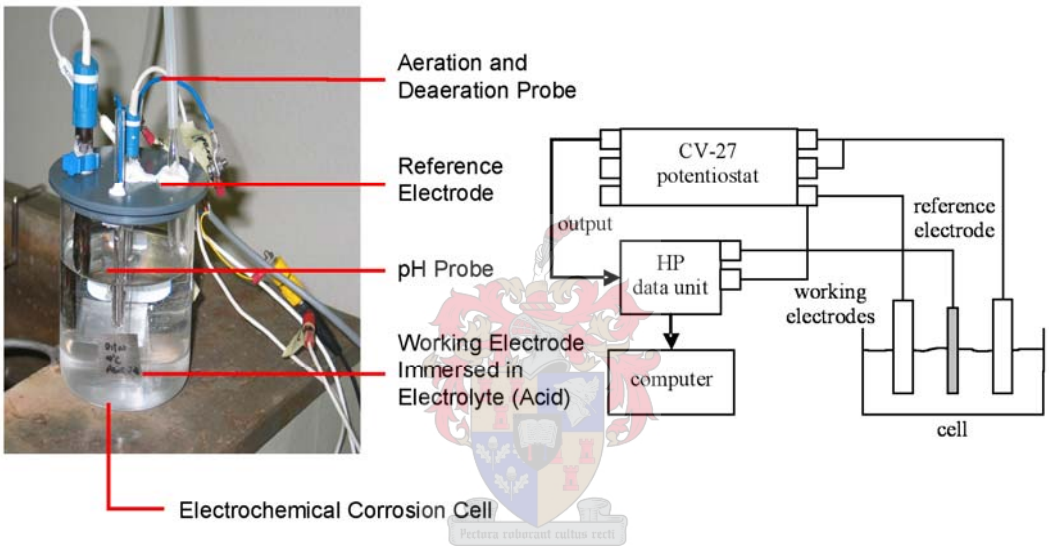


Figure 6.1: Experimental setup of corrosion cell.

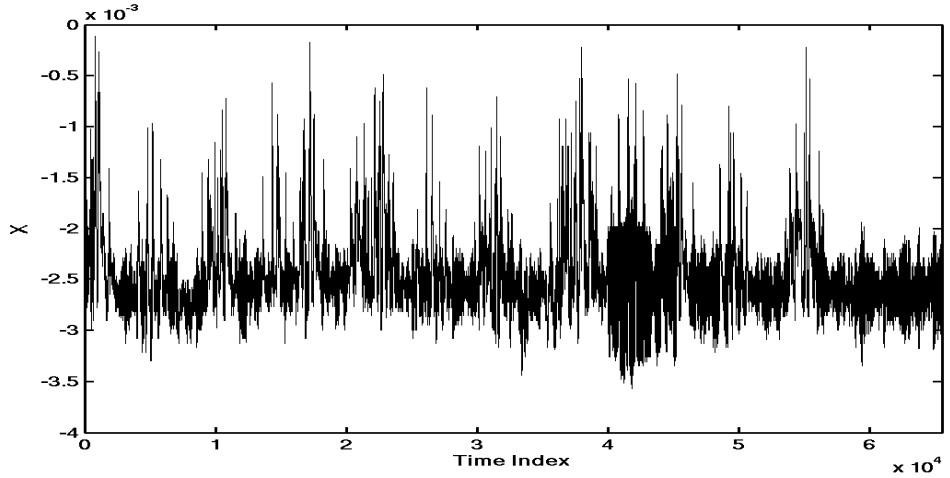


Figure 6.2: Current measurements obtained from corrosion cell with HCl (Run 18) as corrosive solution.

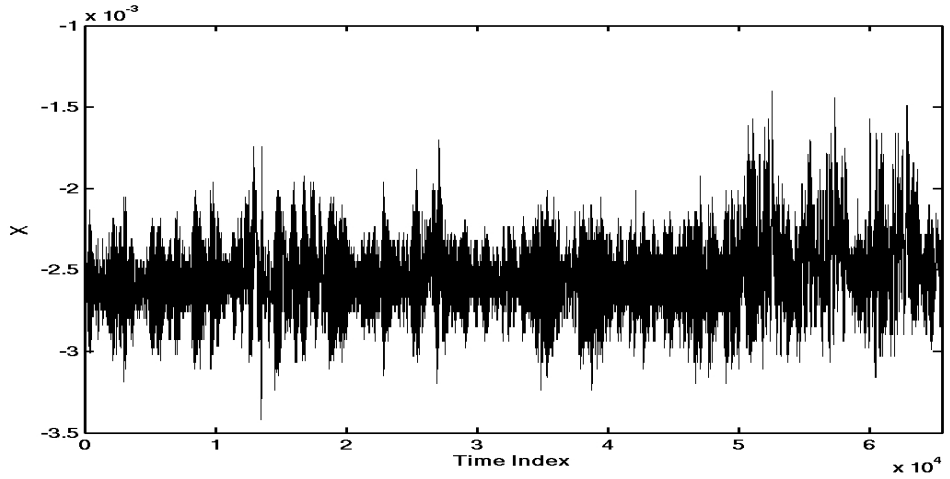


Figure 6.3: Current measurements obtained from corrosion cell with H_2O (Run 7) as corrosive solution.

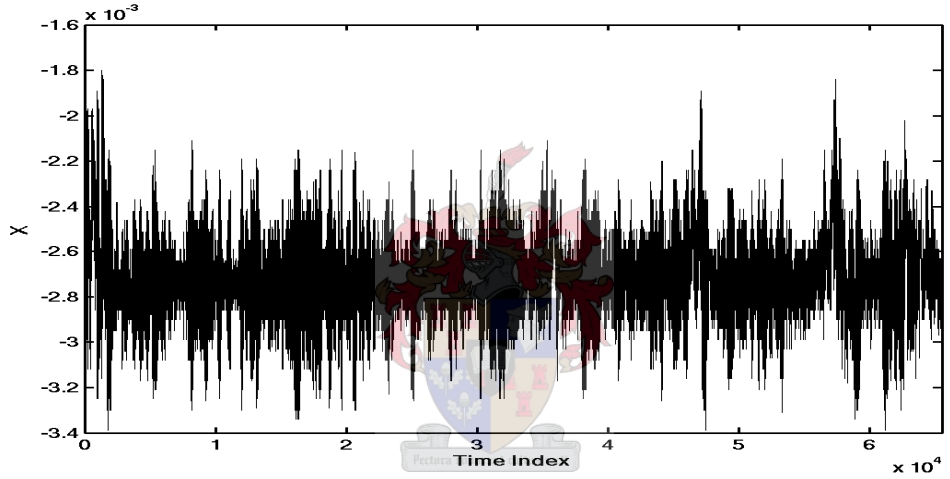


Figure 6.4: Current measurements obtained from corrosion cell with H_2O (Run17b) as corrosive solution.

6.2. Phase space reconstruction

In order to perform surrogate data analysis on the acquired signals phase space reconstruction of the data are required. This is done by time delay embedding for which two parameters are required, a time delay T and embedding dimension d_e. Since the test statistic, the correlation dimension, is relatively invariant under different phase space reconstructions, all 3 time series are embedded with the same embedding parameters. The normal process, Run 18, against which the other runs are tested is considered to estimate the embedding parameters.

Now after considering the first zero of the autocorrelation function at $T = 7$ (Figure 6.5) and the first minimum of the average mutual information at $T = 9$ (Figure 6.6), a time delay of $T = 9$ is selected. Similar results, T between 7 and 9, are

obtained for the other two time series, Run 7 and Run 17. Thus the selected time delay of $T = 9$ should be adequate for the reconstruction of all three time series.

The embedding dimension is obtained by the false nearest neighbours algorithm (Figure 6.7), which suggests that an embedding dimension of $d_e = 8$ should be sufficient. Similar results are also obtained for Run 7 and Run 17 implying that all the dynamics of the three systems should be captured in the phase space reconstructions of Figure 6.8.

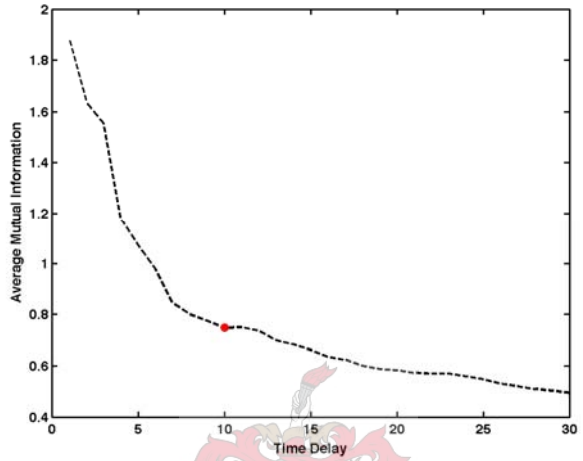


Figure 6.5: Autocorrelation function of electrochemical noise (Run 18).

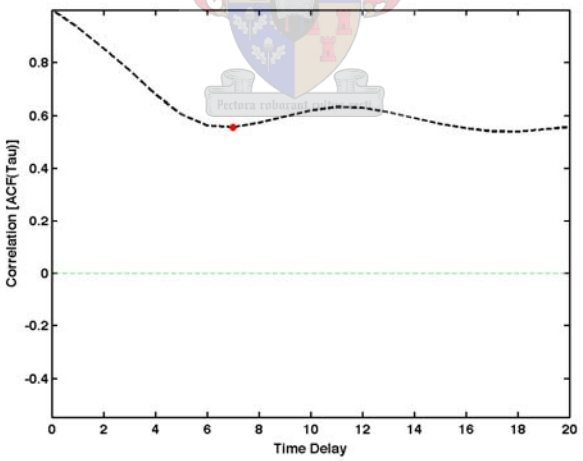


Figure 6.6: Average mutual information of electrochemical noise (Run 18).

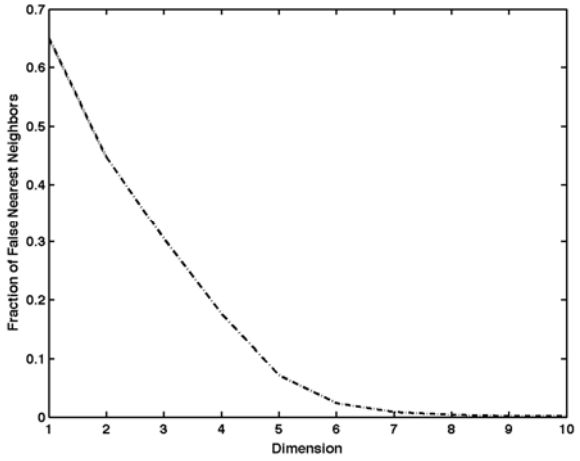
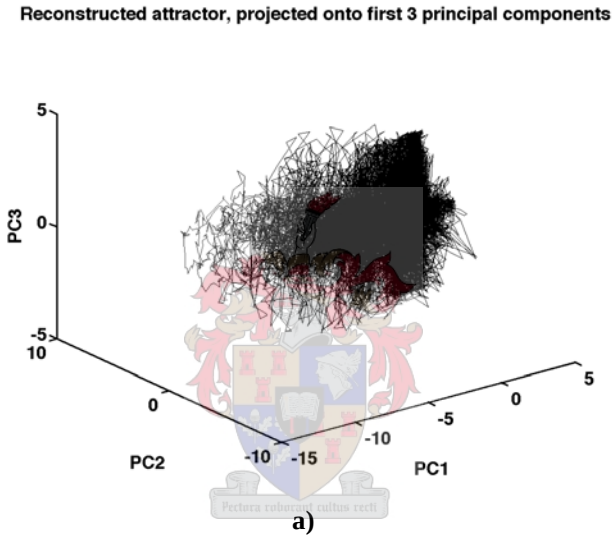
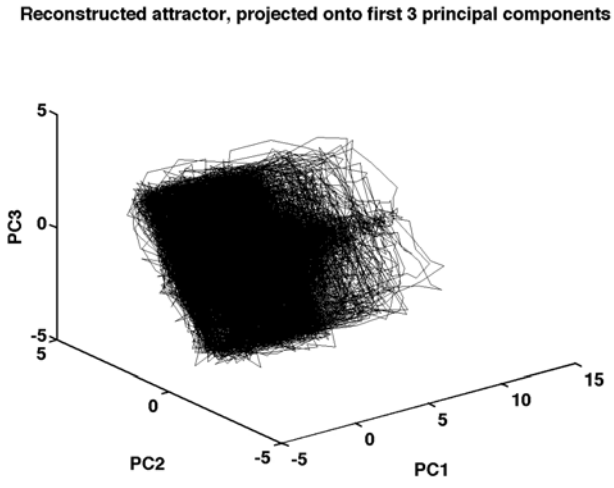


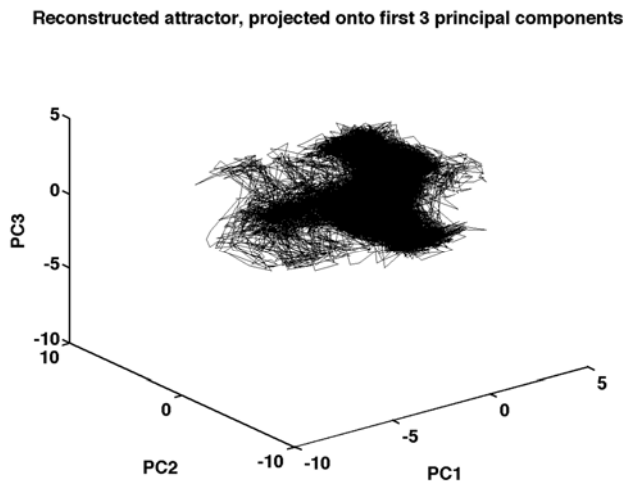
Figure 6.7: False nearest neighbours of electrochemical noise (Run 18).



a)



b)



c)

Figure 6.8: Phase space reconstruction of all 3 signals **a)** Run 18 **b)** Run 7 **c)** Run 17.

Although it might seem that the attractors obtained from the reconstructions in Figure 6.8 has little geometric structure, it should be considered that they are view in only 3 dimensions when they are actually defined in 8 dimensions, where they might have a definite geometric structure.

6.3. Methodology for detecting change in electrochemical noise signals

The methodology for detecting dynamic change in electrochemical noise data is divided into two parts. The first part is associated with specifying the normal process and providing confidence limits where the second part in turn is associated with the monitoring of the system.

The procedure to specify normal process with confidence limits:

1. Collect for a sufficiently long period time series data representative of normal process behaviour.

2. Determine a suitable time delay T and embedding dimension d_e for phase space reconstruction.

3. Divide the data into a number of segments of length N, sufficiently long to capture the full dynamics of the system, and embed each segment using the above embedding parameters (time delay T and embedding dimension d_e).

4. Estimate the correlation dimension of the segments characterising the topology of the attractor that represents the normal process behaviour

(dynamics of the system). Also estimate the mean and confidence intervals from these estimates (refer to section 3.3 for confidence interval estimation).

The procedure for monitoring of the system:

4. Collect new process data until sufficient long time series, N samples, has been obtained.

5. Embed the data with the same embedding parameters, time delay T and embedding dimension d_e, determined for the normal process.

6. Estimate the correlation dimension for the new time series data and compare it with the normal process's estimates and limits. If the correlation dimension falls outside the confidence limits of the normal process data, signal a change in the dynamics of the system.

6.4. Results obtained from electrochemical noise data

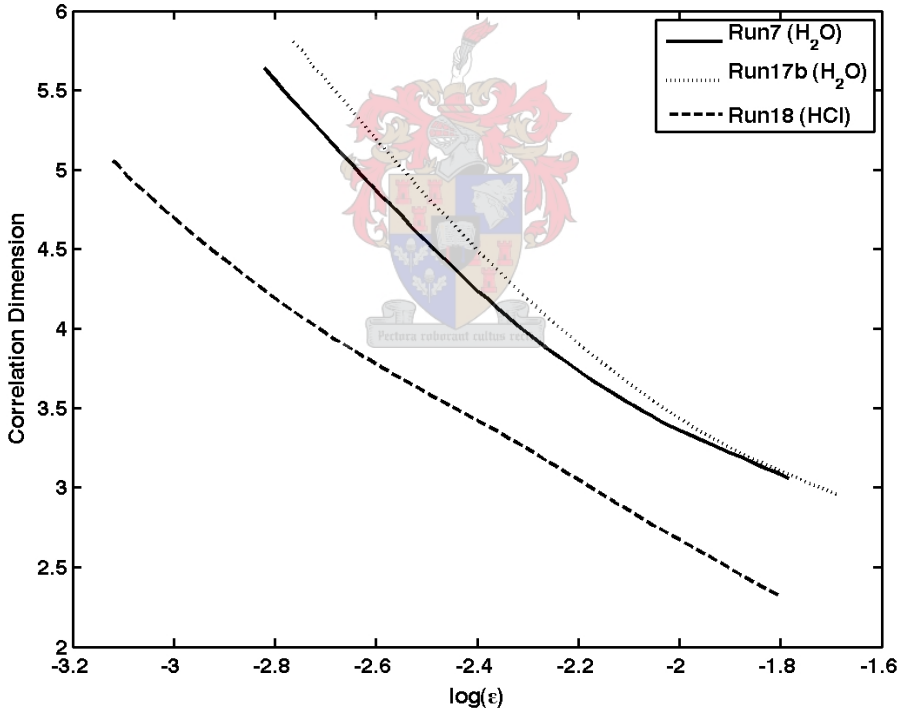


Figure 6.9: Correlation dimension estimates of the 3 different electrochemical (all data).

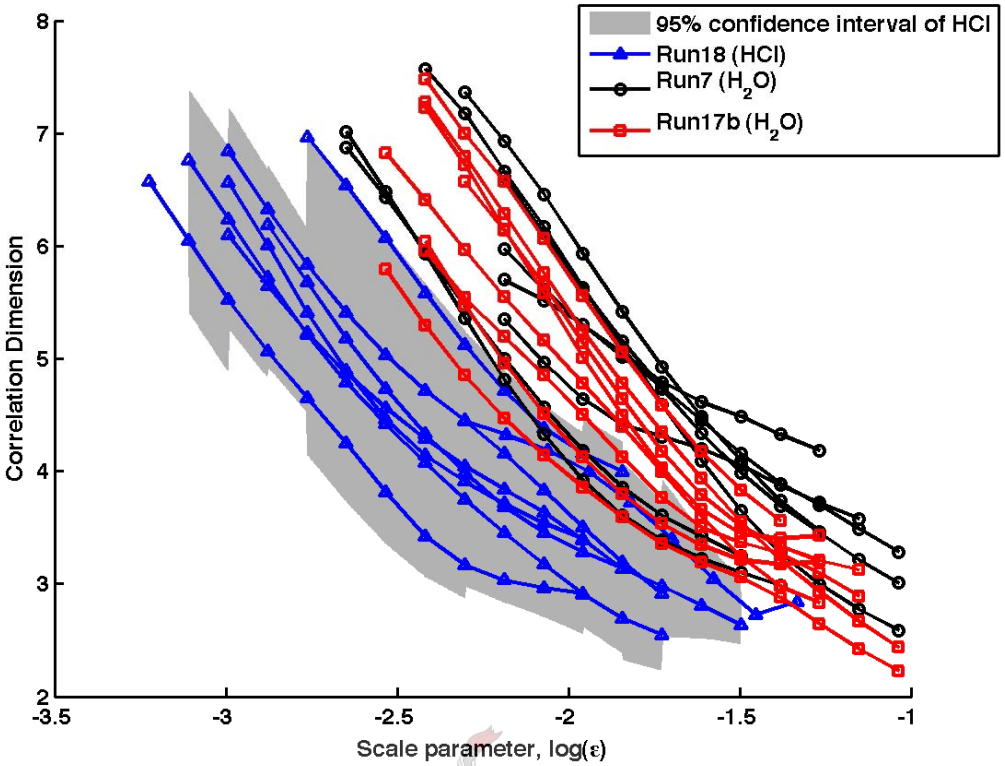


Figure 6.10: Correlation dimension estimates obtained by procedure specified in section 6.3 with $L = 8000$ data points.

6.5. Discussion of results obtained from electrochemical noise

When all the experimental time series data per run are considered, a difference in the correlation dimension estimates from different corrosive mediums can already be observed, as illustrated in Figure 6.9. The correlation dimension of the corrosion in a HCl solution, Run 18, is much lower than that produced by the corrosion in distilled water, Run 7 and Run 17. The correlation dimensions of Run 7 and Run 17 are also very much similar as can be expected since the same corrosive medium, distilled water, are used.

For a more comprehensive study each of the electrochemical noise signals is divided into a number of segments and analysed as specified in section 6.3. The result obtained is illustrated in Figure 6.10. Again a difference in the correlation dimension from the electrochemical noise signals of corrosion in different solutions can be identified. All except one of the correlation dimension estimates from Run 7 and Run 17 (distilled water solution) falls outside the 95% confidence limits for the correlation dimension of Run 18 (HCl solution). The correlation dimension estimates of Run 7 and Run 17 is also very similar indicating that the systems producing these signals might be the same.

The correlation dimension curves are estimated with Judd's (1992) algorithm, which is only accurate up to about 4, and the estimates obtained ranged between 2 and 8. Although this might present a problem the correlation dimension is a pivotal test statistic meaning that the probability distribution is the same for all processes consistent with the hypothesis, in this case corrosion in a HCl solution. Thus the actual correlation dimension value is not as important as the separation between the correlation dimension curves obtained from corrosion in different solutions.

7. Conclusions

In this study a data-driven method for detecting change in nonlinear system is proposed and compared with two other similar methods proposed in the literature. These algorithms were applied to nonlinear simulated data: the BZ reaction, an autocatalytic process and a predator-prey system. The proposed algorithm was also applied to real process data, the electrochemical noise from a freely corroding system. The ability of the different algorithms to detect dynamic changes was compared and the insight gained from the study is briefly discussed below:

- Although a large number of papers has been published on nonlinear systems, relatively few are on detecting change in such systems from data, particularly in process engineering where no satisfactory method has yet been proposed. The most common approach in the literature is the use of the following three nonlinear statistics: Lyapunov exponents, correlation dimension and various entropies. This led to the development of the algorithm proposed in this study, which uses the correlation dimension as test statistic.

- All three the algorithms relied on the embedding of the time series data into phase space. The first minimum of the average mutual information was used to estimate the time delay for time delay embedding, where after the embedding dimension was obtained from the FNN algorithm. In the case of embedding by SSA the point of linear decorrelation was obtained from the first zero of the autocorrelation function. A subjective choice is required to obtain the set of eigenvectors retained in the embedding. Although the embedding parameters were successfully estimated with these guidelines, no objective way of estimating them is available.

- Even though the change-point detection algorithm did identify the dynamic changes in the simulated time series data, the detection statistic had a very irregular behaviour. The irregular behaviour, which is due to the data not having a normal distribution, was aggravated even more by the presence of instrumental noise or chaotic behaviour. This led the algorithm to produce multiple false alarms and even a failure to detect any change in one case. Although the algorithm detects dynamic changes fast, the presence of multiple false alarms and poor robustness to specified parameters have to be reckoned with.

- The two-dimensional map produced by the mutual cross prediction algorithm gives a good indication on the existence of dynamic changes in a system. The

method also appeared to be robust in the presence of measurement noise. Although the algorithm was reliable, it did require additional calculations and the training of am MLP, which was time consuming, making it not suited for online monitoring.

- The proposed algorithm using the correlation dimension as test statistic proved to be useful for the detection of change in nonlinear time series data. The method showed a good robustness to the embedding parameters that were required, despite some indications to the contrary in the literature. Although the performance of the algorithm deteriorated as instrumental noise was added to the simulated data, it was still able to identify the changes in the noisy time series data considered in this study.

- An advantage that the methods of correlation dimension as test statistic and mutual cross prediction has over the change point detection algorithm is that a slow change will not be integrated into the normal state of the system. However, these two methods require a calibration data set where no change is present, which are not required by the change point detection algorithm.

- The proposed method with the correlation dimension as test statistic was also applied to real electrochemical noise data. Here the method was able to distinguish between the different corrosion current signals, which were obtained from the corrosion of stainless steel in water and hydrochloric acid. Thus the proposed method was able to distinguish between two different corrosion phenomena, i.e. uniform and localised corrosion, underscoring its potential as a useful approach to detect dynamic change in nonlinear systems.

7.1. Future developments

Although the proposed method with the correlation dimension as test statistic appears to be a promising approach to detecting change in nonlinear systems, more work needs to be done with real-world data.

More over, future work should also explicitly assess the robustness of the method with regards to parameter misspecification, window length, etc.

8. References

Abarbanel, H.D.I., (1996), *Analysis of observed chaotic data*. New York: Springer Verlag.

Aldrich, C., (1998), Introduction to exploratory data analysis and empirical modelling. Department of process engineering, University of Stellenbosch, Stellenbosch.

Aldrich, C., (2002) *Exploratory analysis of metallurgical process data with neural networks and related method*. Elsevier Science B.V.

Anderson, J.J, Gurarie, E., and Zabel, R.W., (2005), Mean free-path length theory of predator-prey interactions: Application to juvenile salmon migration. *Ecological Modelling*, Vol. 186, pp. 196-211.

Andreou, E., and Ghysels, E., (2005), Monitoring disruptions in financial markets. *Journal of Econometrics*, submitted.

Añino, M.M., Torres, M.E., and Schlotthauer, G., (2003), Slight parameter changes detection in biological models: a multiresolution approach. *Physica A*, Vol. 324, pp. 645-664.

Anita, T., Pujar, M.G., Shaikhm H., Dayal, R.K., and Khatak, H.S., (2006), Assessment of stress corrosion crack initiation and propagation in AISI type 316 stainless steel by electrochemical noise technique. *Corrosion Science*, Vol. 48, No. 9, pp. 2689-2710.

Barnard, J.P., (1999), Empirical state space modeling with application in online diagnosis of multivariate non-linear dynamic systems. *PhD thesis*, University of Stellenbosch.

Basseville, M., and Nikiforov, I. V., (1993), Detection of abrupt changes: Theory and application. Previously published by New Jersey: Prentice-Hall, Inc. available on world wide web at: (http://www.irisa.fr/sigma2/kniga).

Bezuidenhout, L.C., (2004), *Detecting change in nonlinear dynamic process systems*. MscEng thesis, Department of Process Engineering, University of Stellenbosch.

Bhagwat, A., Srinivasan, R., and Krishnaswamy, P.R., (2003), Multi-linear model-based fault detection during process transitions. *Chemical Engineering Science*, Vol. 58, pp. 1649-1670.

Bloch, G., Ouladsine, M., and Thomas, P., (1995), On-line fault diagnosis of dynamic systems via robust parameter estimation. *Control Engineering Practice*, Vol. 3, No. 12, pp. 1709-1717.

Cai, C., Zhang, Z., Cao, F., Gao, Z., Zhang, J., and Cao, C., (2005), Analysis of pitting corrosion behaviour of pure Al in sodium chloride solution with the wavelet technique. *Journal of Electroanalatical Chemistry*, Vol. 578, pp. 143-150.

Cao, F.H., Zhang, Z., Su, J.X., Shi, Y.Y., and Zhang, J.Q., (2006), Electrochemical noise analysis of LY12-T3 in EXCO solution by discrete wavelet transform technique. *Electrochimica Acta*, Vol. 51, pp. 1359-1364.

Casdagli, M.C., (1997), Recurrence plots revisited. *Physica D*, Vol. 108, pp. 12-44.

Dowd, M., (2005), A bio-physical coastal ecosystem model for assessing environmental effects of marine bivalve aquaculture. *Ecological Modelling*, Vol. 183, pp. 323-346.

Eckmann, J.P., and Ruelle, D., (1985), Ergodic theory of chaos and strange attractors. *Review of Modern Physics*, Vol. 57, pp. 617-656.

Eckmann, J.P., Kamphorst, S.O., Ruelle, D., and Ciliberto, S., (1986), Lyapunov exponents from time series. *Physical Review A*, Vol. 34, pp. 4971-4979.

Epureanu, B.I., and Yin, S., (2004), Identification of damage in an aeroelastic system based on attractor deformations. *Computers and Structures*, Vol. 82, pp. 2743-2751.

Fenu, G., and Parisini, T., (1999), A note on nonparametric kernel smoothing for model-free fault symptom generation. *Automatica*, Vol. 35, pp. 1175-1179.

Field, R.J., Körös, F., and Noyes, R.M., (1972), Oscillations in Chemical Systems. II. Thorough Analysis of Temporal Oscillation in the Bromate-Cerium-Malonic Acid System. *Journal of the American Chemical Society*, Vol 94, No. 25, pp. 8649-8664.

Fraser, A.M., and Swinney, H.L. (1986), Independent coordinates for strange attractors from mutual information. *Physics Review A*, Vol. 33, pp. 1134-1140.

García, E., Hernández, M.A., Rodríquez, F.J., Genescá, J. and Boerio, F.J., (2003), Oscillation and chaos in pitting corrosion of steel. *Corrosion*, Vol. 59, No. 1, pp. 50-58.

Gomez-Duran, M., and Macdonald, D.D., (2006), Stress corrosion cracking of sensitized Type 304 stainless steel in thiosulphate solution. II. Dynamics of fracture. *Corrosion Science*, Vol. 48, pp. 1608-1622.

Graig, C., Neilson, R.D., and Penman, J., (2000), The use of correlation dimension in condition monitoring of systems with clearance. *Journal of Sound and Vibration*, Vol. 231, No. 1, pp. 1-17.

Grassberger, P., and Procaccia, I., (1983), Characterization of strange attractors. *Physical Review Letters*, Vol. 50, pp. 246-249.

Greisiger, H., and Schauer, T., (2000), On the interpretation of the electrochemical noise data for coatings. *Progress in Organic Coatings*, Vol. 39, pp. 31-36.

Györgyi, L. and Field, R.J., (1992), A three-variable model of deterministic chaos in the Belvous-Zhabotinsky reaction. *Nature*, Vol. 355, No. 6363, pp. 808-110.

Györgyi, L., and Field, R.J., (1991) Simple models of deterministic chaos in the Belousov-Zabotinsky reaction. *J. Phys. Chem.*, Vol. 95, pp. 6594-6602.

Haykin, S., (1994), *Neural Networks: A comprehensive foundation*. Macmillan College publishing Company, Inc, New York.

Hegger, R., Kantz, H., and Schreiber, T., (1998), *Practical implementation of nonlinear time series methods: The TISEAN package*. available on world wide web at: (http://www.mpipks-dresden.mpg.de/~tisean/TISEAN_2.1/docs/indexf.html).

Hively, L.M., Gailey, P.C., and Protopopescu, V.A., (1999), Detecting dynamical change in nonlinear time series. *Physics Letters A*, Vol. 258, pp. 103-114.

Jemwa, G.T., (2003), *Multivariate nonlinear time series analysis of dynamic process systems*. MscEng thesis, Department of Process Engineering, University of Stellenbosch.

Judd, K. and Mees, A., (1996), Modeling chaotic motions of a string from experimental data. *Physica D*, Vol. 92, pp. 221-236.

Judd, K., (1992), An improved estimator of the dimension and some comments on providing confidence intervals. *Physica D*, Vol. 56, pp. 216-228.

Judd, K., (1994), Estimating dimension from small samples. *Physica D*, Vol. 82, pp. 426-444.

Kantz, H., and Schreiber, T., (1997), *Nonlinear time series analysis*. Cambridge: University Press.

Kennel, M.B., (1997), Statistical test for dynamical nonstationarity in observed time-series data. *Physical Review E*, Vol. 56, No. 1, pp. 316-321.

Kennel, M.B., and Abarbanel, H.D.I., (2002), False neighbors and false strands: A reliable minimum embedding dimension algorithm. *Phys.Rev. E.*, Vol. 66, pp. 1-18.

Kennel, M.B., Brown, R., and Abarbanel, H.D.i., (1992), Determining minimum embedding dimension useing a geometrical construction. *Physical Review A*, Vol. 45, pp. 3403-3411.

Lai, Y-C., and Lerner, D., (1998), Effective scaling region for computing the correlation dimension from chaotic time series. *Physica D*, Vol. 115, pp. 1-18.

Lee, J.S., and Chang, K.S., (1996), Applications of chaos and fractals in process systems engineering. *Journal of Process Control*, Vol. 6, No. 2/3, pp. 71-87.

Legat, A., Osredkar, J., Kuhar, V. and Leban, M. V., (1998) Detection of various types of corrosion processes by the chaotic analysis of electrochemical noise. *Materials Science Forum*, 289-292, 807-812.

Lewis, D.M., and Bala, S.I., (2006), Plankton predation rates in turbulence: A study of the limitations imposed on a predator with a non-spherical field of sensory perception. *Journal of Theoretical Biology*, article in press, accepted 30 Jan. 2006.

Lindfield, G., and Penny, J., (2000), *Numerical methods using matlab*. 2nd edition, Prentice Hall.

Liu, X.F., Wang, H.G., and Gu, H.C., (2006), Fractal characteristic analysis of electrochemical noise with wavelet transform. *Corrosion Science*, Vol. 48, pp. 1337-1367.

Logan, D., and Mathew, J., (1996), Using the correlation dimension for vibration fault diagnosis of rolling element bearings – I. Basic concepts. *Mechanical Systems and Signal Processing*, Vol 3, No. 3, pp. 241-250.

Lu, P., and Teulilo, A., (2001) Oscillating chemical reaction. Department of physics, The University of Sydney, available on world wide web at: (http://oldsite.vislab.usyd.edu.au/education/sc3/2001/paul_ana/paul_ana.pdf).

Manuca, R., and Savit, R., (1996), Stationarity and nonstationarity in time series analysis. *Physica D*, Vol. 99, pp. 134-161.

Marino, A.A, Nilsen, E., and Frilot, C., (2003), Localization of electroreceptive function in rabbits. *Physiology & Behavior*, Vol. 79, pp. 803-810.

Maulud, A., Wang, D., Romagnoli, J.A., (2006), A multi-scale orthogonal nonlinear strategy for multi-variate statistical process monitoring. *Journal of Process Control*, Vol. 16, pp. 671-683.

Montesino, O.M.E, Rolo, N.A., and Azoy, C.A., (2003), The attractor dimension determination applied to monitoring and surveillance in nuclear power plants. *Progress in Nuclear Energy*, Vol. 43, No. 1-4, pp. 389-395.

Moskvina, V., and Zhigljavsky, A., (2000), Application of singular-spectrum analysis to change-point detection in time series. *Journal of Time Series Analysis*, (submited).

Nichols, J.M., Virgin, L.N., Todd, M.D., and Nichols, J.D., (2003), On the use of attractor dimension as a feature in structural health monitoring. *Mechanical Systems and Signal Processing*, Vol. 17, No. 6, pp. 1305-1320.

Park, C., and Kwon., H, (2005), Electrochemical noise analysis of localized corrosion of duplex stainless steel aged at 475 °C. *Materials Chemistry and Physics*, Vol. 91, pp. 355-360.

Perry, R.H., and Green, D.W., (1997), *Perry's chemical engineers' handbook*. 7th edition, New York: McGraw-Hill.

Phillips, J.D., (2006), Deterministic chaos and historical geomorphology: A review and look forward. *Geomorphology*, Vol. 76, pp. 109-121.

Pranatynasto, T.N., and Qin, S.J., (2001), Sensor validation and process fault diagnosis for FCC units under MPC feedback. *Control Engineering Practice*, Vol. 9, pp. 877-888.

Rosso, O.A., and Mairal, M.L., (2002), Characterization of time dynamical evolution of electroencephalographic epileptic records. *Physica A*, Vol. 312, pp. 469-504.

Sano, M., and Sawada, Y., (1985), Measurement of the Lyapunov spectrum from a chaotic time series. *Physics Letters Review A*, Vol. 55, pp. 1082-1085.

Sauer, T., Yorke J.A., and Casdagli, M., (1991), Embedology. *Journal of Statistical Physics*, Vol. 65, 579-616.

Schreiber, T., (1997), Detection and analyzing nonstationarity in time series using nonlinear cross prediction. *Physical Review Letters*, Vol. 78, No. 5, pp. 843-846.

Schreiber, T., (1999), Interdisciplinary application of nonlinear time series methods. *Physics Reports*, Vol. 308, pp. 1-64.

Schreiber, T., and Schmitz, A., (2000), Surrogate time series. *Physica D*, Vol. 142, pp. 346-382.

Small, M., Judd, K., (1998), Correlation dimension: A pivotal statistic for non-constrained realizations composite hypothesis in surrogate data analysis. *Physica D*, Vol. 120, pp. 386-400.

Sobajic, D.J, Pao, Y., and Lee, D.T., (1992), Autonomous adaptive synchronous machine control. *International Journal of Electrical Power & Energy Systems*, Vol. 14, No. 2-3, pp. 166-174.

Sotomayor, O.A.Z., and Odloak, D., (2005), Observer-based fault diagnosis in chemical plants. *Chemical Engineering Journal*, Vol. 112, pp. 93-108.

Takens, F., (1983), *Detecting strange attractors in turbulence*. Lecture notes in mathematics, 898, Springer (Berlin), pp. 366-381.

Theron, J.P., and Aldrich, C., (2000), *Identification of nonlinearities in dynamic process systems*. Department of Process Engineering, University of Stellenbosch.

Torres, H.M., Gurlekian, J.A., Rufiner, H.L., and Torres, M.E., (2006), Self-organizing map clustering based on continuous multiresolution entropy. *Physica A*, Vol. 361, pp. 337-354.

Torres, M.E., and Gamero, L.G., (2000), Relative complexity changes in time series using information measures. *Physica A*, Vol. 286, pp. 457-473.

Torres, M.E., Añino, M.M., and Schlotthauer, G., (2003), Automatic detection of slight parameter changes associated to complex biomedical signals using multiresolution q-entropy. *Medical Engineering & Physics*, Vol. 25, pp. 859-867.

Torres, M.E., Añino, M.M., Gamero, L.G., and Gemignani, M.A., (2001), Automatic detection of slight changes in nonlinear dynamical systems using multiresolution entropy tools. *International Journal of Bifurcation & Chaos*, Vol. 11, No. 4, pp. 967-982.

Trendafilova, I., and Van Brussel, H., (2001), Non-linear dynamics tools for the motion analysis and condition monitoring of robot joints. *Mechanical Systems and Signal Processing*, Vol. 15, No. 6, pp. 1141-1164.

Übeyli, E.D., and Güler, I., (2004), Detection of electrocardiographic changes in partial epileptic patients using Lyapunov exponents with multilayer perceptron neural networks. *Engineering Applications of Artificial Intelligence*, Vol. 17, pp. 567-576.

Venkatasubramanian, V., Rengaswamy, R., Yin, K. and Kavuri, S.N., (2003a), A review of process fault detection and diagnosis Part I: Quantitative model-based methods. *Computer and Chemical Engineering*, Vol. 27, pp. 293-311.

Venkatasubramanian, V., Rengaswamy, R., Yin, K. and Kavuri, S.N., (2003b), A review of process fault detection and diagnosis Part III: Process history based methods. *Computer and Chemical Engineering*, Vol. 27, pp. 327-346.

Wang, K.-Y., Shallcross, D.E., Hadjinicolaou, P., and Giannakopoulos, C., (2002), An efficient chemical system modelling approach. *Environment Moddeling & Software*, Vol. 17, pp. 731-745.

Wise, B. M., and Neal, B.G., (1996), The process chemometrics approach to process monitoring and fault detection. *Journal of process control*, Vol. 6, No. 6, pp. 329-348.

Wittren, I.H., and Frank, E., (2005), *Data mining – Practical machine learning tools and techniques*. 2nd edition, Elsevier.

Wu, B., and Chen, M., (1999), Use of fuzzy statistical technique in change periods detection of nonlinear time series. *Applied Mathematics and Computation*, Vol. 99, pp. 241-254.

Yu, D., Lu, W., and Harrison, R.G., (1999), Detecting dynamical nonstationarity in time series data. *Chaos*, Vol. 9, No. 4, pp. 865-870.

Yu, D.L., Gomm, J.B., and Williams, D., (1999), Sensor fault diagnosis in a chemical process via RBF neural network. *Control Engineering Practice*, Vol. 7, pp. 49-55.

Zbilut, J.P, and Webber Jr, C.L., (1992), Embedding and delays as derived from quantification of recurrence plots. *Physics Letters A*, Vol. 171, No. 3-4, pp. 199-203.

Zhang, D., Györgyi, L., and Peltier, W.R., (1993), Deterministic chaos in the Belousov-Zhabotinsky reaction: Experiments and simulations. *Chaos*, Vol. 3, No. 4, pp. 723-745.

Zhang, Z., Leng, W.H., Cai, Q.Y., Cao, F.H., and Zhang, J.Q., (2005), Study of the zinc electroplating process using electrochemical noise technique. *Journal of Electoanalytical Chemistry*, Vol. 578, pp. 257-367

Appendix A: implementation of multilayer perceptron neural network in this study

Multilayer perceptron networks are a subset of artificial neural networks and are known as universal approximators that can fit continuous curves or response surfaces to data with an arbitrary degree of accuracy, at least in theory. Artificial neural networks were originally inspired by the brain that consists of millions of interconnected neurons. Similarly artificial neural network consists of interconnected simple process units, also called neurons or nodes, organised in different layers. Only through the collective behaviour of theses neurons are artificial neural networks able to form an approximation of complex functions.

A MLP network typically consists of an input layer, one or more hidden layers and an output layer as indicated in Figure A.1. The nodes in a particular layer are connected to the nodes in successive layers by weighted connections. The weights define the internal relationship between the input and output of the network and are adjusted by training the network, once the network structure is defined.

Before the network can be trained, the overall structure of the network has to be defined. This includes the number of layers, number of nodes per layer and types of nodes. First the input layer should be defined. Since the embedding vectors are used as the input vectors, the number of nodes in the input layer should thus be equal to the embedding dimension. The input layer is generally not a true layer since these nodes are only used to distribute the input signals to the nodes in first hidden layer.

The number of hidden layers and number of neurons per hidden layer depends on the complexity of the system under investigation. A single hidden layer of nodes proofed to be adequate for the systems considered in this study. One way to estimate the optimum number of hidden nodes is by minimizing the Schwartz Information Criterion (SIC) (Judd & Mees, 1996).

$$SIC = N\sum_{i=1}^{p}\log\left(\frac{V_{\theta,Z,i}}{N}\right) + d_m \log(N) \tag{A.1}$$

Where θ denote a particular set of model parameters and N are the number of observations in data set Z. The model order d_m is the number of model parameters of a model structure and depends on the number of input nodes m, output nodes p and hidden nodes S.

$$d_m = S(m+p) \tag{A.2}$$

The model fitness norm $V_{\theta,Z}$ is defined as:

$$V_{\theta,Z} = \frac{1}{N}\sum_{t=1}^{N}\frac{1}{2}(y_t - \hat{y}_{t,\theta})^2 \tag{A.3}$$

with $\hat{y}_{t,\theta}$ and y_t being the model predicted system output and the observed system output respectively.

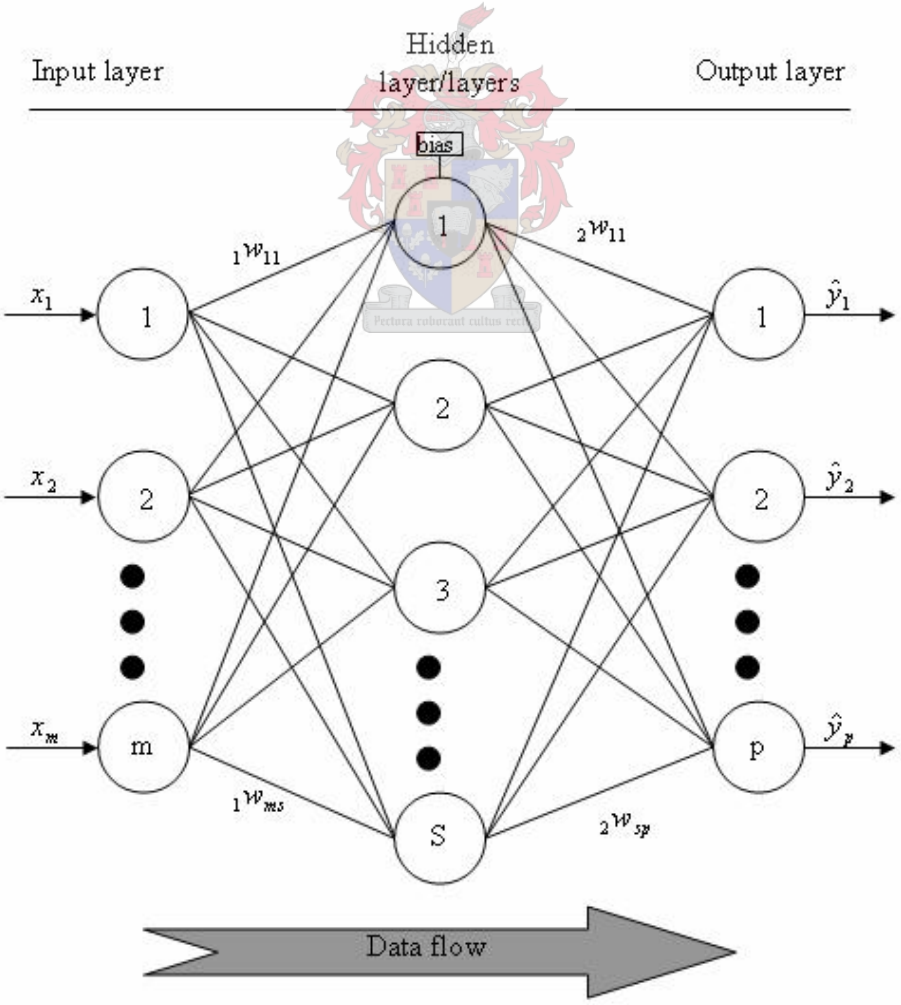


Figure A.1: Structure of a typical multilayer perceptron neural network.

Now the type of nodes used for the hidden layers have to be specified. Each node model consists of a processing element with a set of inputs as well as one output as illustrated in Figure A.2.

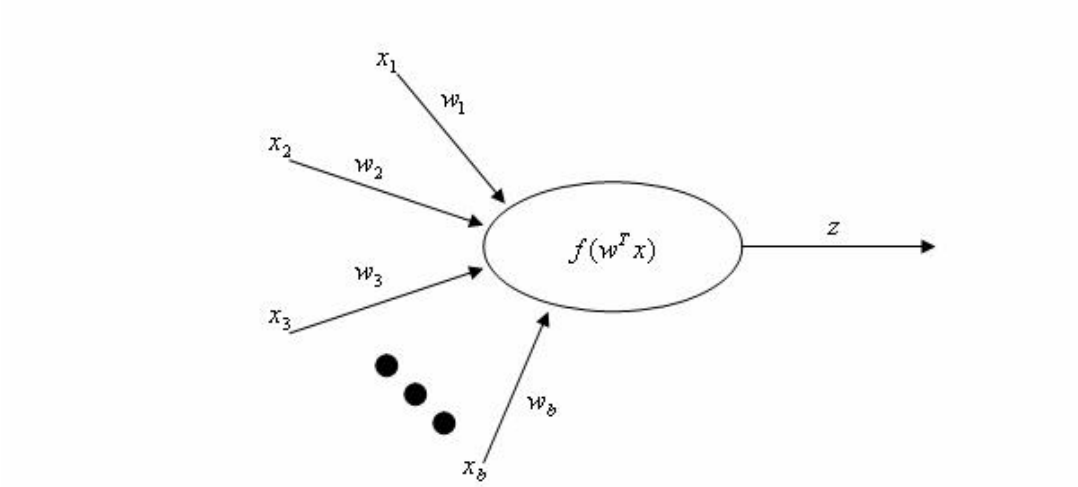


Figure A.2: Model of a single neuron/node.

The output of the node can be expressed as:

$$z = f(\sum_{i=1}^{b} w_i x_i) = f(w^T x) \tag{A.4}$$

The weight vector w and input vector x are defined as:

$$w = [w_1, w_2, ..., w_b]^T \tag{A.5}$$

and

$$x = [x_1, x_2, ..., x_b]^T \tag{A.6}$$

The function $f(w^T x)$ is referred to as the activation function of a node. The argument of the activation function is sometimes referred to as the potential of the node, in analogy to the membrane potentials of biological neurons (Aldrich, 1998). Sigmoidal activation functions are used widely in neural network applications and a typical transfer function is illustrated in Figure A.3. The activation function used for the hidden layer nodes is a hyperbolic tangent sigmoid transfer function ('tansig') with $w^T x = \phi$ and is defined as:

$$f(\phi) = \frac{2}{(1 + e^{-2\phi})} - 1 \tag{A.7}$$

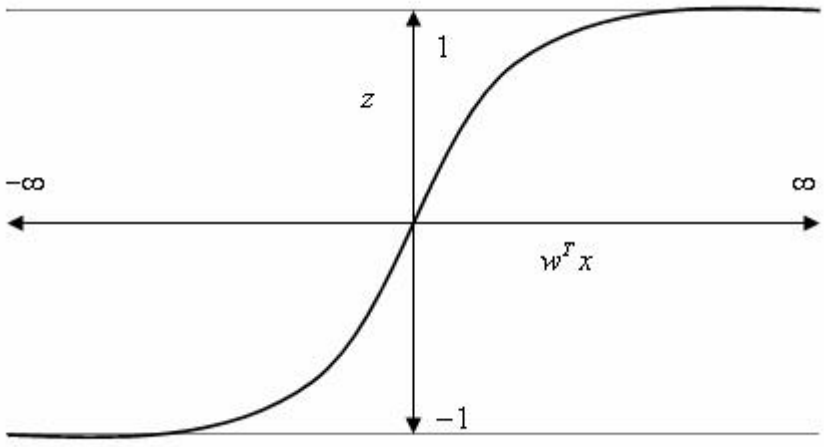


Figure A.3: hyperbolic tangent sigmoid activation function ('tansig').

An additional input x_0, with associated weight w_0, referred to as a bias can be defined for some nodes. This input has a fixed value of -1 but like the other inputs has an adjustable weight. A bias input value is sometimes used to offset the output of the MLP to enable the MLP to form accurate representations of process trends.

The output layer consists of only one node since the there are only one output, the observed time series. The purpose of the output layer node is basically to combine the signals obtained from the hidden layers nodes to produce a single output. Thus the activation function used for the node is a linear transfer function ('purelin').

Now that the network architecture is defined the weights are fitted to a training set consisting of actual input and output values of the system. Training typically occurs by means of an algorithm designed to minimise the mean square error between the target output and the actual output of the MLP through incremental modification of the weight matrix. During the learning process information is propagated back through the MLP in order to update the weights. The Levenberg-Marquardt algorithm optimizes the weights of the MLP through backpropagation and converges faster than most other training algorithms (Barnard, 1999). Training of the MLP network is terminated when the MLP network has leaned to generalise the relationships exemplified by the training data.

The ability of a MLP network to generalise is measured by means of cross-validation. During cross-validation the performance of the network is evaluated against a data set not used during training, the validation set. A serious disadvantage of the MLP trained with backpropagation is it may suffer from overfitting[9]. One simple modification proposed by Witten and Frank (2005) is early stopping. Here the validation error is

[9] Overfitting – The model fits the training data well but fails to generalise untrained data, caused by using a much larger network than what is required.

monitored and the training algorithm is terminated once the validation error start to increase, indicating overfitting to the training data.

A more comprehensive description of MLP networks can be found in textbooks such as (Haykin, 1999).

Appendix B: papers based on work done in this study

B.1. Aldrich, C., Qi, B.C., and Botha, P.J., (2006), Analysis of electrochemical noise with phase space methods. *Minerals Engineering*, Vol. 19, No. 14, pp. 1402-1409.

B.2. Aldrich, C., and Botha, P.J., (2006), Detecting change in complex process systems with phase space methods. *Industrial and Engineering Chemistry Research*, submitted.