D I P L O M A R B E I T

Nonlinear Dependencies in and between Time Series

Ausgeführt am Institut für
Wirtschaftsmathematik
der Technischen Universität Wien

unter der Anleitung von
Em.O.Univ.Prof. Dipl.-Ing. Dr.techn. Manfred Deistler

durch

Markus Waser
Sobieskigasse 32/2
1090 Wien

Wien, Februar 2010 _____

Contents

List of Figures

List of Tables

Abstract

A statistical method for identification of nonlinearity in time series is discussed. The approach is based on Monte Carlo methods and bootstrapping. Artificial surrogate series are generated, which are consistent with a range of null hypotheses that include linearity. In this way, the distribution of test statistics, that are computed for the original data, can be empirically estimated by an ensemble of surrogates. If a test statistic is out of range of its estimated distribution, then the corresponding null hypothesis is rejected, and nonlinearity can be assumed. However, other data characteristics, like nonstationarity, may bias the test result. Thus, surrogates should feature these characteristics as well. By means of surrogate data, nonlinear dependencies can be detected both within time series and between them. The surrogate algorithms are applied to different simultaneously measured time series taken from a diesel combustion engine. The complexity of choosing an appropriate surrogate method is demonstrated for different setups of the engine.

Acknowledgements

I would like to express my gratitude to all those who helped to make this thesis possible:

First of all, I gratefully acknowledge the supervision of Prof. Manfred Deistler, who supported me with his guidance and knowledge throughout this thesis.

Furthermore, I want to offer my regards to Prof. Dr. Luigi del Re, Dipl.–Ing. Andrea Schrems, and Dipl.–Ing. Markus Hirsch, who provided data for testing, and patiently answered all my questions.

I should not forget to thank my fellow students and friends, especially Kristen Swan, Sara Mösenbacher, Clemens Tummeltshammer, Stefan Großwindhager, and Christian Schmid, who assisted in proofreading, and with whom I always could exchange opinions.

I would also like to say thanks to my parents for their steady support throughout my studies.

Last but not least, I want to express my deepest gratitude to Judith Runggaldier for her never–ending patience, and for always being there for me.

Chapter 1

Introduction

1.1 Motivation

In order to understand the underlying dynamics of an observed time series, it is necessary to divide these dynamics into categories. Basically, they can consist of any mix of

- linear deterministic dynamics
- nonlinear deterministic dynamics
- chaos
- stochastic dynamics

Whereas linear system identification is supported by numerous tools that are well implemented in many software environments, the identification of chaos and stochastic dynamics may often not be applicable. Hence, the choice of focusing on nonlinear deterministic structures is a compromise between these options.

A basic assumption of this thesis is that the data series are observed only once, and are not replicable. The reasons for this could be high expenses of measurement, or simply, that the phenomenon in focus only occured once. A good example are geological or astronomical data. In this thesis, a sample techniques based on bootstrapping is applied, in order to estimate distributions of certain characteristics. Randomized samples are created, in order to test quantifiers against their estimated empirical distribution calculated on these samples. In many situations, it is desirable that the randomized samples feature certain properties of the original time series. A method to accomplish this is described and applied in this thesis: *the method of surrogate data*. First published by James Theiler et al. (1992), it has become a standard for testing nonlinearity over the last years. Surrogate data allow

not only testing within, but also between different univariate time series.

The applied data are physical measurements taken from a Diesel combustion engine. A high–quality engine should meet the requirements of both being fuel effective and produce as less harmful exhaust as possible. Especially the amount of emitted soot is central in numerous health–related research projects. As it is very difficult to measure soot particles, the related NO_X particles are observed instead. The amount of NO_X is influenced by a lot of variables, both related to the engine and to external influences like driving behavior. The examination of all variables, if ever possible, is a target, that is set far too high in context of this thesis. Therefore, the dependencies between only two data series, the manifold pressure and the amount of NO_X molecules, are examined here.

1.2 Composition

This thesis is divided into seven chapters:

Chapter 2, that follows this brief introduction, is supposed to give a general overview of basic linear, and nonlinear modelling techniques, including ARMA, and nonlinear parameter varying processes. A definition of wide sense stationary processes is provided as well.

The first part of *Chapter 3* is dedicated to statistical hypothesis testing, followed by more specific theory on testing of nonlinearity. Afterwards, the technique of surrogate data, based on bootstrapping and Monte Carlo simulations is presented. It allows to create artificial time series in accordance with a set of null hypotheses, in order to empirically estimate the distribution of a test statistic.

The next two chapters are the main theoretical parts of this thesis. In *Chapter 4*, different standard methods for the construction of univariate surrogates are presented. The underlying idea is to transfer a time series by discrete Fourier transform into frequency domain, and randomize its Fourier phases. The Fourier amplitudes stay unchanged, hence the linear properties of the time series are preserve. A surrogate is created by inverse discrete Fourier transform. Then, a more flexible method, coming from the field of thermo–dynamics, is presented: simulated annealing is an iterative optimization algorithm, that enables the reproduction of certain characteristics (such as linearity) by means of a cost function on a random series. The subsequent section provides a toolkit of methods for dealing with nonstationarities, that may be difficult to distinguish from nonlinear dependencies, and hence bias the results of nonlinearity testing. Instead of the Fourier transform, a wavelet

transform is used to construct surrogates, that are 'similar' to the original time series. A section on nonlinear test statistics concludes this chapter, that can be seen as justification for utilizing the nonlinear multivariate methods in the next part of this thesis.

Chapter 5 is concerned with the construction of multivariate surrogates, that enable not only testing relationships *within* each univariate subsystem, but also testing dependencies *between* them. The channels are simultaneously recorded series of physical measurements in the same system. Modifications and extensions of both the Fourier–based methods, and the applied simulated annealing procedure are described, as well as a multivariate wavelet–based technique. A section about symmetric and asymmetric nonlinear quantifiers, including transfer entropy, information theoretic measures, and measures of interdependencies, finalizes this part.

Chapter 6 describes an application of the surrogate data method. Time series measured in a Diesel engine are first treated as separate univariate channels, and then as linked system. Surrogates are used to test dependencies between manifold pressure and NO_X exhaust. The operating principle of the engine is addressed, as well as the test results. The software used for testing is MATLAB®7.8.0(R2009a).

Finally, *Chapter 7* concludes the gained findings, and gives a prospect to further possibilities of extending, and applying the presented techniques.

Appendix A provides an overview of the most important algorithms and some useful MATLAB® source code extracts that are used in this thesis.

Chapter 2

Types of Nonlinearity

2.1 Basic Linear Concepts

This section is supposed to give an overview of main linear concepts in time series analysis. Linear concepts provide the tools to gain an understanding for the nonlinear models that are extensions of linear models. Some of the most fundamental ideas are presented in this and the next section.

Let $X = (x_1, \ldots, x_n)$ with $x_t \in \mathbb{R}^m$ and $t = 1, \ldots, n$ be a time series, i.e. any set of observations ordered in time. When necessary, the information whether X is univariate or multivariate is given throughout this thesis. Since there is no need for distinction in this section, let $m \in \mathbb{N}$. In contrast to the usual assumption of classical statistics, the ordering in time is crucial for time series analysis. The basic assumption is an underlying stochastic mechanism, a stochastic process [11].

Definition 2.1.1 (Stochastic process)**.** *A stochastic process is a family of random variables* $(x_t|t \in \mathbb{T})$*, that is defined on an underlying probability space* $(\Omega, \mathfrak{A}, P)$*.*

In general, $\mathbb{T} \subset \mathbb{R}$ is a set of indices. Here, it is understood as set of discrete equidistant time points, thus $\mathbb{T} = \mathbb{N}$. The time series X is assumed to be generated by a stochastic process $(x_t|t \in \mathbb{T})$, and is called *realization* of $(x_t|t \in \mathbb{T})$.

The concept of stationarity is an important property of stochastic processes.

Definition 2.1.2 (Wide sense stationarity)**.** *A stochastic process is called wide sense stationary if*

(i) $\mathbb{E}x_t^* x_t < \infty \quad \forall t \in \mathbb{Z}$

(ii) $\mathbb{E}x_t = const \quad \forall t \in \mathbb{Z}$

(iii) $\gamma(s,t) = \gamma(s+\tau, t+\tau)$ holds $\forall \tau, s, t \in \mathbb{Z}$

where x_t^* is the conjugate transpose of x_t. The last assumption means that the covariance of a stationary process $\gamma(s,t)$ only depends on the lag τ, i.e. time difference. Therefore, the notation $\gamma(\tau)$ with $\tau \in \mathbb{Z}$ is used. Granger describes a time series as stationary if its generating mechanism is time invariant and if the series is short–memory. He points out that stationarity is a useful property but is difficult to test for in practice. What is tested are some particular aspects of stationarity, such as over time constant means and variances [5]. The importance of stationary processes is that a single realization of the process displays the probability law of the process. According to this, any descriptive quantification averaged over time, such as mean or auto–covariance, can be interpreted as estimator of the corresponding population function. For some models in this chapter, stationarity can be established when certain conditions are given.

In order to define linearity for stochastic processes, the following definitions are required:

Definition 2.1.3 (Filtration)**.** *An increasing sequence of sub–σ–algebras* $\{\mathfrak{A}_t \subset \mathfrak{A} : \ t \in \mathbb{N}\}$ *with* $\mathfrak{A}_t \subset \mathfrak{A}_{t+1}$ *is called filtration.*

Definition 2.1.4 (Martingale)**.** *Let* $\{\mathfrak{A}_t \subset \mathfrak{A} : \ t \in \mathbb{N}\}$ *be a filtration, and* $\{x_t : \ t \in \mathbb{N}\}$ *a stochastic process.* $\{x_t\}$ *is a martingale in discrete time if* $\forall t$

(i) $\mathbb{E}|x_t| < \infty$

(ii) x_t is $\mathfrak{A}_t$–measurable

(iii) $\mathbb{E}[x_{t+1}|\mathfrak{A}_t] = x_t$

Assumption (i) says that $\{x_t\} \in \mathbb{L}^1$. For a martingale $\{x_t\}$, the conditional expected value of the next x_{t+1}, given $\mathfrak{A}_t$, is equal to the last x_t. For example, let $\mathfrak{A}_t = (x_1, \ldots, x_t)$, then $\mathbb{E}[x_{t+1}|(x_1, \ldots, x_t)] = x_t$, and the conditional expected x_{t+1}, given all the past observations, is equal to x_t.

Definition 2.1.5 (Martingale difference sequence)**.** *Let* $\{x_t\}$ *be a martingale, and* $\{y_t\}$ *be its difference sequence*[1]*.* $\{y_t\}$ *is called martingale difference sequence, if it is a zero–mean, uncorrelated process with,* $\forall t$*,*

$$\mathbb{E}[y_{t+1}|y_t, \ldots, y_1)] = 0$$

[1] $y_t = x_t - x_{t-1}$

This definition of a martingale difference sequence is more restrictive than the definition of white noise, because not only the unconditional expectation is zero, as for white noise, but also its conditional expectation is zero at every t. Note, that any martingale sequence is white noise.
Wold's representation theorem says that every stationary process can be written as an infinite moving average process

$$x_t = \sum_{j=0}^{\infty} b_j \epsilon_{t-j} \quad , \quad b_j \in \mathbb{R}^{m \times m} \tag{2.1}$$

The infinite sum in equation 2.1 exists, if $\sum_{j=0}^{\infty} |b_j| < \infty$. $\{\epsilon_t\}$ is uncorrelated, and called innovation process of $\{x_t\}$. The right side of equation 2.1 can be interpreted as linear predictor of x_t. These findings allow the following definition of linearity for stochastic processes:

Definition 2.1.6 (Linear stochastic process)**.** *A stochastic process $\{x_t\}$ is considered to be linear, if its innovation process $\{\epsilon_t\}$ is a Martingale difference sequence.*

An equivalent explanation is that the linear least squares predictor of x_t equals the least squares predictor of x_t.

The following types of stochastic processes are the foundation of nonlinear extensions that are presented later.

MA(q). A stochastic process (x_t) is called *moving average process of order q* $(MA(q))$ when $\forall t = 1, \ldots, n$

$$x_t = \sum_{j=0}^{q} b_j \epsilon_{t-j} \quad , \quad b_j \in \mathbb{R}^{m \times m} \tag{2.2}$$

with $b_0 \neq 0$, $b_q \neq 0$ and ϵ_t white noise, i.e. it is a zero mean process with no linear memory. It can easily be proved that every $MA(q)$ and even $MA(\infty)$ is stationary. Because this proof is in no way important for the purpose of this thesis, the reader is referred to the broad range of introductory literature about time series analysis.

AR(p). In data analysis, a phenomenon that is often observed is that the x_t depend on their past $x_{t-1}, \ldots, x_{t-p}$. This is the basic assumption of an *autoregressive process of order p* $(AR(p))$

$$x_t = \sum_{i=1}^{p} a_i x_{t-i} + \epsilon_t \quad , \quad a_i \in \mathbb{R}^{m \times m} \tag{2.3}$$

with $a_p \neq 0$ and ϵ_t white noise again. Note, that $AR(p)$ processes are only stationary in a wide sense under certain stability–conditions.

***ARMA*(p, q).** The combination of the two former models is the *autoregressive moving average process of orders p and q* ($ARMA(p, q)$)

$$x_t = \sum_{i=1}^{p} a_i x_{t-i} + \sum_{j=0}^{q} b_j \epsilon_{t-j} \quad , \quad a_i, b_j \in \mathbb{R}^{m \times m} \tag{2.4}$$

with the same parameter assumptions as before. It is generally considered good practice to find the smallest values of p and q that provide an acceptable fit to the data.

Clearly other more exotic linear models are possible, but they are not the subject of this thesis. Statisticians George Box and Gwilym Jenkins were two of the first to use these kinds of stochastic models for time series analysis in 1970. The Box–Jenkins methodology applies $ARMA$ or $ARIMA$ models to find the best fit of a time series to its past values in order to make forecasts. According to them, the stages of analysis of linear models are

- model specification
- model estimation
- model evaluation

This modelling strategy can also serve as a guideline for nonlinear modelling, which will be discussed in the next section. For further interest on linear stochastic processes, the reader is referred especially to [49] by Box and Jenkins, and also to [51] by Granger and Newbold, and to [50] by Hamilton.

2.2 Modelling Nonlinearity

As mentioned, many nonlinear models are generalizations of AR, MA, or $ARMA$ models. The following generalizations are those which have also found application in practical modelling situations. There is a broad range of alternatives which will not be discussed here.

***NLAR*(p).** An important generalization of AR models is the *nonlinear autoregressive model of order p* ($NLAR(p)$)

$$x_t = f(x_{t-i}, i = 1, \ldots, p) + \epsilon_t \tag{2.5}$$

where $\epsilon_t \sim i.i.d.(0, \sigma^2)$ and $f(\cdot)$ is nonlinear. Some particular models in this class have been intensively investigated and applied. The most important of these are presented, in accordance with [5] and [6].

(S)TAR, SETAR. Another subclass of $NLAR$ models are the *threshold autoregressive models of orders p and d* ($TAR(p,d)$). The assumption of these models is a change in the parameters to a switching rule, which usually depends on a past x_{t-d}

$$x_t = \begin{cases} \sum_{i=1}^{p} a_i^{(1)} x_{t-i} + \epsilon_t^{(1)} & \text{if } x_{t-d} \geq c \\ \sum_{i=1}^{p} a_i^{(2)} x_{t-i} + \epsilon_t^{(2)} & \text{if } x_{t-d} < c \end{cases} \tag{2.6}$$

where d is a positive integer, and $\epsilon_t^{(1)}$ and $\epsilon_t^{(2)}$ are independent white noise processes. The parameter c is called 'threshold parameter'.

There are several possible generalizations regarding the switching rule. In equation 2.6, the parameter switch happens abruptly, which is not wanted in some cases. This leads to the *smooth transition autoregressive models* ($STAR$). A descriptive example that can easily be extended is

$$x_t = a \cdot g(x_{t-2}) x_{t-1} + \epsilon_t \tag{2.7}$$

where $g(\cdot)$ is a smooth nondecreasing function with $g(\underline{x}) = \alpha_1$, $g(\overline{x}) = \alpha_2$, and $\alpha_1 < g(\cdot) < \alpha_2$ all other x_t. Granger compares $g(\cdot)$ to a cumulative density, or a logistic function. [5] A similar extension of TAR models is the *self–exciting threshold autoregressive model* ($SETAR$). The 2.6 model is also in the class of $SETAR$ models, since the switching rule only depends on the past of x_t. Other models with switching rules that depend on exogenous variables y_t and their past values y_{t-d} would be possible as well, but are not discussed here.

NMLA(q). Nonlinear generalizations of MA models are also widely used, for example the *nonlinear moving average models of order q* ($NLMA(q)$). In general $NLMA(q)$ have the following representation

$$x_t = f(\epsilon_{t-j}, j = 0, \ldots, q) \tag{2.8}$$

with a nonlinear function $f(\cdot)$. The errors ϵ_t are not necessarily white noise.

Chapter 3

Testing for Nonlinearity

3.1 Statistical Hypothesis Testing

A formal framework for posing questions about certain properties of experimental data can be found in statistical hypothesis testing. In context of nonlinearity, the questions arising could be

- Do linear deterministic dynamics sufficiently describe the time series?
- Is there a nonlinear structure in the time series?
- What kind of nonlinear structure?

In order to test for nonlinearity, the following has to be specified: a *null hypothesis* H_0 against which observables are tested, a *discriminating statistic* T, and a probability of rejecting H_0 when it is in fact true.

Null hypothesis H_0. The null hypothesis is a potential explanation for the data that we seek to show is inadequate. It is not positively proven because the aim is to show that the data are unlikely to be consistent with H_0, and hence to reject the null hypothesis. Theiler and Prichard point out that it is important to distinguish two types of null hypotheses:

- simple
- composite

Formally, let $\mathcal{F}$ be the space of parameters under consideration, and $\mathcal{F}_0 \subset \mathcal{F}$ be the set of parameters that are consistent with the null hypothesis. H_0 says that the process F, that generated the data, is an element of the set $\mathcal{F}_0$. If this set consists of a single member, then the null hypothesis is simple. Otherwise, the null hypothesis is composite, and it says that the data were generated by some process $F \in \mathcal{F}_0$, but does not specify which F [15].

Discriminating statistic T. A discriminating statistic T is a random variable, that quantifies some aspect of a time series X. A difficulty arises when the null hypothesis is composite, and it is not clear at first sight which process should be considered. Therefore, T has to be *pivotal*, which means that the distribution of T is the same for all members F of the family of processes $\mathcal{F}_0$ consistent with the null hypothesis. If this condition is met, it is not necessary to know the precise $F \in \mathcal{F}_0$, any $F \in \mathcal{F}_0$ can be used. Otherwise it would not make sense to compare the obtained distribution of T to the quantifier obtained for the data set X. To be pivotal is a very strict criterion, and is often fulfilled only in the asymptotic limit as the size n of X approaches infinity. If the condition to be pivotal is met for $n < \infty$ it can be shown that a Monte Carlo test can be even more accurate than a corresponding asymptotical test [15] [42]. The Monte Carlo method used in this paper has the advantage that the test statistic does not necessarily have to be pivotal. This is further discussed in the next chapter. When a test statistic is different for the observed data than would be expected under the null hypothesis, then H_0 can be rejected [9]. This only means that it is *unlikely* that H_0 is correct, but does not prove it to be wrong.

Errors. This uncertainty leads to another main part of statistical testing, the errors. Basically, there are two possibilities for making a mistake when testing a certain null hypothesis: first, to reject H_0 when it is in fact true, and the second, failing to reject H_0 when it is in fact false. The former is called *Type I error*, the latter *Type II error*. In order to evaluate a test, it is designed with a predefined *size* α, which corresponds to the expected rate of Type I errors. Hence, the smaller α is set, the less probable a Type I error is. Usually, $\alpha = 0.05$ is the upper limit that is considered *significant*. A test is well–posed if α corresponds to the actual probability of committing a Type I error. This probability can only be calculated a posteriori, therefore a test has to be applied in practice before being evaluated. The probability of Type II errors occuring is commonly denominated β, and $1 - \beta$ is called the *power* of the test.

3.2 Testing of Nonlinearity

Very useful advice from many authors on testing a time series for nonlinearities is the following: always consider a linear approach first. Occam's razor tells us that we should rule out the simplest explanation. In this case, an underlying linear process, before we venture to construct more complicated models. In many cases, though not obvious, a linear model may be good enough to describe the dependencies in X. Even if a system contains nonlinear components, this does not prove that this nonlinearity is also reflected in a specific signal we measure from that system. In particular, we do not

know if it is of any practical use to go beyond the linear approximation when analyzing the signal. After all, we do not want our data analysis to reflect our prejudice about the underlying system, but rather to represent a fair account of the structures that are present in the data. Consequently, for data driven analysis, the application of nonlinear time series methods has to be justified by establishing nonlinearity in the time series [25]. The benefits of assuming an underlying linear process are numerous. There is more literature, theory, and knowledge disposable, as well as a variety of software solutions and packages for linear analysis tools. When the linearity of a data series is not visible at first sight, e.g. when the time series is short and unrepeatable, one may be misleaded to suppose an underlying nonlinear process. The resulting scenario of applying a wrong model is drawing apparently wrong assumptions about the elementary procedures behind the data series. Needless to say, this should be avoided. Hence, the usage of nonlinear methods has to be justified, which in the language of statistics means nonlinearity has to be tested against linearity before applying nonlinear techniques.

In [5] Granger and Teräsvirta divide the nonlinearity tests into two broad categories:

- tests without any specific nonlinear model in mind against which could be tested
- tests against a predefined nonlinear alternative

For further insights into this categorization, the reader is referred to chapter 6 of [5]. The authors provide several tests against unspecified, as well as certain specified nonlinear models.

Different univariate and multivariate statistics will be presented in their respective chapters. For the purpose of nonlinearity testing, quantities should be used that are particularly powerful in discriminating linear dynamics and weakly nonlinear signatures — strong nonlinearity is usually more easily detected [25].

3.3 The Idea of Surrogate Data

For finite sample size, where only one realization X is available, it is often not possible to derive the distribution of a given statistic T_X analytically. In order to perform a hypothesis test the sampling distribution of T_X under the null hypothesis H_0 has to be determined. H_0 is the assumption of linearity and certain other restrictions of the time series. To be able to approximate the distribution of T_X, *randomization* and *Monte Carlo resampling* are used. These methods are statistical sampling techniques, which are often used when analytical solutions are difficult to find. They are based on (pseudo)

random sample computations. For further information, the reader is referred to B.Efron's paper [27]. With the use of Monte Carlo methods, it is possible to estimate the distribution of T_X numerically. The idea is to compute values T_S for a sufficient number N of different random realizations of the null hypothesis, and to empirically estimate the distribution of T_X from this ensemble of values. One has to specify what a 'sufficient number N' means in terms of statistical hypothesis testing. When data created by Monte Carlo methods is tested, the first decision to be made is about the test statistics T_X and T_{S_i}. Depending on the statistics, the test is either one or two–sided. The size α, i.e. the significance level of a test, gives the probability of incorrectly rejecting the null hypothesis. In case of a one–sided test, the null hypothesis H_0 is rejected if T_X is amongst either the $(N+1)\alpha$ biggest (or smallest) values of the T_{S_i}. Therefore, one has to create at least $N = \frac{1}{\alpha} - 1$ surrogate data series. In case of a two–sided test, H_0 is rejected if T_X is amongst either the $\frac{(N+1)}{2}\alpha$ biggest or smallest values of the T_{S_i}. Hence the sufficient amount of realizations is $N = \frac{2}{\alpha} - 1$ [1]. For $\alpha = 0.05$ this means at least 19 or 39 realizations have to be created. In order to gain better test power, it is advisable to create more than this mimimal sufficient amount [25]. For further insight, the reader is referred to Marriott who investigates the effects of using a higher number of simulations in [14].

Surrogate data. The method used for creating the different realizations is the *method of surrogate data*, first published by James Theiler et al. in [9]. The term *surrogate* originates in the Latin verb *surrogare*, which means 'to put in another's place'. A *surrogate data series* is an artificial time series, created with Monte Carlo methods, of the same dimension as X, which is consistent with the null hypothesis H_0. If a sufficient number N of surrogates has been created, it is possible to estimate the distribution of T_X by computing the test statistics T_{S_i}, $i = 1, \ldots, N$, and using the empirical distribution. This technique can be regarded as an application of the *bootstrap method* of modern statistics [9]. This name was first used by Efron in [12] in 1979. It is a general approach to statistical inference based on building a sampling distribution for a statistic by resampling from the data at hand. The term bootstrapping is an allusion to the expression 'pulling oneself up by one's bootstraps'. The sample data are used as a population from which repeated samples are drawn [13]. The method of surrogate data is described as a statistical approach for identifying nonlinearity in time series. The actual quantification of nonlinearity within time series by statistical testing of surrogate data is the purpose of this section. It is an indirect approach, attempting to exclude that the data are linear [40]. In general, this method can be used with any nonlinear statistic that characterizes a time series by a single number. The basic assumption is that there is only one finite time series X available. In practice, this may be the case if measurement is very

expensive, or if the observed phenomenon only occured once, for example geological or astronomical measurements. The (wide sense) stationarity of the underlying process is a very restrictive, but important assumption. In [9], the authors refer to several papers that discuss problems in nonlinear statistics that arise with nonstationary data. Because nonstationarity is difficult to detect and to specify, it may be often misinterpreted as nonlinearity. In this thesis, methods to create surrogate data for an underlying process with nonstationarities will be addressed as well.

In order to test a time series X for nonlinearity, surrogate data S_i with the same *linear* properties as X have to be created. The methods are commonly divided into two types [1]:

Typical realization. This type of realization pursues the approach of defining null hypotheses H_0 over the class of linear models. Many other traditional bootstrap methods also use explicit model equations that have to be extracted from the data. The surrogate data series are N different realizations of one of the models. Typical realization is only possible, if the unknown parameters can be estimated through fitting to X. This is one of the biggest disadvantages of the typical realization: it may not be possible to estimate the parameters with sufficient precision. Furthermore, there has to exist a known model dependent on the null hypothesis. In case of testing for nonlinearity, a model for linear processes is the *ARMA(p,q) model* (see equation 2.4).

The approach used in [9] falls into the category of typical realization: the authors specify a well–defined underlying linear process, and determine the distribution of the quantity of interest, i.e. T_X, for an ensemble of surrogate data sets. These are different realizations of the hypothesized linear stochastic process. By means of the surrogate data series, they estimate the deviation from the standard deviation of the estimated quantities T_{S_i} numerically.

Constrained realization. This approach doesn't follow the path of estimating properties through fitting. Instead, the desired properties are *exactly* re–produced on the surrogate data series. When testing for nonlinearity, the mean, variance, and also the probability distribution of the data points can be imposed in this way onto an artificial randomized time series. According to the assumption that X consists of independent draws from a fixed probability distribution, surrogates could be created through randomly shuffling the original time series. The maximum number of different surrogates would be $n!$ in this case. If there are significantly different serial correlations in the data and the shuffles, the hypothesis of independence can be rejected. Constrained realizations are obtained by creating permutations *without re-*

placement. The surrogates are constrained to have exactly the same values as the data, just in random temporal order. One could also have used the data to infer the probability distribution and drawn new time series from it. These permutations *with replacement* would then follow the way of typical realization [25].

In [15], Theiler and Prichard advocate that the constrained realization should be preferred in general, as it comparatively gives better test results. They find the reason is dua to whether a test statistic is *pivotal* or not, i.e. its distribution is the same for all processes consistent with the null hypothesis. The typical realization method requires that the discriminating statistic satisfies this property. The constrained realization approach, on the other hand, does not share this requirement, and can provide an accurate and powerful test without having to sacrifice flexibility in the choice of a discriminating statistic. In the context of testing nonlinearity, the following finding of [15] is of assistance: accurate and powerful statistical testing for nonlinearity requires either a pivotal statistic T_X which does not depend on any autocorrelation, or a Monte Carlo method which constrains the surrogate data to match the sample autocorrelation of the original data. The algorithms presented in the next chapter are in line with these findings.

To sum up the idea, surrogate data are artificial time series with two basic properties:

- They have no dynamical nonlinearities. By construction, a surrogate is equivalent to passing Gaussian white noise through a linear filter, that reproduces the linear properties of X on it.
- The process that generates surrogate data is stationary. If the linear filter is seen as this process, it does not change during the duration of the surrogate data [60]. As mentioned before, there are nonstationary extensions applied as well.

Chapter 4

Nonlinear Dependencies within Time Series

4.1 Fourier–based Algorithms

The focus of this section is on algorithms for the creation of surrogate data. Most of these algorithms are motivated by the ideas of [9] and [10], and summarized in [1]. Basically, the standard procedure is to reproduce the auto–covariance on the surrogate data, and thus preserve the linear properties of a time series X. This is relatively straight–forward, and it does not require any model–fitting. In this chapter, the data are univariate of length $n \in \mathbb{N}$:

$$X = (x_1, \ldots, x_n) \in \mathbb{R}^{(1 \times n)} \tag{4.1}$$

A common estimator for the auto–covariance is

$$\gamma_X(\tau) = \frac{1}{n-\tau} \sum_{j=1}^{n-\tau} (x_j - (\bar{x}))(x_{j+\tau} - (\bar{x})) \quad , \quad \tau \in \{0, \ldots, n-1\} \tag{4.2}$$

where τ is the time lag. In order to reproduce γ_X on a surrogate, the time series X is transformed into frequency domain, and a relation of discrete Fourier amplitudes and auto–correlation of X is used.

The information that X contains can either be described on a time scale or in frequency domain. A commonly used transformation from time to frequency domain is the *discrete Fourier transform (DFT)*:

$$\mathcal{F}(x_t) = \tilde{x}_k = \frac{1}{\sqrt{n}} \sum_{t=1}^{n} x_t e^{-it\frac{2\pi k}{n}} \quad , \quad k = -\lfloor (\frac{n-1}{2}) \rfloor, \ldots, \lfloor (\frac{n}{2}) \rfloor \tag{4.3}$$

The DFT corresponds to a finite number of sample values at equidistant frequencies, the *Fourier frequencies*

$$\lambda_k = \frac{2\pi k}{n} \quad , \quad k = -\lfloor (\frac{n-1}{2}) \rfloor, \ldots, \lfloor (\frac{n}{2}) \rfloor \tag{4.4}$$

where $e^{i\lambda_k t}$ are the primitive n^{th} roots of unity. $\{e^{i\lambda_k t} : k = -\lfloor(\frac{n-1}{2})\rfloor, \ldots, \lfloor(\frac{n}{2})\rfloor\}$ build an orthonormal basis of $\mathbb{C}^n$ [11].

$$\frac{1}{n}\sum_{t=1}^{n} e^{-i\lambda_k t} e^{i\lambda_j t} = \begin{cases} 1 & for\ \lambda_j = \lambda_k \\ c = \frac{e^{i(\lambda_j-\lambda_k)}}{n}\frac{1-e^{i(\lambda_j-\lambda_k)n}}{1-e^{i(\lambda_j-\lambda_k)}} & else \end{cases} \tag{4.5}$$

with $c \overset{n\to\infty}{\longrightarrow} 0$. Hence, the $\{e^{i\lambda_k t}\}$ are orthonormal in t. The complex $\tilde{x}_k$ express X in terms of a sum of sinusoidal components. They can be described by their *amplitudes* $A_k = |\tilde{x}_k|$, and their *phases* $\phi_k = arctan(\frac{Im(\tilde{x}_k)}{Re(\tilde{x}_k)})$. The *inverse discrete Fourier transform (IDFT)* of $\tilde{x}_k$ is defined as

$$x_t = \sum_{k=1}^{n} \tilde{x}_k e^{ik\frac{2\pi t}{n}} \quad , \quad k = -\lfloor(\frac{n-1}{2})\rfloor, \ldots, \lfloor(\frac{n}{2})\rfloor \tag{4.6}$$

The reason for the transformation into frequency domain is a relation of the amplitudes A_k and the auto–correlation function ϱ_X. The *periodogram* $I_n(\lambda_k)$ is defined as the averaged squared Fourier amplitudes, and is a non–consistent estimator for the power spectrum. Furthermore, the power spectrum nearly equals the DFT of the time–averaged auto–correlation function $\varrho_X(t)$.[1] As a consequence, and due to equations 4.5,

$$\{A_k^2\} \approx \mathcal{F}\{\varrho_X(t)\}. \tag{4.7}$$

holds for a real–valued wide sense stationary process with $\varrho_X(t) \in l^1(\mathbb{R})$[2]. Hence, the set of squared Fourier amplitude is estimator for the DFT of the time–averaged auto–correlation. These findings are summarized in [17]. Liu and Varwani extend the Wiener–Khinchin theorem on any random process in [16]. A certain symmetry exists for the $\tilde{x}_k$ of a real–valued signal X:

Theorem 4.1.1. *If x_t with $t = 1, \ldots, n$ is real–valued, the relation $\tilde{x}_{n-k} = \overline{\tilde{x}_k}$ holds.*

Proof. As a result of Euler's identity[3] and $\overline{e^{i\phi}} = e^{-i\phi}$, following equation holds:

$$\tilde{x}_{n-k} = \frac{1}{\sqrt{n}}\sum_{t=1}^{n} x_t e^{-it\lambda_n} e^{it\lambda_k} = \frac{1}{\sqrt{n}}\sum_{t=1}^{n} \overline{x_t e^{-it\lambda_k}} = \overline{\tilde{x}_k}$$

□

Conversely, if $\tilde{x}_{n-k} = \overline{\tilde{x}_k}$ is true $\forall k$, then the IDFT of $\tilde{x_k}$ is real–valued.

[1] According to the Wiener–Khinchin theorem: see [17] and [16]
[2] i.e. the space of sequences, whose series is absolutely convergent
[3] $e^{2\pi i} = 1$.

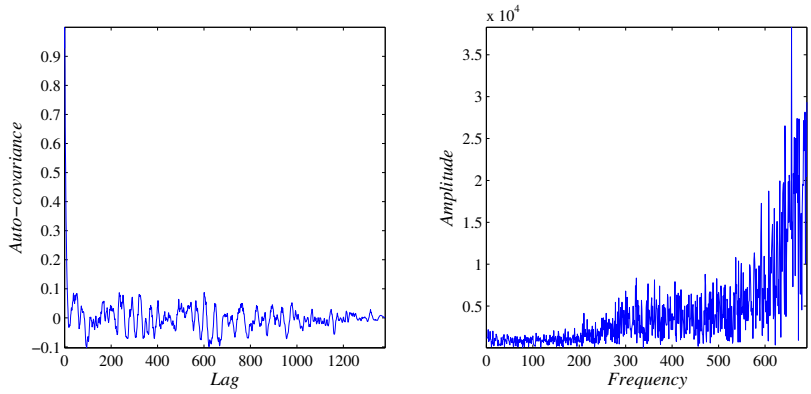

Figure 4.1: The normalized one–sided auto–covariance, and the corresponding squared one–sided Fourier amplitudes of a NO_X data series. The high frequencies have more contribution to the power spectrum than the low frequencies, which indicates strong oscillations in the data.

4.1.1 Unwindowed FT Algorithm

The aim of this section is to create surrogate data series, which have the same Fourier amplitudes as X. First, X is transformed into frequency domain (see equation 4.3). Amplitudes stay unchanged, but the phases are replaced by random numbers $\phi \in [0, 2\pi)$. In order to construct real–valued surrogates, the ϕ must have the same symmetric character as the original phases. Finally, the constructed series has to be transformed back into time domain (see equation 4.6) to get a surrogate for X. Formally, the linear characteristics of X are specified by the squared amplitudes of the DFT

$$|\tilde{x}_k|^2 = |\frac{1}{\sqrt{n}} \sum_{t=1}^{n} x_t e^{2\pi i \frac{tk}{n}}|^2. \tag{4.8}$$

This is the periodogram estimator of the power spectrum. A surrogate s_t is readily created by multiplying the DFT of X by random phases, and then transforming the constructed series back to time domain:

$$s_t = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} e^{i\phi_k} \tilde{x}_k e^{-2\pi i \frac{tk}{n}} \tag{4.9}$$

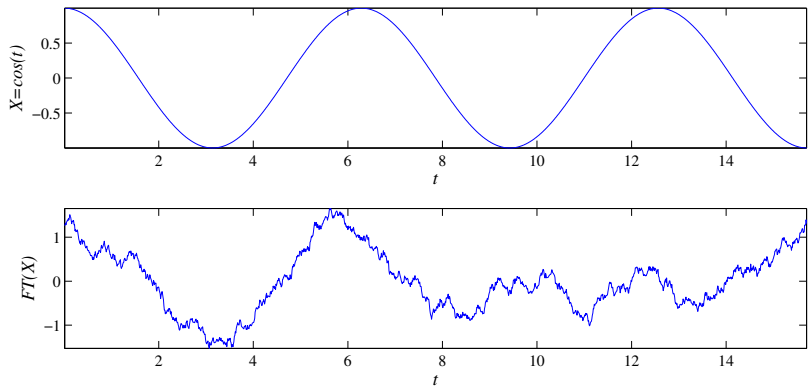

Figure 4.2: The upper pane shows a simple cosine function in the interval $[0, 5\pi]$ measured at a rate of 0.01. The lower pane displays a surrogate that was created by the unwindowed FT algorithm. It demonstrates the occuring spurious high frequencies due to a jump discontinuity in the data.

where $0 \leq \phi_k < 2\pi$ are independent uniform random numbers [25]. The randomization of the phases and the IDFT back to time domain can be interpreted as white noise, that is filtered by a linear Gaussian wide sense stationary filter. Therefore, the null hypothesis, that corresponds to these surrogates, is

> $\boldsymbol{H}_0$: X is realization of a linear Gaussian wide sense stationary process.

This H_0 is highly restrictive. In [1], Schmitz gives the example, that X is obtained by a nonlinear measuring function. If the distribution of X differs from a Gaussian distribution, a test would correctly abandon H_0. The matter with the unwindowed Fourier transform algorithm (FT) is the implicit assumption, that the data represent one period of length n of a ∞–dimensional series. In case of a jump discontinuity from x_1 to x_n, spurious high frequencies can be introduced as artifact on the surrogates. Figure 4.2 exemplifies this effect for a simple cosine function. A way to avoid these spurious frequencies is to shorten the time series to a length $\tilde{n}$, with $x_1 \approx x_{\tilde{n}}$. Because the time series is assumed to be stationary, it only has to be long enough[4] to make this solution work. It should be longer than the coherence time of any given frequency as well [9]. The effect of this procedure is illustrated in figure 4.3.
Another matter is the quality of the amplitude estimation for the power spectrum [1]. Other possibilities to numerically estimate the power spectrum are

[4] i.e. longer than its dominant frequency

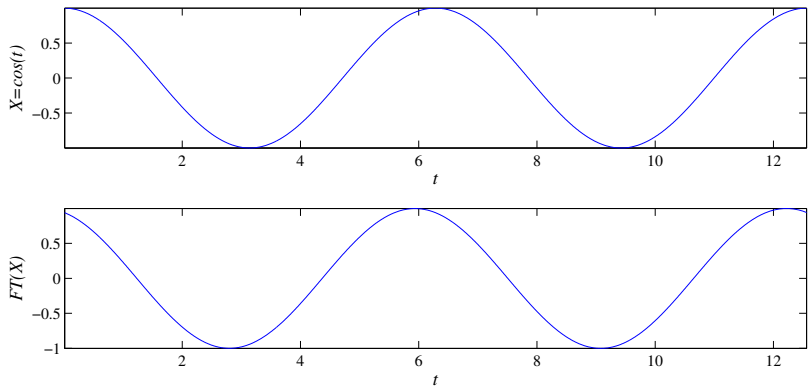

Figure 4.3: The upper pane shows the cosine function in the interval $[0, 4\pi]$. With this, $x_1 \approx x_n$ holds. The surrogate in the lower pane is as smooth as the original data set.

discussed by Press et al.. Besides the commonly used fast Fourier transform (FFT) algorithm, the authors use a maximum entropy method for estimation as well. For further discussion, the reader is referred to [54].

4.1.2 Windowed FT Algorithm

Besides of shortening the data, the issue of a jump discontinuity could also be adressed by the introduction of a weight function

$$w(t) = \alpha \sin(\frac{\pi t}{n}) \quad , \quad w(1) = w(n) = 0, \alpha \in \mathbb{R} \tag{4.10}$$

This strategy is called *windowing*, and it represses the jump from x_1 to x_n. Thereby, the problem of residual high frequencies has vanished, but additional low frequencies are introduced by the power spectrum of the weight function $w(t)$. Therefore, windowing does not solve the problem of additional 'noise' in the frequencies, but only changes the trigger of it. Theiler et. al. set the magnitude of the offending frequency to zero, but if there is significant power at that frequency in X, it too will be suppressed [9]. However, the issue of the FT algorithm still exists for the WFT algorithm. Hence, this method is not applied in this thesis. In several cases, the following algorithms provide much better results.

4.1.3 Amplitude Adjusted FT Algorithm

The null hypothesis, that corresponds to the amplitude adjusted FT algorithm (AAFT), is

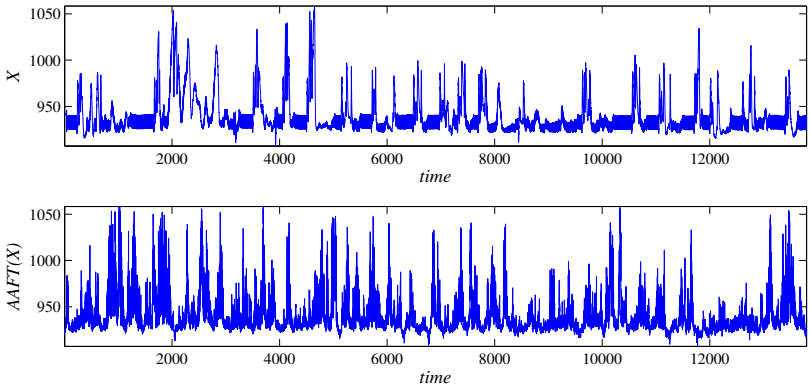

Figure 4.4: The upper pane displays manifold pressure observations in a Diesel engine, measured by frequency 10Hz. The lower pane shows a AAFT surrogate for the measurements.

> $\boldsymbol{H}_0$: X is a static monotonic nonlinear transformation of a linear Gaussian wide sense stationary process.

This is true, if a linear process is observed by a nonlinear measure function. A surrogate according to this H_0 is constructed as follows: First, n Gaussian random numbers y_i[5] have to be computed. Then, these y_i are rescaled according to the order of the x_i. Hence, the smallest y_i has to have the same index as the smallest x_i, the second smallest y_i the same as the second smallest x_i,.... The elements of the re–arranged sequence are denoted as $\widetilde{y}_i$. After that, the (W)FT algorithm is applied on $\{\widetilde{y}_i\}_{i=1}^n$. Finally, the x_i are re–ordered according to the order of the (W)FT series, in order to get a surrogate for X. The point–wise distribution of the surrogate matches exactly with the point–wise distribution of X. Moreover, the autoc–ovariances are approximately identical. X and a AAFT surrogate are displayed in figure 4.4 Although the null hypothesis has been generalized, it is still not very flexible. The approximation of the unknown monotonic transformation is not exact for $n < \infty$, thus there is an additive spectrum of the emerging residuals. The phase randomization preserves the Gaussian distribution only on average. The precision of the spectra match gets worse, as n gets smaller. In other words, the AAFT algorithm, though asymptotically correct, cannot reproduce the spectrum of the data very precisely. There is a certain bias towards a too flat spectrum [10]. A comparison between the auto–covariances of data X, and a typical surrogate are illustrated in figure 4.5.

[5] $i = 1, \ldots, n$

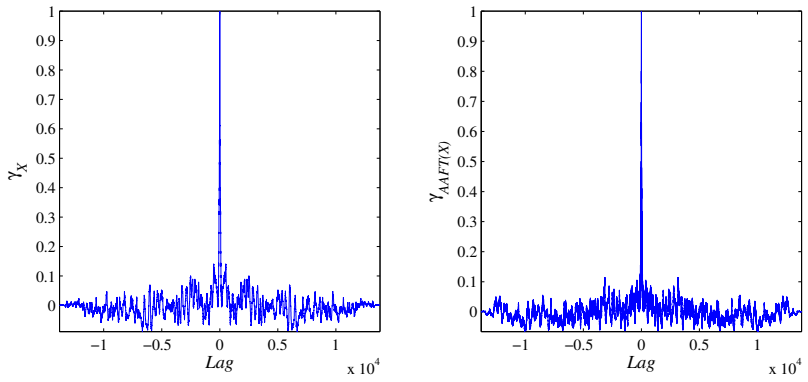

Figure 4.5: The normalized auto–covariances of X and $AAFT(X)$ (as seen in figure 4.4).

4.1.4 Iterative AAFT Algorithm

In [10], another surrogate algorithm is presented. The iterative AAFT algorithm (IAAFT) allows a more general null hypothesis

> $\boldsymbol{H}_0$: X is a time–independent static instantaneous invertible transformation of a linear Gaussian wide sense stationary process.

Note, that instead of monotony of the measure function, an invertible tranformation is considered now. This transformation has to be continuously differentiable, but not monotonic. [10]. The advantage of the IAAFT is a better approximation of the amplitudes, and therefore of the auto–covariances. In [1], the algorithm is described as follows: first, the Fourier amplitudes of X are calculated. The x_i are rearranged in ascending order. The following iteration method consists of two consecutive procedures. It starts with a random permutation of X, denoted as $S^{(0)}$. In iteration step i, $S^{(i)}$ is Fourier transformed, and the amplitudes are replaced by the original amplitudes of X. Then, the inverse Fourier transform is computed to get $\widetilde{S}^{(i)}$. By rearranging the order of $\widetilde{S}^{(i)}$ according to the sorted X, one gets $S^{(i+1)}$. After each step, the remaining discrepancy between the spectra is checked. Eventually, the transformation towards the correct spectrum will result in a change, that is too small to cause a reordering of the series. Thus, after rescaling, no change happens any more, and the iteration procedure has reached a fixed point $S^{(i+1)} = S^{(i)}$. $S^{(i)}$ is the generated surrogate. The nonlinear measure function is approximated with this algorithm. The result cannot be exact for finite time series. The IAAFT follows the constrained realization approach: the variations in spectrum and distribution within the class, defined by the null hypothesis, are suppressed by constraining the surrogates, in order to

have approximately the same power spectrum, as well as the same distribution as X [10]. The final accuracy, that can be reached, depends on the size and structure of the data, but is sufficient for hypothesis testing in general [25].

Note, that although the null hypothesis of the IAAFT algorithm is less restrictive, its rejection does not imply nonlinear dynamics. For instance, noninstantaneous measurement functions are not included, and would (correctly) lead to a rejection of H_0, although the underlying dynamics may be linear [10].

4.1.5 Constraints

The main disadvantage of the presented algorithms is the constrained class of null hypotheses. Even the most general H_0 is still highly restrictive. This is not the only constraint, Schmitz adresses two other restrictions and disadvantages of the Fourier methods in [1]:

Periodicity. As mentioned before, the Fourier transform is periodic in the frequencies k with period n. Therefore, by the presented standard Fourier methods, not the estimated auto–correlation ϱ_X is reproduced on the surrogate, but the estimated *periodical auto–correlation* ϱ_{X_P}. The corresponding estimator for the *periodical auto–covariance* writes

$$\gamma_{X_P}(\tau) = \frac{1}{n}\sum_{j=1}^{n}(x_j - (\bar{x}))(x_{mod(j+\tau-1,n)+1} - (\bar{x})) \tag{4.11}$$

In general, X is not a realization of a period of an infinite periodical process. The difference between $\gamma_X(\tau)$ and $\gamma_{X_P}(\tau)$ produces unmeant artefacts in the surrogates, especially if x_1 and x_n vary significantly [26]. Windowing, or attaching zeros to the ends of X, implicate that this modified X cannot be an invertible transform of a process any more. Thus, these techniques cannot be applied. Another possibility to avoid unmeant artefacts is to cut X to match its ends. If n is large enough, this is an easy way to improve quality of the test results. The slopes should match as well to avoid an unwanted peak, especially if X is very smooth.

Non–Stationarity. All surrogates created with the presented methods are stationary: the Fourier phases have been replaced by random phases. The IDFT is then the linear combination of independent, periodical, stationary functions, and is therefore also stationary. A non–stationary X may often lead, depending on the test statistic, to a rejection of H_0, and the validity of the test can be questioned. The data applied for this thesis are measured in a Diesel engine. As its level of activity is changing during a simulation,

it would be unreasonable to expect unchanging signals [60]. Hence, before creating surrogates, X should be inspected for non–stationarity. If n is big, it is possible to divide X in segments, and to compare key figures like means, or variances on them. If they differ a lot from each other, non–stationarity could be a possible reason. Figure 4.6 displays the means calculated on segments of a time series. Also, only a single stationary segment of X can be tested. A rejection of H_0 would justify the use of nonlinear techniques. Schmitz discusses a possibility to create non–stationary surrogates for short X, when non–stationarity cannot be excluded. The idea is to divide X in segments, and create a surrogate for each of them. By stringing them together, a surrogate with the same non–stationarities as X on the segment length is constructed. The difficulty is to choose the right segment length. If it is too short, then too much information is lost, and if it is too long, then the non–stationarities cannot be reproduced. For further reading I refer to [20], where Schmitz and Schreiber give examples for successfully creating surrogate data with this method. They use the iterative algorithm for creating the surrogates, and test for an ARMA process with slowly varying coefficients. The issue of nonstationarity is addressed in detail later in this chapter.

4.2 Applied Simulated Annealing

In this section, a more flexible surrogate algorithm is presented. This method enables to reproduce γ_X on a surrogate with arbitrary precision, and H_0 may include known non–stationarities as well.

The idea of *applied simulated annealing* is to translate the problem of creating surrogates with certain characteristics into a single, real–valued cost function E. This cost function is minimized by a simulated annealing algorithm. Usually, the set of surrogates, that have exactly the same characteristics as X, consists of trivial translations of X only. Therefore, not the global optimum of E, but a local optimum close to the global optimum is intended to be reached. Any characteristic, that can be described by a single number, can be reproduced on a surrogate in this way. Kirkpatrick et al. have introduced this application of simulated annealing in [2]. A disadvantage of simulated annealing is its high computing effort. However, if the observation of a phenomenon is very expensive, or if this phenomenon only occurs once, then the high computing effort may be justified. Besides, the computing power increases as technology proceeds.

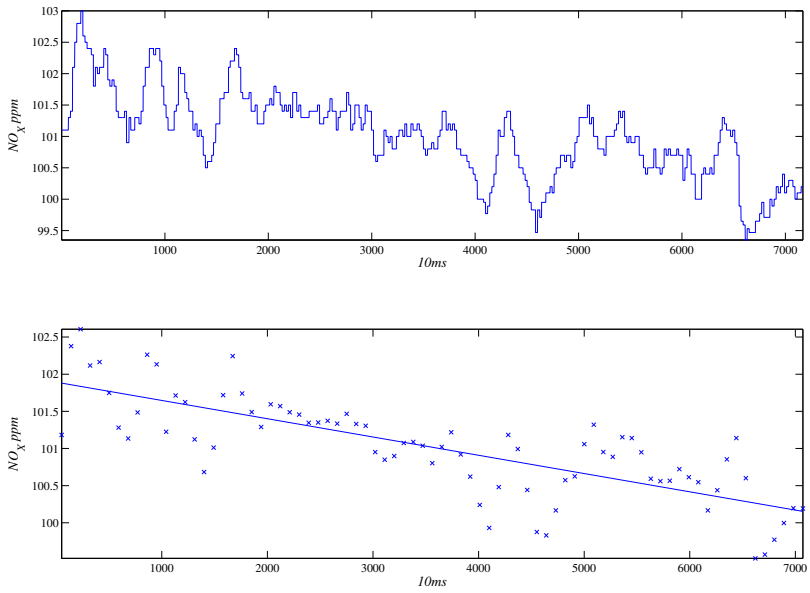

Figure 4.6: The upper pane displays NO_Xppm data with a drifting mean. The time series was divided into segments with length 100 and overlap 10, and the mean was calculated segment–wise. The result is shown in the lower pane, together with a linear least square regression line to highlight the slope.

4.2.1 Cost Function

The method of constructing surrogates by the useage of applied simulated annealing is described by Schreiber in [19]. This section outlines the basic ideas of this paper.
First, a set of characteristics $\{C_{X,i}\}_{i=1}^{I}$[6], that a surrogate $S \in \mathbb{R}^{(1\times n)}$ is meant to feature, has to be defined. C_{X_i} is exactly reproduced on a surrogate, when

$$C_i := C_{S,i} - C_{X,i} = 0 \tag{4.12}$$

holds[7]. All C_i have to be summarized to a single, real–valued cost function

$$E = \left(\sum_{\tau=0}^{n-1} |C_{S,i}(\tau) - C_{X,i}(\tau)|^q \right)^{\frac{1}{q}} \tag{4.13}$$

where $q \in \mathbb{N}$ defines the norm.[8] At the global optimum, $E = 0$ holds, and the features of X are exactly reproduced on S. In thermo–dynamics, E is

[6] $i = 1, \ldots, I < \infty$
[7] $C_{S,i}$ describes the i^{th} characteristic on S.
[8] e.g. Manhattan–norm for $q = 1$, L^2–norm for $q = 2$, or maximum–norm for $q \to \infty$.

interpreted as energy in a molecular system, and T as temperature, that controls the energy decrease. In principle, any function $C_{S,i}(\tau)$ with global minimum at $C_{X,i}(\tau)$ can be used. In testing for nonlinearity, $C_{X,i}$ and $C_{S,i}$ are γ_X and γ_S. In this way, linear features of X are reproduced on surrogates, and the use of $\gamma_{P,X}$ (see equation 4.11) is avoided.

4.2.2 Optimization by Simulated Annealing

Minimization of a single real–valued cost function is an entirely different approach than the Fourier methods. One of the major difficulties of minimizing a cost function is the ∞*–dimensional solution space* of the problem[9]. An expedient is the introduction of additional constraints, in order to reduce the solution space dimension to a finite number. A common requirement is that the surrogates have to follow the same single time probability distribution as X [19]. This is the same constraint as used in the standard Fourier methods. Only permutation of X feature the same single time probability distribution as X. Thus, solution space dimension is reduced to the number of possible permutations of X, $n!$ [1].

Although solution space is not infinite any more, it is still vast for large n. The possibly *high number of local extremes* pose a problem for minimization of E. Many optimization algorithms do not provide the possibility to leave these local extremes once they are reached. These algorithms are constructed to respond to improvement of E in immediate proximity to the current position only. Simulated annealing offers the possibility to leave local extremes, and thus it is suitable for optimizing the constructed cost function.

The main problem of simulated annealing is, as already mentioned, its *high computing effort*. It is advisable to introduce a limit of sufficient precision to reduce CPU time. Several ways to introduce this limit will be further discussed in a later part of this section.

In [2], Kirkpatrick et al. divide the strategies for solving complex combinatorial optimization problems into two categories:

Divide– and Conquer. A complex problem is divided into many small, hence often easier solvable, parts. The difficulty is to chose a suitable number of subsystems, the splitting points, and to not misinterpret results for the single small problems.

Iterative Improvement. Optimization starts with a known state of a system. A predefined operating mechanism continuously adapts the system

[9] A optimum has to be found within *all possible* surrogates.

state, until a cost reduction is attained. The system is in a new state, and the procedure starts again. This happens, until either the cost cannot be lowered any more, or a predefined limit is reached.

Applied simulated annealing falls into the latter category. It origins in thermo–dynamics, which is, among others, concerned with the behavior of atoms and molecules in a system. The considerations of Austrian physicist Ludwig Boltzmann laid the foundation for this discipline in the late nineteenth century. The energy E of a thermo–dynamical system depends on the constellation of its molecules at a given temperature T. The focus is on the relation between a decrease in temperature, and a corresponding change in energy. Each temperature–energy pair defines a probability factor

$$\beta = e^{-\frac{E}{T}}. \tag{4.14}$$

The *Boltzmann factor* is crucial for reaching a low temperature–energy state. It enables to leave any local extreme with certain possibility, if the temperature decrease is slow enough. In practice, the state equilibria are examined by cautious annealing. After heating a system, temperature is lowered slowly. Especially the stadium in close vicinity to 'freezing point'[10] of a system is critical. The temperature has to stay long time at this transitional phase. Otherwise, the substance may not reach an optimal state, and it has to be heated again. In industry, this method is used to correct defects in materials like alloys. Cautious annealing results in regular molecular structure.

By simulating this annealing procedure, combinatorial optimization problems can be solved. The real–valued cost function represents the energy of a thermo–dynamical system, and the $n!$ possible permutations can be seen as possible configurations of n molecules. By rearranging the molecules, energy varies. Control parameter is the temperature T, that decreases during the process. If only permutations that reduce E are considered, the probability of getting stuck in a local extreme is very high. *Metropolis algorithm* offers a way out of this issue: In [3], Metropolis et al. introduced this modified Monte Carlo method. The authors assume, that a substance is built of squares, and each square consists of n molecules. In order to find out more about the features of the system, Monte Carlo methods are applied. The authors choose a state with probability $e^{-\frac{E}{T}}$, and assign the same weights to each state. For a new state, the energy difference to the previous state ΔE is calculated. The probability of accepting a new configuration with given ΔE and T is

$$p_{(\Delta E,T)} = \begin{cases} 1 & for\ \Delta E < 0 \\ e^{-\frac{\Delta E}{T}} & for\ \Delta E \geq 0 \end{cases} \tag{4.15}$$

[10] $T = 0$

In case $\Delta E < 0$, a state is accepted[11]. The crucial case is $\Delta E \geq 0$, where a state is accepted with probability $e^{-\frac{\Delta E}{T}}$. This scheme can solve the issue of getting stuck in local extremes. It enables the energy function to walk 'uphill' with a certain probability. Numerically, this is implemented by generating uniformly distributed random numbers in $[0, 1)$. If a random number is smaller than $e^{-\frac{\Delta E}{T}}$, then a new state is accepted. Note, that the Metropolis algorithm is asymptotically optimal with probability 1 [29]. Van Laarhoven et al. describe the convergence conditions for the procedure in [33].

The Metropolis algorithm enables to solve combinatorial optimization problems by simulating an annealing process. In [1], this method is used to create surrogates with nearly the same characteristics as X.

4.2.3 Annealing Scheme and Parameter Tuning

In practice, the aim is to find a local optimum of E, that is sufficiently close to the global optimum. Thus, characteristics of a surrogate should not be identical, but very similar to characteristics of X. The finite–time characteristics of simulated annealing algorithm depend on several control parameters. Deliberate choice is the key aspect of the procedure. Control parameters are discussed in this section.

The set of parameter choices is divided into two categories [4]:

Generic decisions. This category contains all parameters of the *annealing scheme*[12]:

- T_0... initial temperature
- L_k... length of k^{th} Markov chain[13]
- T_k... temperature function
- truncation criteria

A good annealing scheme is characterized by both nearly optimal solutions and a short execution time [4].

Problem specific decisions. Problem specific decisions vary with the problem, and they consist of

- S_0... initial state of the system

[11] The aim is to minimize E.
[12] i.e. the way in which the temperature is manipulated
[13] i.e. the number of iterations, after which T decreases

- neighborhood generation
- $E_k \ldots$ energy function, and its evaluation

The parameter setting has to be followed by a set of experiments, in order to fine–tune the procedure.

Algorithm 4.1: Applied simulated annealing scheme.

Calculate $E(S^{(0)})$ and $T^{(0)}$
▷ Initial energy and temperature
repeat
  $S^{(k)} \longrightarrow \tilde{S}^{(k)}$: $s_i^{(k)} \leftrightarrow s_j^{(k)}$
  ▷ Switch indices $i, j \in 1, \ldots, n$, $i \neq j$
  Calculate $E(\tilde{S}^{(k)})$
  ▷ Energy of the surrogate candidate
  if $E(\tilde{S}^{(k)}) \leq E(S^{(k)})$ **then**
    $S^{(k+1)} \leftarrow \tilde{S}^{(k)}$ ▷ Configuration accepted
  else
    if $e^{(-\frac{E(\tilde{S}^{(k)})-E(S^{(k)})}{T^{(k)}})} > r \in [0,1)$ **then**
      $S^{(k+1)} \leftarrow \tilde{S}^{(k)}$ ▷ Configuration accepted
    else
      $S^{(k+1)} \leftarrow S^{(k)}$ ▷ Configuration rejected
    end if
  end if
  if $l = L_k$ **then**
    $T^{(k+1)} \leftarrow \alpha T^{(k)}$ ▷ Decreasing the temperature
    $l \leftarrow 0$
  else
    $T^{(k+1)} \leftarrow T^{(k)}$
  end if
  $k \leftarrow k + 1$
  $l \leftarrow l + 1$
until truncation condition
return

In [4], Vidal divides simulated annealing into *inhomogeneous versions* and *homogeneous versions*. The difference between the two procedures is their different annealing schedule. The inhomogeneous algorithm decreases T_k after each transition, whereas the homogeneous algorithm decreases T_k after a number of transitions L_k. In other words, the homogeneous algorithm is repeated L_k times at each temperature. This allows to 'escape' local extremes easier than the inhomogeneous algorithm. The inhomogeneous algorithm is

a special case of the homogeneous algorithm with $L_k = 1 \; \forall k = 1, \ldots, k_{max}$. A pseudo–code for simulated annealing is shown in algorithm 4.1.

There is a broad range of possible realizations of this scheme. Basically, they can be divided into two groups [4]. *Fixed annealing schedules* are characterized by a fixed initial temperature T_0, a constant temperature decrement $\alpha \in (0, 1)$, and a constant L_k. In order to choose appropriate parameters, they have to be tested in several configurations. Due to high computing effort of simulated annealing, this is not practicable in many cases. Cooling schedules with parameters that adapt themselves during the process are called *self–adapting schedules*. They are more efficient, and less sensitive to initial parameter values.

The following three approaches describe a fixed, a self–adaptive, and a 'hybrid' cooling scheme.

Schmitz uses a fixed annealing schedule in [1]:

Initial temperature. Initial temperature T_0 has to be chosen high enough. Otherwise, the algorithm is likely to get stuck in a local extreme soon. T_0 should be higher than the maximal reachable energy–difference ΔE, in order to attain a high acceptance rate. A possibility to estimate ΔE and T_0 is to compute $m \in \mathbb{N}$ independent random states, and to calculate the energy–differences $\Delta E_{0,m}$ of them. T_0 is then the maximum of these $\Delta E_{0,m}$. According to Schmitz, practical applications show that this value is too high, because no energy decrease happens at the beginning. In general, more than one surrogate is needed. The temperature, at which the energy of the first surrogate decreases, could be used as benchmark for T_0 for other surrogates. In this way, redundant calculations can be avoided.

Cooling process. If temperature is lowered too fast, the algorithm is likely to get stuck in a local extreme with inadequate value. On the other side, if temperature is lowered too slow, then the result is in general better, but CPU time increases drastically. Schmitz uses an *exponential cooling scheme*, as described in [2]:

$$T_0 = \max_m(E_{0,m}) - \min_m(E_{0,m}) \tag{4.16}$$

$$T_{k+1} = \alpha T_k \tag{4.17}$$

$$\alpha \in (0, 1) \quad \text{a priori fixed} \tag{4.18}$$

The decrement $\alpha \in (0, 1)$ can be chosen randomly, the smaller α, the faster the annealing. T_k decreases either after L_k steps, or after the number of accepted states at T_k L_k^{acc}. L_k and L_k^{acc} should bear a relation to the length

of X. Schmitz chooses $L_k = 10n$ and $L_k^{acc} n$ as boundaries. He also introduces a lower limit for the energy as truncation criterion. This limit is given by the requested exactness of the surrogate properties. Another possibility is to use L_k^{acc} as a benchmark. If L_k is reached, while still $L_k^{acc} = 0$ holds, then no more changes are accepted.

Initial solution. Due to the constraint, that point–wise distributions of X and S have to match, the generation of the initial state[14] is comparatively easy. $S^{(0)}$ is a random permutation of X. Finding an appropriate way to come to the next step is more difficult. Energy decrease, that results of swapping the positions of only two data points, may be too small. Then again, the swap of several indices at the same time may have a negative impact on the final result.

Huang et al. give an example of a self–adapting schedule in [7]:

Cooling process. Initial temperature T_0, and the temperature decrement α_k between step k and step $k+1$, are functions of the energy standard deviation σ_E.

$$T_0 = c\sigma_E \tag{4.19}$$

$$\alpha_k = \max(0.5, e^{(-\frac{\lambda T_k}{\sigma_E(T_k)})}) \tag{4.20}$$

$$T_{k+1} = \alpha_k T_k \tag{4.21}$$

$\sigma_E(T_k)$ denotes the energy standard deviation at temperature T_k. Huang et al. choose c high enough[15], in order to make sure that a deterioration of $3\sigma_E$ is accepted. $\lambda \leq 1$ regulates the speed of temperature decrease. The authors suggest to choose $\lambda = 0.7$, based on their practical experiments. The construction of α_k brings an expected energy mean at T_{k+1}, that lies in a range σ_E around the energy mean at T_k. σ_E is estimated in the same way as before.

In [28], Atqullah describes a 'hybrid' annealing scheme with elements of both fixed and self–adapting algorithms:

Temperature. Initial temperature T_0 has to be chosen high enough, so that virtually every transition to another state is accepted. All states exist with equal probability of acceptance. Temperature has to converge to zero as the algorithm progresses. In practice, temperature is reduced to a sufficiently small value, until no further significant energy decrease can be expected.

[14] i.e. a high–temperature state of the system
[15] $c = 10$

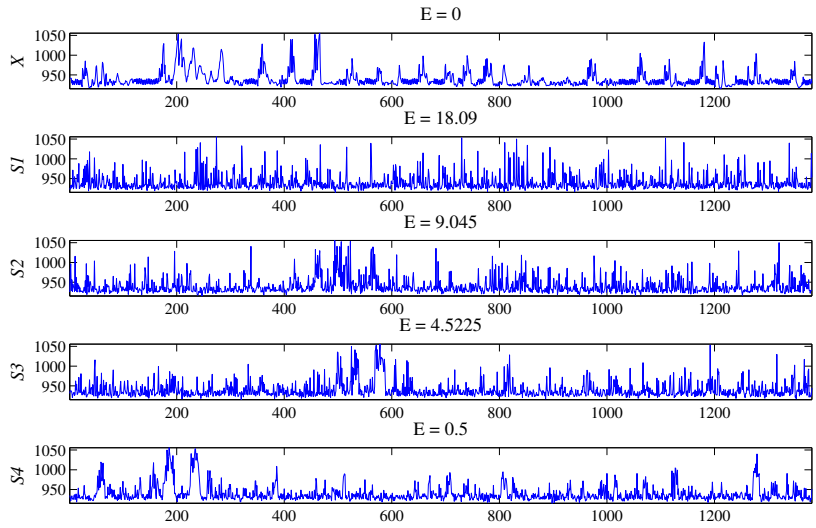

Figure 4.7: The panes show how a surrogate changes during the annealing process. The upper pane displays the original data X, i.e. manifold pressure measurements. The first surrogate represents the initial configuration. It is a random permutation of X. Above each pane is the corresponding value of the energy given. The corresponding auto–covariances to these four surrogates can be found in figure 4.9.

Markov chain. The number of transitions L_k at any specific T_k is the length of the k^{th} Markov chain. Chain length is governed by the 'closeness' of the current probability distribution to a predefined stationary distribution.

Temperature Decrement. A large decrement of T_k makes a large number of transitions necessary, in order to reach a quasi–equilibrium state. Most schedules adopt the strategy of small temperature decrements, in order to avoid long chains L_k.

Atqullah introduces the following parametric cooling schedule:

$$T_0 = \frac{(\overline{\Delta E} + 3\sigma_{\Delta E})}{ln(\frac{1}{A_0})} \tag{4.22}$$

$$T_{k+1} = \alpha T_k \tag{4.23}$$

$$\alpha = a^{-(\frac{k}{d \cdot k_{max}})^b} \tag{4.24}$$

$$L_k = \min(L_k^{acc}, L_k^{rej}) + B \tag{4.25}$$

where $\overline{\Delta E}$ is an estimator for the expected change in energy. It is the sum of

the absolute energy differences, divided by the number of trials applied. $\sigma_{\Delta E}$ is the sample standard deviation of $|\Delta E|$[16], and A_0 is the initial acceptance ratio[17]. T_0 is constructed to expect $A_0 \approx 1$. $a \in \mathbb{R}$ and $d \in \mathbb{R}$ are control parameters for the temperature decrement, $b \in \mathbb{N}$ depends on both of them, as well as on T_0, and the (predefined) final temperature. k is the number of chains executed, and k_{max} is the (predefined) maximal number of chains to be executed. By construction, the annealing is manipulated by the choice of a, b, and d. For example, $a = 3$, $b = 1$, and $f = \frac{1}{4}$ control the annealing in such a way that $T_k \approx \frac{1}{3}T_0$ holds at $\frac{1}{4}k_{max}$. L_k is defined by the number of accepted moves L_k^{acc}, the number of rejected moves L_k^{rej}, and an upper boundary B for both of them. Additional stop criteria are introduced, such as the requirement, that the absolute energy decrease over a predefined number of chains has to be beyond a predefined ϵ_E. This cooling scheme allows to control the maximal number of computing steps $(2B-1)k_{max}$. Figure 4.8 shows a temperature function, and the according energy created by this cooling scheme. The temperature reaches $\frac{T_0}{2}$ after $\frac{1}{100}$ of $k_{max}10n$, and thus describes a very fast cooling. $k_{max} = 10n$ enables the algorithm to reach an appropriate energy value. The boundary B for both accepted and rejected steps is 200. This is still high enough to get good results with comparatively small computing effort. For further discussion on this annealing schedule, the reader is referred to [28].

For further discussion of annealing schemes, and a comparison between them, the reader is referred to [30], [31], and [32].

4.2.4 Improving the Performance

As already mentioned, the disadvantage of simulated annealing is the required computing power, and the time to get good results. In [1], Schmitz discusses several possibilities to increase efficiency of the procedure, in order to reduce CPU time. Some of these ideas are described in the following paragraphs.

Calculating E_k. If simulated annealing is naively implemented, energy E_k is calculated anew at each algorithm step, and the same happens with $\Delta E_k = E_k - E_{k-1}$. A new state is the result of a random swap of indices i and j. This change does not affect every term of the estimated auto–covariance. The only values that change are those with indices i, j, $i-\tau$, and $j-\tau$. Instead of re–calculating all $n-\tau$ terms, just four of them have to be calculated anew to get the new energy value. Nonetheless, after some steps, the energy should be calculated anew, in order to avoid possibly occurring

[16] The states are assumed to be normally distributed.

[17] i.e. the number of accepted moves divided by the number of proposed moves.

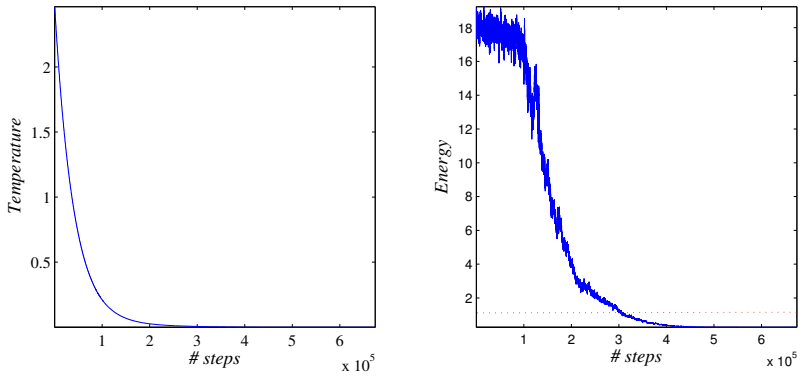

Figure 4.8: The left pane shows a typical temperature function. At the beginning of the procedure, the temperature falls away sharply, whereas it gets very flat in the end. The right pane displays the corresponding energy function. Both functions are computed by the cooling schedule described in section 4.2.3.

rounding errors. Besides, there are some special cases to pay attention to, e.g. $i, j \leq \tau$, or $|i - j| = \tau$.

Rejecting configurations. Another possibility to increase efficiency is to reject a new state in an early stage. A state with $\Delta E > 0$ is only accepted if

$$e^{-\frac{\Delta E}{T}} \geq r \tag{4.26}$$

holds. A possibility to reduce CPU time is the following: the random number $r \in [0, 1)$ has to be generated *before* the calculation of ΔE. $\Delta E = E_{new} - E_{old}$ and inequation 4.26 result[18] in

$$E_{new}^{q} \leq (E_{old} - Tln(r))^{q} \tag{4.27}$$

The left side of inequation 4.27 is positive. In order to reject a new state, both sides of the inequation have to be observed during the summation. If the inequation is not fulfilled, then the state is rejected.

[18] exponentiate $q \in \mathbb{N}$

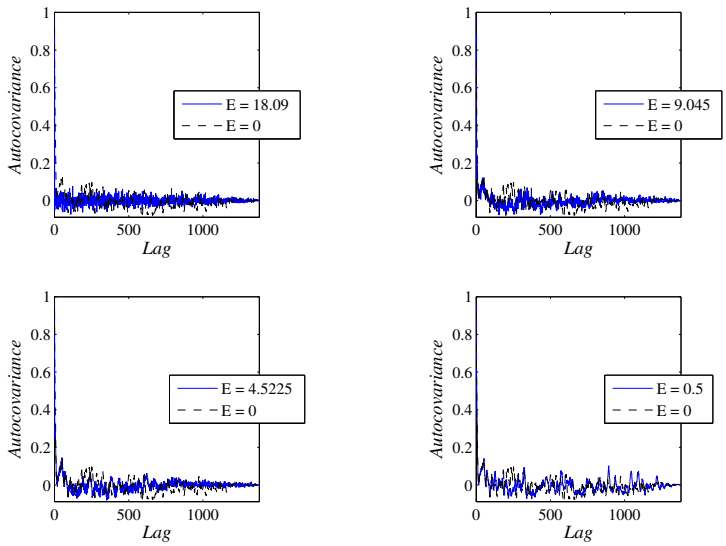

Figure 4.9: The panes display the normalized one–sided sample auto–covariances of the four surrogates in figure 4.7 as continuous lines, and the normalized one–sided sample auto–covariance of X as broken line. The smaller the energy is, the better the auto–covariances match. The smallest energy displayed is 0.5; there are still discrepancies visible.

Acceptance rate. An appropriate way of generating the next state improves efficiency as well. At the beginning of the annealing procedure, the swap of only two indices results in a very small energy decrease. On the other hand, the interchange of two indices may have a large effect on the energy in the end of the process. According to [1], the rate of acceptance is denoted A, and is the ratio of accepted updates to proposed updates. A decreases with the decrease of T. A constant acceptance rate $A \approx 50\%$ is optimal [8]. If the acceptance rate is too high, it can be lowered by interchanging more than two indices. In order to increase the acceptance rate in the end of the procedure, one has to differ between indices in close vicinity, and those further away from each other. The indices with the biggest difference are $i = 1$ and $j = n$. An interchange of i and j would result in a big change of E that is unlikely to be accepted. Hence, in order to increase A, two additional variables have to be introduced: First, one has to define the distance d of indices i and j with $i \neq j$, e.g. $d = |i - j| - 1$. d reaches a minimum, if x_i and x_j are neighbours in time. A probability function $p_{ij}(d)$ has to be introduced as well. It has to reach a maximum for $d = 0$, and decrease with an increasing d. In other words, it should be more likely that a pair $\{x_i, x_j\}$ with small difference d is chosen. The exact shape of p_{ij} is not important. Both an exponential and a

Gaussian probability function result in a decrease of computing time [1].

Testing. There is another, comparatively easy way to reduce the computing time. In order to approve a null hypothesis, the test statistic T_X has to be within a certain range of the surrogate test statistics. This range depends on the significance level α, and on the total number of surrogates N. N and α are known a priori, and therefore also the range. After creating a surrogate, T_X can be compared to the already created surrogate test statistics. If T_X is out of range, it is not necessary to create other surrogates any more, and the null hypothesis can be rejected.

4.2.5 Advantages

The simulated annealing method enables to avoid some problems of the Fourier–based methods. Simulated annealing implements a characteristic directly on a surrogate. Therefore, this method is much more flexible than the standard methods. Schmitz gives two examples of this flexibility in [1].

Periodicity. One problem of the Fourier standard methods is the implicit assumption that X is a period of an infinite periodical process. This may lead to artefacts in the surrogates in form of high frequencies, especially when there exists a jump discontinuity from x_n to x_1. This effect even occurs for $x_1 \approx x_n$, but an untypical peak occurs when the ends of X are attached to each other. All Fourier–based surrogates have about the same initial and final values. The simulated annealing method avoids these periodicity issues. γ_X can be directly implemented on a surrogate, and it is possible to construct surrogates with the same initial and final values as X. Fixing the endpoints means to implicitly introduce long–reaching correlations of X on a surrogate. Therefore, γ_X does not have to be reproduced for all lags, which lowers the computing effort. For a detailed description, the reader is referred to [1].

Precision. In general, it is possible to reproduce quantifiable characteristics with arbitrary precision on a surrogate. However, it may be unwanted to reproduce these characteristics *exactly* on a surrogate. Only some trivial transformations of X, like the time–inverted series, have the same γ as X, for example. The exclusion of these unwanted surrogates a priori is not possible with Fourier methods. With simulated annealing, it is. Energy function can be adapted to exclude these surrogates. The size of n and the shape of the energy function determine, if enough local optima exist in vicinity to $E = 0$.

4.2.6 Extension of H_0

Up to now, the most general null hypothesis is that X is a realization of a linear wide sense stationary process, measured by an instantaneous invertible

measure function. Any a priori–known quantifiable characteristics of X can be reproduced on a surrogates with simulated annealing. This enables the construction of surrogates also in cases, where the standard methods fail. Therefore, a broader range of null hypotheses can be tested. This helps to gain better insights into a system.

Non–Stationarity. In context of this thesis, the flexibility concerning non–stationarity is the biggest advantage of simulated annealing. The applied data indicate a lot of non–stationarities that can falsify test results. A null hypothesis may be rejected due to non–stationarity, and not due to nonlinearity, as intended. When X evidently contains non–stationarity, there are several ways to deal with it. In section 4.1.5, one possibility has been discussed. Another way to get rid of unwanted non–stationarities is the use of the pointwise difference between X and the non–stationarity as new data series. In order to do that, the non–stationarites have to be observed by additional parallel measurements, or modeled through fitting to X. Both options may be difficult to achieve. In some cases, not only the fact, that a time series is non–stationary is known, but also the type of non–stationarity. A simple example is yearly cycles. By implementing the non–stationarity into the cost function E, surrogates with the same non–stationarities as X can be created [1]. This aproach will be discussed in a later section.

Varying time intervals. So far, the focus has only been on evenly measured time series. In practice, it is often not possible to observe and measure data evenly. A common example are financial data, like stock quotations, which are not listed on weekends and holidays. Another example is given by Scargle in [21]. The author examines astronomical time series, which can only be measured if circumstances permit[19] [22]. Interpolating X on evenly spaced observation times is not advisable in case of testing for nonlinearity. Unwanted nonlinearities could be introduced, which cannot be distinguished from nonlinear structures in X. Filling the data gaps with zeros is only possible if the gaps appear regularly [1]. Frequency domain provides a possibility to create appropriate surrogates with the same linear properties as an unevenly measured X. Scargle and Lomb give a new periodogram definition, the so–called *LS–periodogram*. It is an alternative estimator for the power spectrum that is based on the least squares fitting of sine and cosine functions to the data. The LS–periodogram enables to estimate the spectrum of X with randomly, but chronologically fixed observation times. Vityazev compares the periodogram and the LS–periodogramm, and concludes, that, under certain conditions, they are identical. For further discussion, the reader is referred to his paper [23]. In general, there is no inversion of the LS–periodogram,

[19] e.g. night time, weather, availability of telescope time, the position of the object of observation, etc.

which makes the Fourier–based standard methods not practicable. The simulated annealing method does not require the invertibility. By implementing the LS–periodogram into the cost function E, the linear properties of X are reproduced on a surrogate. For application examples, the reader is referred to [22] and [21]. Schmitz and Schreiber implement the LS-periodogram into the cost function of a simulated annealing process in [24].
Another option is the implemenation of the linear properties of X on time domain. As for the periodogram, it is not possible to use the common estimator for the auto–covariance in case of unevenly sampled time series. Schmitz provides an extension of this estimator, which equals γ_X for evenly measured data. He divides the lags τ into intervalls of length δ_τ, and counts the pairs $\{x_i, x_j\}$, $i, j \in \{1, \ldots, n\}$, whose difference is within δ_τ. In this way, it is possible to estimate the auto–covariance with lag $\tau + \frac{\delta_\tau}{2}$ in the middle of the interval $[\tau, \tau + \delta_\tau]$. δ_τ has to be chosen properly, in order to make sure that the estimator exists [1]. In [25], this approach is used to create surrogates for a part of the Greenland ice core data set.
The two approaches give the possibility to investigate certain properties of unevenly sampled time series. Which of them should be used depends on the data.

Spike trains. A spike train is a sequence of events occurring at times $\{t_i\}$, e.g. heart beats. Variations in the events beyond their timing are ignored. This very common kind of data is fundamentally different from unevenly sampled time series. The sampling instances $\{t_i\}$ are not independent of the measured process. In fact, between these instances, the value of x_t is undefined, and the $\{t_i\}$ contain all the information. The discrete sequence of inter–event intervals $x_{t_i} = t_i - t_{i-1}$ is often treated as if it were an ordinary time series. The index i is not proportional to time any more. It depends on the nature of the process, if it is more reasonable to look for correlations in time, or in number of events. Because the applied data are evenly measured over time t, the cases of varying time intervals, or spike trains are not further discussed in this thesis. The interested reader is referred to [25].

Multivariate data. Another extension of null hypotheses that can be applied by simulated annealing will be discussed in the next chapter: X is a realization of a multivariate Gaussian linear process. Thus, dependencies *between* simultaneously measured data series can be tested.

4.3 Dealing with Nonstationarity

This section addresses the issue of nonstationarity in the data. Because the null hypotheses of the standard methods always imply the stationarity of X, any nonstationary behavior could lead to a false interpretation of the

test results. A rejection of H_0 can be caused by nonstationarity instead of nonlineariy. However, the stationarity restriction could be used to perform nonstationarity tests with surrogates. Several possibilities to reproduce certain nonstationarities on surrogates are discussed. Some of them are based on the already described methods, others are entirely new approaches.

4.3.1 Detecting Nonstationarities

Whereas the detection of nonlinearity allows to know, when linear analysis techniques are capturing all information in X, detecting nonstationarity allows to make informed decisions in data analysis, e.g. whether longer runs of data provide better estimates for a test statistic [60]. The first option of proceeding with nonstationarity is to simply regard nonstationarity as not suitable for time series analysis. The next step is to try to establish stationarity, i.e. to transform X in a way that it becomes stationary. The motivation for this is the assumption, that the underlying process may stay unchanged, but the measurement of the data does not reflect this fact [62]. A possibility to accomplish this is to use first differences

$$\Delta x_t = x_t - x_{t-1} \quad , \quad t = 2, \ldots, n \tag{4.28}$$

This eliminates nonstationarities of mean and variance.
Another possibility is to divide X into short stationary segments, and to analyze each of them separately. The motivation for this is that stationary analysis methods are feasible, while a change of the underlying process can still be tracked. Therefore, time–varying dynamics are seen as essential parts of the underlying process. Examples are processes with power–law[20]. This is called heteroscedasticity, and can be modeled by the ARCH model that was introduced before. These processes are considered as nonstationary, since more measurements do not provide better estimates. In testing for nonlinearity, *nonlinear* statistics are used, as well as the assumption that the data were generated by a stationary process. By using *linear* statistics, it is not possible to discriminate between X and a surrogate, as all linear test statistics on the surrogates have the same value as T_X. However, if X and the surrogates are divided into equally long segments, linear statistics differ from segment to segment in general. If X satisfies H_0, then the segment–by–segment values of T_X should be within the sampling distribution given by the segment–by–segment statistics T_{S_i}. The use of linear statistics limites the sensitivity to nonlinearity, and a difference of T_X and its estimated distribution is evidence for nonstationarity. Alternatively, by the use of nonlinear test statistics, nonlinearity on the segment–by–segment level can be tested by calculating surrogates for each segment individually. [60] The latter is applied, if there is a certain type of nonstationarity suspected. It is difficult

[20] i.e. time–dependent mean and variance.

to choose an appropriate segment length. If it is too short, the estimations of any quantifiers will be very poor. And if it is too long, the segments may be influenced by nonstationarities. Another approach is to use the null hypothesis, that X is a realization of a stationary process. However, such a test is not sufficient, because it simply might have no power against the type of nonstationarity. A test against *non*stationary would be desirable, but this H_0 is so composite, that there is no statistical test for it available [62].

In [59], Timmer makes use of the fact that it is difficult to distinguish between nonlinearity and nonstationarity. He investigates the power of data testing against two violations of stationarity. The use of cyclostationary processes[21] makes him conclude, that surrogate testing for linear Gaussian wide sense stationary stochastic processes is powerful against a violation of the assumption of stationarity. [59] Borgnat's approach is similar, but he utilizes an estimator for a time–varying spectrum, the multitaper spectogram. He shows, that for certain given nonstationary X, the multitaper spectrogram displays a clear organized structure and evolution along time. In contrast, a surrogate drawn from this signal reveals no specific structure in time, which is an evidence of stationarity. For further interest, the reader is referred to [61].

The following sections address the issue of creating surrogates with the same nonstationarities as X. Hence, the already discussed null hypotheses have to be extended with respect to the assumption of stationarity.

4.3.2 Fourier–based nonstationary Surrogates

The approach of Schmitz and Schreiber has already been mentioned before: they split the time series X into segments that can be considered nearly stationary, and generate a surrogate for each segment by means of the IAAFT algorithm. In order to receive a surrogate for X, the segment–surrogates are simply joined together. Clearly, this approach is not suitable for short X, or X with quickly varying nonstationarities, because the segments would have to be too short to work with. Moreover, all correlations between the segments get lost. The null hypothesis that corresponds to these surrogates is

> $\boldsymbol{H}_0$: X is realization of an ARMA process with slowly varying coefficients.

Faes et al. use a typical realization in [63]: in order to test the presence of nonlinear dynamics in potentially nonstationary signals, they are fitting

[21] i.e. linear processes with a periodic auto–covariance function in time

a time–varying (TV) autoregressive model to the original series:

$$x(t) = a(0,t) + \sum_{i=1}^{p} a(i,t)x(t-i) + \epsilon(t) \tag{4.29}$$

The model coefficients $a(\cdot, t)$ are regressed with random replacements of the model residuals to generate *TVAR surrogate series*. These are used in combination with a TV sample entropy discriminating statistic to assess nonlinearity in both simulated and experimental time series. For parameter optimization, the authors use a modified AIC criterion. For further description and test results, the reader is referred to [63].

4.3.3 Simulated Annealing and Nonstationarity

The flexibility of simulated annealing enables to include arbitrary characteristics in the cost function. If nonstationarity is known to be present, it has to be included in the null hypothesis explicitly. In general, this is difficult, but can be undertaken in some well behaved cases [25]. An easy example is a slow drift of mean and variance of the data. In [19], Schreiber gives the example of an AR(2) process with periodically modulated variance. He uses the annealing scheme to preserve the first 100 lags of $\gamma_X(\tau)$, but also the running variance in blocks of 200 and an overlap of 100 at $n = 2000$. The corresponding null hypothesis is

> $\boldsymbol{H}_0$: X is realization of a correlated linear stochastic process with time dependent local mean and variance.

Schmitz and Schreiber create surrogates, that satisfy this H_0. Their cost function was set up to match the auto–correlation function up to five days, and the moving mean and variance in sliding windows of 100 days. The authors also examine a time series with a single peak. They conclude that the spike results from some external process, and thus consider the time series nonstationary. A violation of the null hypothesis may be caused by that nonstationarity rather than nonlinearity. The simulated annealing method causes the spectral content of the single peak to be represented in the surrogates by a large number of shorter spikes [25].

4.3.4 Pseudoperiodic Surrogates

The method in this section was first presented by Small et al. in [64], and in [65]. It is based on the observation, that the standard surrogate techniques have very limited utility, when applied to a time series with a strong pseudoperiodic behavior that is not generated by any noise. The length of X is unlikely to be an integer multiple of the dominant period P, and the

jump discontinuity, as already described, may lead to additional fluctuations. In case of strong periodicities, the difference of the spectra of X and the surrogates is likely to influence the test result. Data that exhibit periodic structure are in most cases inconsistent with the hypothesis of a static monotonic nonlinear transformation of linearly filtered noise. The simulated annealing method, on the other hand, is able to reproduce the periodicities on the surrogates, but the computing effort is enormous. Another method suggested by Theiler is to shuffle the individual cycles, in order to destroy any structure with a period longer than the cycle length. However, this only works, if there exist convenient points at which to split the cycles [66].
The *pseudoperiodic surrogate* (PPS) algorithm offers an entirely new surrogate generation algorithm, which tests the null hypothesis

> $\boldsymbol{H}_0$: X is consistent with an (uncorrelated) noise–driven periodic orbit.

Assuming that X is nearly periodic with period P[22], the periodic orbit is the image of the periodicity interval under X in the state space[23]. In other words, the PPS preserve the periodic structure of the data, but are contaminated with dynamic noise in such a way that any existing additional structure, also linear and nonlinear determinism, is destroyed. Hence, these surrogates are not suitable for testing against linearity. Still, the basic ideas of the method are described in this section, because it is an interesting approach regarding nonstationarity.

At first, the underlying dynamic of the process is reconstructed by *time delay embedding*: An embedding vector in m dimensions has the following appearance:

$$\vec{x}_t = (x_{t-(m-1)\tau}, x_{t-(m-2)\tau}, \ldots, x_{t-\tau}, x_t) \tag{4.30}$$

where τ is the embedding lag, and m is called embedding dimension. A time series with length n results in $\tilde{n} = n - (m-1)\tau$ embedding vectors, which are indexed in accordance to the notation in [64] as $\{\vec{x}_t\}_{t=1}^{\tilde{n}}$. The idea is to get the underlying dynamic of a process from a local constant model over neighbors in state space, and to contaminate it with dynamic noise. The algorithm described by Small and Tse is displayed in algorithm 4.2. The authors use a parameter *noise radius* to define the type of noise applied. If this radius is chosen too small, the surrogate will be very similar, or even identical, to X. If radius is too large, then the surrogate will be effectively i.i.d. noise.
For a more detailed description of this method, and for examples of how to apply it, the reader is referred to [65].

[22] i.e. the minimum such $x_t = x_{t+P}$.
[23] i.e. the set of all possible states of a dynamical system.

Algorithm 4.2: Pseudoperiodic Surrogate

1: Random choice of initial condition $s_1 \in \{\vec{x}_t\}_{t=1}^{\tilde{n}}$
2: $i = 1$
3: **repeat**
4: Random choice of one neighbor of s_i from $\{\vec{x}_t\}_{t=1}^{\tilde{n}}$, say $\vec{x}_j$
5: $s_{i+1} \leftarrow \vec{x}_{j+1}$
6: $i \leftarrow i + 1$
7: **until** $i = n$
8: $PPS = \{(s_t)_1 : t = 1, \ldots, n\}$
▷ $(\cdot)_1$ is the scalar first coordinate of a vector
9: **return**

4.3.5 Wavelet–based Surrogates

In this section, a fairly new method of constructing surrogates for hypothesis testing of nonlinearity is presented. Whereas standard Fourier methods are based on the DFT of X, this method applies a *wavelet decomposition* of the time series. In context of surrogates, it was first used and discussed by Breakspear in [69], and later by Keylock, who presented a modified version in [67] and [68]. The following description is based on these papers.

Wavelet Decomposition. The main difference between Fourier transform and wavelet transform is that the former is only localized in frequency, whereas the latter is localized both in time and frequency. The idea of a wavelet representation is an orthogonal decomposition across a hierarchy of temporal and spatial scales by a set of *wavelet* and *scaling functions*. The (weighted) wavelet functions give the detail of the data at each scale, whereas the (weighted) scaling functions provide an approximation of the original time series at each scale, with the detail of smaller scales removed. These scaling functions form a nested sequence, as each successive level gives a better approximation of X. Instead of randomizing the Fourier phases as before, the wavelet coefficients are randomly permuted, in order to destroy any nonlinear information, while preserving other structures. A surrogate is constructed by combining each scaling level together with the detail at each level.

Formally, be $\mathbb{L}^2$ the space of square integrable functions. By scaling and shifting a so–called *mother wavelet function* $\psi \in \mathbb{L}^2(\mathbb{R})$, families of functions

$$\psi_{j,k}(x_t) = \frac{1}{\sqrt{2^j}}\psi(\frac{x_t - 2^j k}{2^j}) \quad , \quad j, k \in \mathbb{Z} \tag{4.31}$$

are generated. They form an orthonormal basis of $\mathbb{L}^2$. In equation 4.31, j is the scaling, and k is the shifting (translating) factor. For every mother

wavelet function, there is a unique *father wavelet function* $\phi \in \mathbb{L}^2(\mathbb{R})$[24]. $\phi_{j,k}$ are derived in a similar way as $\psi_{j,k}$. Effectively, ψ and ϕ act as band–pass filters. There are many pairs of mother and father wavelet functions in literature. The pair applied in this thesis is the *Daubechies 6 tap wavelet*, which is a pair of orthogonal wavelets defining a discrete wavelet transform. It is designed to have the highest number M of vanishing moments:

$$\int_{-\infty}^{\infty} y^l \psi(y) dy = 0 \quad , \quad l = 0, \ldots, M-1 \tag{4.32}$$

Precise appearance of ψ and ϕ goes beyond the scope of this thesis, the interested reader is referred to the works of Daubechies, especially to [70]. Numerically, wavelet functions can be derived with the Wavelet Toolbox$^{\mathrm{TM}}$ in MATLAB®.
From the wavelet functions, one defines linear subspaces of $\mathbb{L}^2(\mathbb{R})$

$$V_j = span(\phi_{j,k} : k \in \mathbb{Z}) \tag{4.33}$$
$$W_j = span(\psi_{j,k} : k \in \mathbb{Z}), \tag{4.34}$$

and requires the sequence

$$\{0\} \subset \cdots \subset V_1 \subset V_0 \subset V_{-1} \subset \cdots \mathbb{L}^2(\mathbb{R}) \tag{4.35}$$

to form a multiresolutional decomposition, i.e. to satisfy certain self–similarity, completeness, and regularity relations in time/space and scale/frequency. By construction, W_j is the orthogonal complement of V_j in V_{j-1}:

$$V_{j-1} = V_j \oplus W_j \tag{4.36}$$

Therefore, any function $f \in \mathbb{L}^2$ can be represented as

$$f = \sum_{k \in \mathbb{Z}} c_{J,k} \phi_{J,k} + \sum_{j \geq J} \sum_{k \in \mathbb{Z}} d_{j,k} \psi_{j,k} \tag{4.37}$$

The first sum represents the orthogonal subspace projector from $\mathbb{L}^2(\mathbb{R})$ onto the subspace V_j, and it describes the approximation of the function f at scale j. The second sum gives the orthogonal subspace projector from $\mathbb{L}^2(\mathbb{R})$ onto W_j, and stands for the details of f at scale j. The constants $c_{J,k}$ and $d_{j,k}$ are called approximation and detail coefficients. Breakspear points out, that these coefficients vanish outside an interval $k \in [k_1, k_2]$ for a finite time series X. The length of this interval at level j is determined by a constant *const*, that is derived by the wavelet function and n

$$N_j = 2^j n + const \tag{4.38}$$

[24] i.e. a scaling function in the time domain.

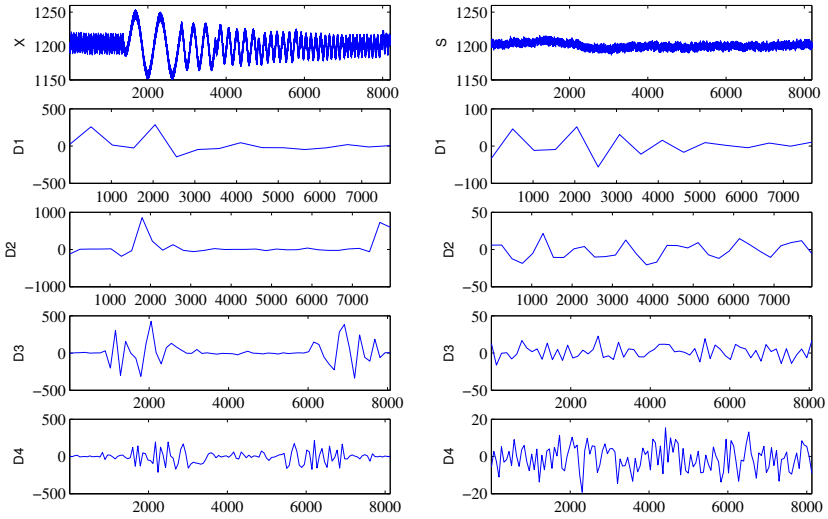

Figure 4.10: The top left pane displays a data series of manifold pressure with an indicated sine oscillation. The first four scales of the corresponding detail coefficients are depicted beneath. The right side displays a surrogate, that was created by random permutation through all scales, and the corresponding detail coefficients beneath.

Manipulating the Detail Coefficients. The main question that arises is to what extent does the wavelet decomposition include information about the linear properties of X. Breakspear and Keylock conclude, that a large part of nonlinear properties are described by the detail coefficients $d_{j,k}$. Though, a manipulation of these coefficients, in order to destroy nonlinearities in the surrogates, may also destroy linear correlation structures. The central assumption in constructing wavelet–based surrogates is, that the correlations between the detail coefficients are much weaker than in the data. The reason can be found in the vanishing moments, that the detail coefficients possess: if the mother wavelet is chosen to have a sufficient number of vanishing moments, the correlations between coefficients within and between levels decay rapidly. Hence, a manipulation of the linear correlation structure of $d_{j,k}$ has a small impact on γ_X [69]. Because the approximation coefficients $c_{j,k}$ do not have vanishing moments, their correlations cannot be assumed to be much weaker than those of the data. Hence, they are not manipulated in any way. These findings are adressed, among others, by Percival and Walden in [71].

There are several ways to manipulate detail coefficients. A resampling,

that involves permutations of detail coefficients *between* different scales, leads to a large discrepancy of the linear properties of X and a surrogate. Therefore, this is not a reasonable option [69]. Manipulations *within* the different scales should be utilized only. Breakspear describes three possible ways to manipulate the detail coefficients within each scale:

- Free permutation
- Cyclic rotations
- Block resampling

The first method is the simplest, it only uses a random shuffle of $d_{j,k}$. This can preserve the correlations of the original data only if the detail coefficients are either nearly uncorrelated

$$\mathbb{E}[d_{j,k}, d_{\tilde{j},\tilde{k}}] \approx 0 \quad , \quad \forall j, k, \tilde{j}, \tilde{k} \in \mathbb{Z}, \tag{4.39}$$

or there are enough vanishing moments in the mother wavelet function. However, there may be regions of frequency mismatch in the surrogates, as well as significant flattening of local peaks, because the shuffling also destroys correlations between different scales.
The second possibility is a cyclic rotation. The same random number is added to all indices within each level, and taken modulus N_j (see equation 4.38). This results in a spectral density function of the surrogates, that matches closer with the original spectrum than free permutation. However, the number of possible realizations is reduced from $\prod_{j=1}^{J} N_j!$ to $\prod_{j=1}^{J} N_j$.
The third technique divides the coefficients into blocks of length M_j at each level. These blocks are randomly permuted among themselves. All correlations within each block are preserved, which leads to a better fit of the spectra. The number of possible realizations is $\prod_{j=1}^{J} \lfloor \frac{N_j}{M_j} \rfloor !$.
These techniques are able to preserve certain behavior in the time–frequency plane. However, as they also produce, like the Fourier–based methods, surrogates with on average Gaussian amplitude distribution, they offer no clear advantage over the iterative AAFT algorithm presented earlier, at least for univariate time series. [67] Furthermore, the inherent periodicity of the wavelet coefficients at a each scale is not preserved by these coefficient manipulations [68].

WIAAFT and PWIAAFT. In order to overcome these problems, Keylock presents a different wavelet–based method in [67]. Again, X is transformed via a certain type of discrete wavelet transform, the *maximal overlap discrete wavelet transform* (MODWT). It has the advantage of being well defined for any length n, and is not restricted to a multiple of 2^J. Additionally, it produces coefficients and spectra, that are not affected by a shift of the data. The variance of the $d_{j,k}$ at each level is equivalent to the Fourier

spectrum of X. The important difference between this algorithm and the previous ones is the treatment of the detail coefficient sequences as time series themselves. The iterative AAFT algorithm is used to create a surrogate $d_{j,k}^{surr}$ for each of the J scales. This retains the frequency behavior of the coefficients. In order to locate peaks at the right locations in the signal, one has to transpose the $d_{j,k}^{surr}$, and fit both the surrogate and its transpose by circular rotation and a least–square criterion to the original coefficients. The series with the smallest error term is chosen. This helps to preserve the temporal structure. The next step is to perform the inverse MODWT with the detail coefficient surrogates, and the original approximation coefficients. An iterative procedure, that is identical to that of the IAAFT, has to be used to retain a *wavelet–based IAAFT (WIAAFT) surrogate* for X. Besides the fact, that these WIAAFT surrogates are optically very similar to X, they preserve the local mean and variance structure of X [67].

Sometimes it may be necessary to create surrogates, that leave certain striking data points in their exact position. In [68], Keylock presents a wavelet–based method, that allows to pin certain striking data points by means of a threshold criterion. Certain detail cofficients are fixed at their position within a scale, in order to construct *pinned WIAAFT (PWIAAFT) surrogates*. The algorithm is a slightly modification of the WIAAFT method: the shuffling within the IAAFT algorithm, that is applied on the detail coefficients, is not purely random any more. Some of the coefficients are simply excluded from the shuffling process and held in place. The iteration scheme in the IAAFT removes irregularities at higher frequencies. The rest of the procedure is in accordance to the WIAAFT technique. The question is, how many, and which, coefficients should be pinned. In order to answer this question, a threshold parameter t^* is introduced. The local energy of a coefficient at location i and scale j has to be defined as well. Only normalized detail coefficients[25] are used as in [68]. Because the power spectrum is proportional to the variance of the $d_{i,j}$, the local energy is described by the squared coefficients $d_{i,j}^2$, and the overall energy by

$$K = \sum_{i=1}^{n} \sum_{j=1}^{J} d_{i,j}^2. \tag{4.40}$$

The threshold is defined as the ratio of the energy of the unpinned points K^{up} to that of all points K

$$t^* = \frac{K^{up}}{K}. \tag{4.41}$$

Setting $t^* = 0$ would produce identical surrogates, because no energy comes from the unpinned coefficients. The WIAAFT algorithm is a special case

[25] with mean ~ 0

when $t^* = 1$, where no coefficients are held in place. Keylock concludes that a threshold of 0.1 is reasonable for data that consist of more than 512 observations. For a comparison of different thresholds for different data length, the reader is referred to [68].

4.4 Nonlinear Test Statistics

4.4.1 Higher and Cross Moments

The easiest and most natural nonlinear statistics are basically generalizations of γ_X. Higher and cross moments provide a class of discriminating statistics, in fact, many of them are the basis of traditional tests for nonlinearity in time series [9]. The justification for that comes from the fact, that higher moments than the second vanish for linear time series. Therefore, when T_X significantly differs from zero, nonlinearity could be a possible reason.

Three point autocovariance. The three point auto–covariance is a generalization of linear (two point) auto–covariance. A second lag is introduced:

$$\gamma^{3P}(\tau_1, \tau_2) = \frac{1}{n - \tau_2 - 1} \sum_{i=1}^{n-\tau_2} (x_i - \bar{x})(x_{i+\tau_1} - \bar{x})(x_{i+\tau_2} - \bar{x}) \tag{4.42}$$

Without loss of generality, let $\tau_1 < \tau_2$. Schreiber proposes to use $\tau_2 = 2 \cdot \tau_1$ in [36]. In order to normalize the test statistics, the *three point auto–correlation* may be estimated in the same manner:

$$\varrho^{3P}(\tau_1, \tau_2) = \frac{\gamma^{3P}(\tau_1, \tau_2)}{\gamma^{3P}(0, 0)} \tag{4.43}$$

Time reversal asymmetry. This test statistic is based on third moments as well. It is the estimator for time reversal asymmetry:

$$T^{rev}(\tau) = \frac{\sum_{i=1}^{n-\tau} (x_i - x_{i+\tau})^3}{\sum_{i=1}^{n-\tau} (x_i - x_{i+\tau})^2} \tag{4.44}$$

A time series is said to be reversible, if its probabilistic characteristics are invariant with respect to time reversal. Formally, the joint probability of $(x_i, x_{i+1}, \ldots, x_{i+\tau})$ equals the joint probability of $(x_{i+\tau}, x_{i+\tau-1}, \ldots, x_i)$ $\forall i, \tau$ [37]. This test statistic is both rapidly computable, and often quite powerful. Informally, it indicates the asymmetry between rise and fall times in the time series [9]. For linear processes, $T^{rev} = 0$ holds, whereas nonlinear processes often show a strong asymmetry with respect to time reversal. Hence, time irreversibility can be a strong signature of nonlinearity. For a detailed analysis and description of time reversal asymmetry, the reader is referred to [37].

4.4.2 Time Delay Embedding

The following nonlinear statistics are inspired by the theory of nonlinear dynamical systems, and they rely on a time delay embedding of the time series (see equation 4.30). Recall, if X is a scalar time series, an embedding vector in 2 dimensions consists of two elements, x_t and $x_{t-\tau}$. This is shown in figure 4.11 for a data series and its surrogate. The vectors are not identical but similar, which can be understood as a good reproduction of the characteristics of X on S.

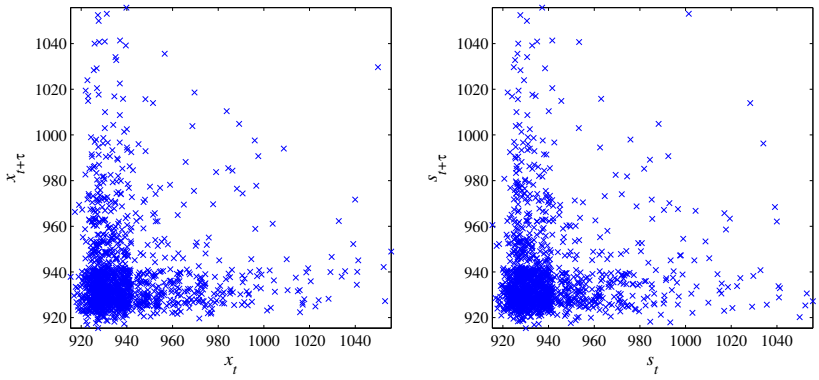

Figure 4.11: Embedding vectors in 2 dimensions with $\tau = 20$ for a data series X of manifold pressures, and a surrogate, which was created by the simulated annealing algorithm presented in equation 4.25.

Prediction error. Many quantities, that have been proposed in literature for nonlinearity testing, quantify the nonlinear predictability of a signal. A particularly stable representative of this class is the nonlinear prediction error with respect to a locally constant predictor F [36]

$$T^{pe}(m,\tau,r) = \left(\frac{1}{n-1} \sum_{i}^{n-1} [\vec{x}_{i+1} - F(\vec{x}_i)]^2 \right)^{\frac{1}{2}} \tag{4.45}$$

The local constant predictor F is an average over the future values of all neighboring delay vectors in m dimensions, that are closer than r:

$$F(\vec{x}_i) = \frac{1}{\|U_i(r)\|} \sum_{\vec{x}_j \in U_i(r)} x_{j+1} \tag{4.46}$$

where $\|U_i(r)\|$ is the number of points within the r–neighborhood of $\vec{x}_i$ [1].

Correlation dimension. The estimate of the correlation dimension has been widely used as a measure of the dimensionality of the underlying system. First, one has to define a correlation function, the *correlation sum*:

$$C_n(\epsilon) = \binom{n}{2}^{-1} \sum_{0 \le i < j \le n} I(\|\vec{x}_i - \vec{x}_j\| < \epsilon) \tag{4.47}$$

where $I(\cdot)$ equals 1 if the condition is fulfilled, and 0 otherwise. $\|\cdot\|$ is the usual distance function in the embedding room. The correlation dimension is defined as

$$d_c = \lim_{\epsilon \to 0} \lim_{n \to \infty} \frac{\log C_n(\epsilon)}{\log \epsilon} \tag{4.48}$$

The normalization of $C_n(\epsilon)$ is chosen so that rather than being an estimate of the average volume of an object within radius ϵ of a point, $C_n(\epsilon)$ is instead an estimate of the probability, that two points chosen at random are within a distance ϵ [38]. A widely used estimator for the correlation dimension is the *Grassberg–Procaccia correlation dimension*. Schreiber and Schmitz use two different algorithms for estimating the correlation dimension in [36], the BDS (Brock, Dechert, and Scheinkman) estimation, and the ML (maximum–likelihood) estimation. For further discussion, the reader is referred to Panagiotidis as well, who uses the BDS and several other tests in [39].

There is no general rule, which test statistic should be used. Schreiber and Schmitz conclude, that the root mean squared error of a simple nonlinear predictor gives consistently good discrimination power. They state, that other nonlinearity measures give even better performance in some cases, but fail in others. In particular, the time reversal asymmetry does very well most of the time, but can fail completely as well. The reason for this can be found in the fact that asymmetry under time reversal is a sufficient and powerful indicator of nonlinearity, but not a necessary condition. In the case where only a short time series is available, it seems advisable to use a robust, general purpose statistic with few adjustable parameters, for example a simple prediction error. If asymmetry under time reversal appears under visual inspection of the data, a simple statistic like T^{rev} will probably give best results [36]. For further information on nonlinear test statistics, the reader is referred to [36], and to [35].

Chapter 5

Nonlinear Dependencies between Time Series

In practice, several time series are often observed simultaneously in a system. In this thesis, two different physical measurements in a Diesel engine are applied. Hence, X is bivariate, and it consists of two channels

$$X_1 = (x_{1,1}, \ldots, x_{1,n}) \in \mathbb{R}^{(1\times n)} \tag{5.1}$$

$$X_2 = (x_{2,1}, \ldots, x_{2,n}) \in \mathbb{R}^{(1\times n)} \tag{5.2}$$

A generalization of bivariate methods can easily be found sometimes, but it is not part of this thesis. The analysis of nonlinear dependencies between different univariate time series is a broad and complex field. Therefore, its use has to be justified first. A common approach is to observe nonlinearities within the univariate time series respectively. By doing this, X_1 and X_2 are implicitely assumed to be independent. If there is evidence for nonlinearity, then the nonlinear dependencies between the series can be examined. With this in mind, the structure of this thesis was defined. The methods, that were discussed in the previous chapter, are able to justify methods in this chapter.

In order to construct surrogates with the same linear properties as a bivariate X, the reproduction of the linear characteristics of X_1 and X_2 respectively[1] is insufficient. The linear dependencies *between* them have to be reproduced as well. These dependencies are described by the *cross–covariance* which can be estimated through

$$\gamma_{X_1X_2}(\tau) = \frac{1}{n-\tau}\sum_{t=1}^{n-\tau}(x_{1,t} - \bar{x}_1)(x_{2,t+\tau} - \bar{x}_2) \quad , \quad \tau \geq 0 \tag{5.3}$$

Note, that $\gamma_{X_1X_2}(-\tau) = \gamma_{X_2X_1}(\tau)$. The cross–correlation is

[1] i.e. γ_{X_1} and γ_{X_2}.

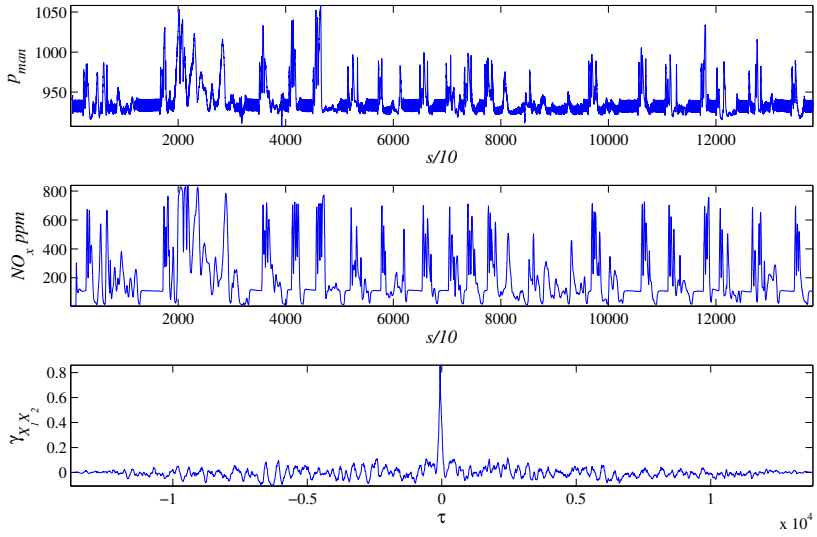

Figure 5.1: The upper pane displays a charging pressure data series, the next pane shows a simultaneously measured NO_X data series, and the lower pane shows the corresponding (normalized) empirical cross–covariance.

$$\varrho_{X_1X_2}(\tau) = \frac{1}{n-\tau}\sum_{t=1}^{n-\tau}\frac{(x_{1,t}-\bar{x}_1)(x_{2,t+\tau}-\bar{x}_2)}{\sqrt{Var_{X_1}\cdot Var_{X_2}}} \quad , \quad \tau \geq 0 \tag{5.4}$$

If X_1 and X_2 are linearly independent, then $\varrho_{X_1X_2} = 0$. In case of a perfect linear relation, $\varrho_{X_1X_2} = 1$ holds. Some algorithms, that are presented in the next section, are based on Fourier methods. These algorithms reproduce the *periodioc* cross–covariances on a surrogate. An estimator for the cross–covariance is

$$\gamma^{(P)}_{X_1X_2}(\tau) = \frac{1}{n-\tau}\sum_{t=1}^{n-\tau}(x_{1,t}-\bar{x}_1)(x_{2,mod(t+\tau-1,n)+1}-\bar{x}_2) \quad , \quad \tau \geq 0 \tag{5.5}$$

5.1 Fourier–based Algorithms

In the previous chapter, (inverse) discrete Fourier transform and phase randomization are used to create univariate surrogates. The amplitudes stay unchanged, because they contain the information on the linear characteristics of a data series. In the bivariate case, an extension of the Wiener–Khinchin theorem can be applied: the *cross–correlation theorem*. The basic statement of this theorem is that the Fourier transform of the (periodic) cross–correlation

equals the cross–spectrum:

$$\mathcal{F}_X = \mathcal{F}^*_{X_1}\mathcal{F}_{X_2} = A_1 A_2 e^{i(\phi_2-\phi_1)} \tag{5.6}$$

where $\mathcal{F}$ is the DFT, $\mathcal{F}^*$ its complex conjugate, and A_i and ϕ_i are the Fourier amplitudes and phases, $i \in \{1,2\}$. Note, that the cross–spectrum reflects only the phase difference $\phi_2 - \phi_1$. The normalized cross–spectrum is called *coherence*. The cross–correlation theorem enables to create bivariate surrogate data with the same linear characteristics as X. These characteristics are described by γ_{X_1}, γ_{X_2}, and $\gamma_{X_1X_2}$.

The simplest null hypothesis is

> $\boldsymbol{H}_0$: X_1 and X_2 are realizations of two independent linear Gaussian wide sense stationary processes.

Creating surrogates consistent with this H_0 has already been accomplished by simply creating univariate FT surrogates for X_1 and X_2, respectively. This preserves linear dependencies within X_1 and X_2, but does not preserves any (linear or nonlinear) interdependencies between them. Andrzejak et al. use the IAAFT method to construct surrogates, and they refer to them as type–I surrogates [44].

5.1.1 Unwindowed FT Algorithm

Theiler and Prichard describe an extension of the unwindowed FT algorithm in [43]. According to them, it is necessary to fix $\mathcal{F}^*_{X_i}\mathcal{F}_{X_j}$ for all pairs $\{i,j\}$, in order to preserve the auto–correlations and cross–correlations. This can be achieved by adding the same random sequence $\phi_r \in [0, 2\pi)$ to ϕ_1 and ϕ_2.[2] Formally, the bivariate surrogate S consists of

$$S_i = \mathcal{F}^{-1}_{X_i e^{i\phi_r}} \quad , \quad i = 1,2 \tag{5.7}$$

Theiler and prichard apply this modified FT algorithm both to a simulated example[3] and to multichannel EEG data. The reader is referred to the paper [43].

5.1.2 Windowed FT Algorithm

The technique of windowing works as before. In order to suppress the jump from $x_{1,1}$ to $x_{1,n}$, and that from $x_{2,1}$ to $x_{2,n}$, a weight–function is used. The procedure is described in the previous chapter. Note, that the weight–function has to be the same for X_1 and X_2. Otherwise, the cross–correlation function is not reproduced on a surrogate.

[2] There are only phase differences involved in equation (5.6).

[3] i.e. components of the Lorenz equations.

5.1.3 Iterative Algorithm

Schreiber and Schmitz discuss a combination of bivariate phase randomisation and an amplitude adjustment step in [25]. The extension of the iterative refinement scheme to the bivariate case is relatively straightforward. The authors point out, that deviations from a Gaussian distribution are very common, and may occur due to a simple invertible rescaling. This is supposed to simulated a measurement process. The IAAFT method consists of two procedures, which are applied in an alternating fashion, until convergence to a fixed point is achieved. The amplitude adjustment by rank ordering is readily applied to each channel individually. However, the spectral adjustment in the Fourier domain has to be modified. The change has to be applied to the filtering subprocedure. The randomized phases $\phi_{r1,k}$ and $\phi_{r1,k}$ have to be replaced by phases $\psi_{1,k}$ and $\psi_{2,k}$[4]. The replacement should be minimal in the least squares sense, i.e. $\psi_{1,k}$ and $\psi_{2,k}$ should minimize

$$h_k = \sum_{m=1}^{2} |e^{i\psi_{m,k}} - e^{i\phi_{rm,k}}|^2 \tag{5.8}$$

Also, the new phases must implement the same phase differences

$$e^{i(\psi_{1,k}-\psi_{2,k})} = e^{i(\phi_{1,k}-\phi_{2,k})}. \tag{5.9}$$

Equation 5.9 holds if

$$\psi_{1,k} - \psi_{2,k} = \phi_{1,k} - \phi_{2,k} = \alpha_k \tag{5.10}$$

Under this additional constraint, the minimization problem of h_k becomes one–dimensional. An appropriate value for α_k is found by eliminating $\psi_{1,k}$ and $\psi_{2,k}$ from equation 5.8:

$$h_k = 4 - 2\cos(\alpha_k - \phi_{r1,k} + \phi_{1,k}) - 2\cos(\alpha_k - \phi_{r2,k} + \phi_{2,k}) \tag{5.11}$$

This is extremal for

$$\tan\alpha_k = \frac{\sin(\phi_{r1,k} - \phi_{1,k}) + \sin(\phi_{r2,k} - \phi_{2,k})}{\cos(\phi_{r1,k} - \phi_{1,k}) + \cos(\phi_{r2,k} - \phi_{2,k})}, \tag{5.12}$$

and minimal if α_k is chosen in the correct quadrant. For negative k, the values have to be adjusted as in the previous chapter, in order to make sure that the surrogate is real–valued [44]. Andrzejak et al. refer to surrogates, that are designed to test the null hypothesis

> $\boldsymbol{H}_0$: X is realization of a bivariate linear stochastic processes, with an arbitrary degree of linear cross–correlation.

as type–II surrogates. In [25], the design of type–II surrogates for a simultaneous recording of breath rate and instantaneous heart rate of a human during sleep is described.

[4] $k = -\lfloor(\frac{n-1}{2})\rfloor, \ldots, \lfloor(\frac{n}{2})\rfloor$.

5.1.4 Constraints

When creating surrogates with Fourier–based algorithms, only the linear characteristics within and between X_1 and X_2 are reproduced. All nonlinear dependencies vanish during the procedure. In many cases, this is unwanted. If there are found nonlinearities within X_1 and X_2 respectively, the question may arise whether the nonlinearities in one of the series depend on the nonlinearities in the other. Schmitz gives the example of breath and heart rate [1]. A possibility is to fix on of the time series, and to create surrogates of the other series with the same auto–covariances and cross–covariances. This is not possible with the standard Fourier–based methods.

5.2 Applied Simulated Annealing

5.2.1 Cost Function

Standard Fourier–based methods are not able to satisfy more general null hypotheses like

> $\boldsymbol{H}_0$: X_1 and X_2 are realizations of processes with arbitrary structure, and an arbitrary degree of linear cross–correlation, but without nonlinear interdependence.

Andrzejak et al., for example, assume two independent nonlinear deterministic dynamics, which have been measured by some kind of a linear superposition. It is not possible to distinguish a linear superposition of independent nonlinear deterministic dynamics from interdependent nonlinear deterministic dynamics by the previous null hypotheses. In some cases, it might be sufficient to preserve only X_1 and $\gamma_{X_1X_2}$, and to ramdomize X_2. However, at least the linear characteristics of X_2, γ_{X_2}, should be preserved. The constraints to simultaneously preserve X_1, $\gamma_{X_1X_2}$, and γ_{X_2} would still overspecify the problem. The only surrogate possible would be an exact copy of X. As a way out of this dilemma, Schreiber proposes to preserve $\gamma_{X_1X_2}$ and γ_{X_2} only up to a maximum lag τ_{max}. This can be achieved by minimizing an energy function

$$E = \omega_1 \sum_{\tau=1}^{\tau_{max}} (\gamma_{X_2} - \gamma_{S_2}) + \omega_2 \sum_{\tau=-\tau_{max}}^{\tau_{max}} (\gamma_{X_1X_2} - \gamma_{X_1S_2}) \tag{5.13}$$

where $\omega_1 > 0$ and $\omega_2 > 0$ are weights. Only positive τ have to be summed up, because the empirical auto–covariance is symmetric. In contrary, the cross–covariances require all τ in the sum.

5.2.2 Additional Remarks

The simulated annealing procedure works exactly the same as before. The additional parameters ω_1 and ω_2 have to be adjusted propperly by back–

testing several configurations. The parameters depend on X, hence there is no general tuning rule for them.

Regarding efficiency, this algorithm should not be naively implemented. Similar to the univariate case, there are several possibilities to reduce computing time, for example by calculating the by change affected elements of the sums only. Andrzejak et al. describe disadvantages of the bivariate simulated annealing in [44]. First, they mention a whitening of the surrogates power spectrum, which is caused by a mismatch of the auto–correlations. Even for extremely low energy values, a persistent noise cannot be suppressed. The second problem arises for time series X_1 and X_2 with very strong cross–correlation. The surrogates are constructed to preserve the cross–correlation of X. If one channel X_1 is fixed, the surrogates tend to converge towards a copy of the original time series. This is hardly relevant for long time series.

5.3 Wavelet–based Surrogates

Similar to the Fourier methods, the univariate wavelet methods can be extended as well. A possible approach is provided by Breakspear, and is described in [69]. In order to test the null hypothesis of no (linear or nonlinear) correlation between two time series X_1 and X_2, two univariate wavelet surrogates are created, with different permutations of the detail coefficients $d_{j,k}$. This preserves only the linear structure within each channel.

The null hypothesis of linear dependency between X_1 and X_2 can be tested by a procedure analog to the synchronical Fourier phase randomization. The same manipulations of the detail coefficients on the same scales are implemented. Breakspear concludes, after comparing the empirical coherence of X and the surrogates, that the surrogate coherence matches the overall trend, but does not preserve peaks and troughs, that are present in the coherence of the original data. A cyclic rotation of the coefficients enables a better preservation across all frequencies. The best match with the original coherence can be achieved by block sampling, which is capable of preserving even fine patterns.

If dependencies between different scales are supposed to be tested as well, the coherence is not a suitable tool. It is sensitive to the same frequencies in the channels only. A possibility is to observe correlations between detail coefficients at different scales.

5.4 Nonlinear Test Statistics

The way in which surrogates are designed, highly influences the class of possible test null hypotheses, and also the class of nonlinear test statistics. Concerning a test statistic T_X, the questions, that may arise, are

- Is the focus on just detecting nonlinearity?
- What is the direction of the nonlinearity?
- Is one part of the system the driver of nonlinearity? Or is it interactive?
- How strong is the nonlinear dependency?

According to the answers, the test statistic is chosen. The purpose of this section is to introduce several possible test statistics, and to outline their advantages and weaknesses respectively. The guideline for this section can be found in [1] and [44].

5.4.1 General Properties

The multitude of possible dependencies between different channels in a system leads to an immense number of descriptive quantifiers. The fact, that only bivariate systems $X = \{X_1, X_2\}$ are considered in this thesis, does not make the task less multi–faceted. For illustration, consider a physical (sub)system consisting of X_1 and X_2. Depending on the purpose of testing, surrogates are created, and several types of dependencies between the two channels are explored by surrogate testing. But what, if any found nonlinearity depends on a third unknown channel X_3? Are the results still valid? And what if X_3 is not only unknown, but also even not observable? Although this is a simple example, it shows the difficulty not only of testing, but even more of interpreting the results appropriately. The better the properties of the used test are understood, the better they can be interpreted afterwards. Therefore, it makes sense to think about classes and basic features of test statistics, before applying them in practice. In this section, the notation $T_{X_1X_2}$ is used for a general test statistic, in order to indicate that $T_{X_1X_2}$ describes the dependency between X_1 and X_2.

Symmetric vs. Asymmetric. Symmetry is concerned with the interchangeability of X_1 and X_2. Formally, an observable $T_{X_1X_2}$ is called symmetric if

$$T_{X_1X_2} = T_{X_2X_1} \tag{5.14}$$

Any symmetric $T_{X_1X_2}$ does not tell anything about the direction of the dependency. Also, there are trivial asymmetries that do not give any information on the direction of the coupling, e.g. $T_{X_1X_2} = -T_{X_2X_1}$. This is a basic

property of the cross–covariance $\gamma_{X_1X_2}$ (see equation 5.3)

$$\gamma_{X_1X_2}(-\tau) = \gamma_{X_2X_1}(\tau) \tag{5.15}$$

Hence, the cross–covariance with lag $\tau = 0$ is symmetric, and does not provide any information about the direction of dependency. In other words, it is not possible to determine any instantaneous driver–response relationship between X_1 and X_2 with $\gamma_{X_1X_2}$.

Static vs. Dynamic. Another way to distinguish classes of test statistics, is to divide them into static and dynamic. $T_{X_1X_2}$ is static, if it is neither based on the dynamics of X_1 and X_2 nor on the coupling between them. Instead, only embedding vectors and their distribution are utilized. Remember, the embedding vectors of X_1 and X_2 have the representation

$$\vec{x}_{k,i} = (x_{k,i-(m-1)\tau}, x_{k,i-(m-2)\tau}, \ldots, x_{k,i-\tau}, x_{k,i}) \tag{5.16}$$

with $k = 1, 2$, $i = 1, \ldots, n$, and $m \in \mathbb{N}$ is the embedding dimension. Whereas static $T_{X_1X_2}$ enable to ascertain strength and direction of the coupling, dynamic $T_{X_1X_2}$ may even give the possibility to determine any dynamic in it.

5.4.2 Nonlinear Extension of $\gamma_{X_1X_2}$

In the univariate case, three test statistics have been presented, which are all nonlinear extensions of the auto–covariance function. The same strategy can be pursued in the multivariate case. Schmitz proposes $T_{X_1X_2}$ to have the following representation:

$$\gamma^{(rs)}_{X_1X_2}(\tau) = \frac{1}{n-\tau}\sum_{t=1}^{n-\tau}(x_{1,t} - \bar{x}_1)^r(x_{2,t+\tau} - \bar{x}_2)^s \quad , \quad r, s \in \mathbb{N}. \tag{5.17}$$

$\gamma^{(rs)}_{X_1X_2}$ allows to detect any arbitrary dependencies between X_1 and X_2. The channels are statistically independent, only if $\gamma^{(rs)}_{X_1X_2}(\tau) = 0 \ \forall r, s, \tau$ holds. The broad parameter space makes $\gamma^{(rs)}_{X_1X_2}$ computing intensive and unhandy. There are several better suited alternatives of nonlinear dependency measures.

5.4.3 Cross–correlation Sum

As mentioned in section 4.4.2, the correlation dimension is a widely used tool for better understanding of an underlying system. In this section, a slightly different notation (see [1]) will be used, in order to find the multivariate extension for the correlation sum, the cross–correlation sum. Recapitulate,

the correlation dimension is an estimate of the probability, that $\vec{x}_i$ and $\vec{x}_j$ chosen at random are within a distance ϵ of each other. This is represented by the correlation sum

$$C(\epsilon) = \frac{2}{(n - t_{min})(n - t_{min} - 1)} \sum_{i=1}^{n} \sum_{j=i+t_{min}}^{n} \Theta(\epsilon - \|\vec{x}_i - \vec{x}_j\|) \tag{5.18}$$

where $\Theta(\cdot)$ is the Heaviside–function[5]. By choosing $t_{min} > 1$, neighboring points are excluded from summation. The correlation dimension is estimated as in equation 4.48

$$d_c = \lim_{\epsilon \to 0} \frac{\log C(\epsilon)}{\log \epsilon} \tag{5.19}$$

In the univariate case, $\vec{x}_i$ and $\vec{x}_j$ have to be construed by embedding the scalar x_i in m dimensions with a certain lag τ. In the multivariate case, the $\vec{x}_i$ and $\vec{x}_j$ could also be the multivariate data points of the time series. The extension of the correlation sum is the cross–correlation sum

$$C_{X_1X_2}(\epsilon) = \frac{1}{n^2} \sum_{i,j=1}^{n} \Theta(\epsilon - \|\vec{x}_{1,i} - \vec{x}_{2,j}\|). \tag{5.20}$$

$C_{X_1X_2}$ is symmetric, and when it is identical with the individual correlation sums $C_{X_1X_1}$ and $C_{X_2X_2}$, identical distributions can be assumed. It can be seen as measure of similarity of $\vec{x}_{1,i}$ and $\vec{x}_{2,j}$. The main critique of this test statistic is that it can hardly be interpreted properly, if X_1 and X_2 are measured in different systems. Prichard and Theiler also use the cross–correlation and an estimator of correlation dimension as test statistic in [43].

5.4.4 Information Theoretic Measures

The discipline of information theory was crucially affected by the work of Shannon about communication theory [45]. The author examines the elements of communication systems and their interaction. According to his findings, a system consists of an information source, a transmitter, a channel, a receiver, and a destination. Furthermore, he classifies communication systems into three main categories: discrete, continuous and mixed. A discrete system is one, in which both the message and the signal are a sequence of discrete symbols. An example is telegraphy, where the message is a sequence of letters, and the signal a sequence of dots, dashes, and spaces. A sequence of choices from a finite set of elementary symbols, i.e. the 'letters' $L = \{L_1, L_2, \ldots, L_B\}$, can be transmitted from one point to another. Each of the symbols L_i is assumed to have a certain probability of occurrence $p(i)$ with $\sum_i p(i) = 1$,$i = 1, \ldots, B$. The question proposed by Shannon is whether it is possible to find a measure of uncertainty of the outcome. He states, that

[5] $\Theta(v > 0) = 1$, and $\Theta(v \leq 0) = 0$

if there is such a measure $H(p(1), p(2), \ldots p(B))$, it is reasonable to require the following properties:

- H should be continuous in the $p(i)$.
- If all $p(i)$ are equal, then H should be a monotonic increasing function of B.[6]
- If a choice is broken down into two successive choices, the original H should be the weighted sum of the individual values of H.

H fulfilling these conditions is

$$H(X_1, L) = -K \sum_i p(i) \log p(i) \tag{5.21}$$

where K is a positive constant, and w.l.o.g. $K = 1$. In literature, equation 5.21 is referred as *Shannon–entropy*. It is used in information theory as measure of information, choice, and uncertainty. In this thesis, it is used as quantity, that measures the information, that can be gained by observing X_1 [1]. In the scalar case, Schmitz uses an intervall in $\mathbb{R}$ for L, or a subset of $\mathbb{R}^m$ for m–dimensional embeddings. By partitioning L into $L_1, \ldots, L_B$, the definition of $p(i)$ is the probability, that $x_{1,t}$, or $\vec{x}_{1,t}$, is within L_i. This probability can be estimated by dividing the number of observed points within the partition by the number of data n, or $n - (m-1)\tau$ (when using embedding vectors). In the latter case, the partitions are m–dimensional disjoint boxes which overlay L. The box–size δ is a free parameter with

$$d_\delta = \lim_{\delta \to 0} \frac{-H(X_1, \delta)}{\log \delta}, \tag{5.22}$$

that is called *information dimension*.

The extension to the bivariate case follows the same principle. Two finite sets of elementary symbols L^{X_1} and L^{X_2} have to be introduced. Let $p(i,j)$ be the probability, that $x_{1,t}$, or $\vec{x}_{1,t}$, is within $L_i^{X_1}$, and $x_{2,t}$, or $\vec{x}_{2,t}$, is within $L_j^{X_2}$. The *entropy of the joint events* is then

$$H(X_1, X_2) = -K \sum_{i,j} p(i,j) \log p(i,j) \tag{5.23}$$

with

$$H(X_1) = -K \sum_{i,j} p(i,j) \log \sum_j p(i,j) \tag{5.24}$$

$$H(X_2) = -K \sum_{i,j} p(i,j) \log \sum_i p(i,j) \tag{5.25}$$

[6] When events are equally likely, there is more uncertainty, as the number of events increases.

It can easily be shown that

$$p(i,j) \leq p(i) \cdot p(j) \tag{5.26}$$

and hence

$$H(X_1, X_2) \leq H(X_1) + H(X_2) \tag{5.27}$$

Thus, the uncertainty of a joint event is less than, or equal to, the sum of the individual uncertainties. The equality is only given if X_1 and X_2 are independent.

Schmitz uses these findings to design a *test statistic of mutual information*

$$M(X_1, X_2) = H(X_1) + H(X_2) - H(X_1, X_2). \tag{5.28}$$

When X_1 and X_2 are independent, then $M(X_1, X_2) = 0$ holds. In case of total dependency of X_2, $M(X_1, X_2) = H(X_2)$ holds, and in case of identical channels $X_1 = X_2$, then $M(X_1, X_2) = H(X_1)$ holds. This test statistic is symmetric, and is not suitable to give any information on the direction of the dependency. For a multivariate extension the reader is referred to [1].

Kullback and Leibler introduce a related, but different entropy in [46]. Their motivation is the statistical problem of discrimination, and they introduce a measure of 'divergence' between statistical populations in terms of their measure of information. They define

$$\log \frac{p_1(X_1)}{p_2(X_1)} \tag{5.29}$$

as the information in X_1 for discrimination between the hypotheses, that X_1 realizes a probability measure p_i, $i = 1, 2$. By using Equation 5.29, the information between X_1 and X_2 can be measured by the *Kullback–Leibler–entropy*

$$M(X_1, X_2) = \sum_{i,j} p(i,j) \log \frac{p(i,j)}{p(i)p(j)} \tag{5.30}$$

$M(X_1, X_2)$ can be interpreted as surplus information resulting from using the probability distribution $p(i)p(j)$ instead of $p(i,j)$. This corresponds to the assumption of inderpendence of X_1 and X_2. Only if this assumption is true, $M(X_1, X_2) = 0$ holds.

Schmitz gives another possibility to compute entropies. The basic idea is to use neighborhoods of the embedding vectors $\vec{x}_{1,t}$ and $\vec{x}_{2,t}$ instead of boxes, and to use a relationship of correlation sum and entropy. For a detailed description the reader is referred to [1].

The main disadvantage of the presented test statistics is their symmetry. Schreiber introduces several asymmetric statistics in [47], which are presented in the following sections.

5.4.5 Time–delayed Mutual Information

The mutual information measures, that are presented in the previous sections, contain neither dynamical nor directional information. One possibility for improvement is introducing a time delay τ in one of the observations. Applying this idea on the Kullback–Leibler–entropy, for example, yields

$$M_{X_1X_2}(\tau) = \sum_{i,j} p(i_t, j_{t+\tau}) \log \frac{p(i_t, j_{t+\tau})}{p(i_t)p(j_{t+\tau})} \tag{5.31}$$

The asymmetry in $M_{X_1X_2}$ originates from the time delay τ only.

$$M_{X_1X_2}(-\tau) = M_{X_2X_1}(\tau) \tag{5.32}$$

In some cases, the time–delayed entropy is able to identify non–instantaneous couplings, if the delay is smaller than the time that coupling needs. Also, instantaneous couplings may be identified, if the information does not get lost in time dynamics [1]. Still, this method does not explicitly distinguish information, that is actually exchanged, from inrformation, that occurs due to the response to a common input signal or history [47].

5.4.6 Conditional Entropy

A different approach is to use conditional information. The proceeding is the same as for the already discussed entropies, except for the probability distributions. Instead of the mutual probability $p(i, j)$, the conditional probability

$$p(i|j) = \frac{p(i, j)}{p(j)} \tag{5.33}$$

is applied. It is the probability, that X_1 is observed in state i when X_2 is in state j. Starting from the Shennon–entropy, the conditional entropy for the state j is given by

$$H_j(X_1|X_2) = -K \sum_i p(i|j) \log p(i|j) \tag{5.34}$$

and by summation over j with respect to $p(j)$

$$H(X_1|X_2) = -K \sum_{i,j} p(i, j) \log p(i|j). \tag{5.35}$$

The relation between Shannon and conditional entropy is given by

$$H(X_1|X_2) = H(X_1, X_2) - H(X_2) \tag{5.36}$$

Substitution in equation 5.28 gives

$$H(X_1|X_2) = H(X_1) - M(X_1, X_2) \tag{5.37}$$

The asymmetry of the constrained entropy is only based on the differences of the individual entropies $H(X_1)$ and $H(X_2)$

$$H(X_1|X_2) - H(X_2|X_1) = H(X_1) - H(X_2) \tag{5.38}$$

and not on information flow. [47]

5.4.7 Transfer Entropy

The concepts of time–delayed and conditional entropy lead to the so–called transfer entropy. This denomination arises from the method of studying dynamical structures by transition probabilities, rather than static probabilities. The basic assumption is that the system X is approximated by a stationary Markov process of order k

$$p(i_{t+1}|i_t, \ldots, i_{t-k+1}) = p(i_{t+1}|i_t, \ldots, i_{t-k+1}, i_{t-k}) \tag{5.39}$$

Equation 5.39 states that the conditional probability to find X_1 in state i_{t+1} at time $t+1$ is independent of state i_{t-k}. In line with [47], the notation $i_t^{(k)} = (i_t, \ldots, i_{t-k+1})$ is used from now on. The average information needed to describe one additional state of the system, if all previous states are known, is given by the *entropy rate*

$$G_{X_1}^{(k)} = -K \sum p(i_{t+1}, i_t^{(k)}) \log p(i_{t+1}|i_t^{(k)}) \tag{5.40}$$

$G_{X_1}^{(k)}$ equals the difference between the Shannon–entropies of the processes, given by $k+1$ and k dimensional delay vectors constructed from X_1. Schmitz and Schreiber give a more detailed derivation for these findings in [47] and [1]. A way to extend the univariate $G_{X_1}^{(k)}$ to a bivariate $G_{X_1X_2}^{(k,l)}$ is to adapt equation 5.39 to

$$p(i_{t+1}|i_t^{(k)}) = p(i_{t+1}|i_t^{(k)}, j_t^{(l)}) \tag{5.41}$$

If equation 5.41 holds, then the state of X_2 has no influence on the transition probability on X_1. By utilizing this matter of fact, the *transfer entropy* can be introduced

$$H(X_2 \to X_1) = \sum p(i_{t+1}, i_t^{(k)}, j_t^{(l)}) \log \frac{p(i_{t+1}|i_t^{(k)}, j_t^{(l)})}{p(i_{t+1}|i_t^{(k)})} \tag{5.42}$$

By construction, $H(X_2 \to X_1)$ is asymmetric, and therefore well–suited for the purpose of identifying the direction of couplings between X_1 and X_2. For a further interpretation of the transfer entropy, and an accurate choice of k and l, the reader is again referred to [47] and [1].

5.4.8 Interdependencies

A main concept, that is used to measure interdependencies, is that of *synchronization*. This section is supposed to summarize the findings of [1] and [48]. The first, and most demonstrative observation of synchronization was that of Huygens in 'Horologium Oscillatorium' in 1673. He describes two pendulum clocks arranged on the same wall, which after a certain time swing regularly, i.e. isochronously, irrespective of their amplitudes. The wall conducted a coupling of the clocks. Schmitz describes the different types of synchronization, that have been examined so far, in [1]:

- Identical synchronization
- Generalized synchronization
- Strong synchronization
- Weak synchronization

A system X consisting of two coupled channels X_1 and X_2 with

$$\vec{x}_{1,t+1} = F(\vec{x}_{1,t}) \tag{5.43}$$
$$\vec{x}_{2,t+1} = G(\vec{x}_{2,t}, \vec{x}_{1,t}) \tag{5.44}$$

is a drive–response system with X_1 as driver and X_2 as responder.

An *identical synchronization* is given, if the relation

$$F(\cdot) = G(\cdot, 0) \tag{5.45}$$

holds, and the systems move, independent of initial values, synchronically, i.e. $\vec{x}_{2,t} = \vec{x}_{1,t}$.

The *generalized synchronization* extends the condition of equality of $\vec{x}_{1,t}$ and $\vec{x}_{2,t}$ to $\vec{x}_{2,t} = \psi(\vec{x}_{1,t})$. Still, the response system X_2 is only dependent on the drive–system X_1. When the transfer–function $\psi(\cdot)$ is a smooth function[7], the synchronization is considered to be *strong*. Depending on the context, 'smooth' can also be referred to infinitely continuous differentiable. Synchronizations of systems without this property are called *weak*.

The concept of synchronization does not provide the possibility to identify directions of a coupling, for example when $\psi(\cdot)$ is bijective. Also, generalized synchronizations are difficult to identify in general. When the underlying process of X_1 and X_2 is known, the latter issue may be solved. Schmitz describes two methods to do so: the first is based on estimating the Lyapunov–coefficient of the overall system X. Sufficient, and necessary condition to detect generalized synchronization is the negativity of all Lyapunov–coefficients

[7] i.e. it is continuously differentiable

of the response system X_2. The second approach considers two response systems X_{2a} and X_{2b}, which are used to compare the synchronizations. The main disadvantage of these methods is that the underlying process has to be known. As this is often not the case, the process has to be estimated. The estimation has to be very well–fitting, because synchronizations may be weak and difficult to identify. Also, any noise could falsify the results. Therefore, this method is not suitable for most time series that are observed only once. Especially in biology, and the research of procedures in the human body, the methods of detecting synchronization are promising. An example are brainwaves during epileptic seizures. Identification of synchronization in between the different channels of brainwaves could give important insights.

Even though it is hardly ever possible to identify synchronizations precisely, it is desirable to know, which part of the system is the driver, and which the responder[8]. Arnhold et al. propose a measure, that is asymmetric, and provides information about the direction of interdependence. The authors design the measure to be robust against noise, and to be able to detect weak couplings as well. The following methods are explained in detail in [48]:

Let $X_1 = (x_{1,1}, \ldots, x_{1,n})$ and $X_2 = (x_{2,1}, \ldots, x_{2,n})$ denote two different simultaneously observed time sequences. The internal dynamics of the system is not known. In particular, it is not known, whether the system is deterministic, or stochastic. The embedding vectors $\vec{x}_{1,i}$ and $\vec{x}_{2,i}$ are defined as before. The arrays of all delay vectors will be denoted $\vec{X}_1 = (\vec{x}_{1,1}, \ldots, \vec{x}_{1,n})$ and $\vec{X}_2 = (\vec{x}_{2,1}, \ldots, \vec{x}_{2,n})$. Let $r_{i,j}$ and $s_{i,j}$, $j = 1 \ldots, k$ denote the time indices of the k nearest neighbors of $\vec{x}_{1,i}$ and $\vec{x}_{2,i}$respectively. For each $\vec{x}_{1,i}$, the squared mean Euclidean distance to its k closest neighbors is defined as

$$R_i^{(k)}(\vec{X}_1) = \frac{1}{k} \sum_{j=1}^{k} (\vec{x}_{1,i} - \vec{x}_{1,r_{i,j}})^2 \tag{5.46}$$

while the conditional mean squared Euclidean distance, conditioned on the closest neighbor times in $\vec{X}_2$, is

$$R_i^{(k)}(\vec{X}_1|\vec{X}_2) = \frac{1}{k} \sum_{j=1}^{k} (\vec{x}_{1,i} - \vec{x}_{1,s_{i,j}})^2 \tag{5.47}$$

$R_i^{(k)}(\vec{X}_2)$ and $R_i^{(k)}(\vec{X}_2|\vec{X}_1)$ are defined in complete analogy. If the dynamic of $\vec{X}_2$ is independent of $\vec{X}_1$, then there is no particular relation between $r_{i,j}$ and $s_{i,j}$, and

$$R_i^{(k)}(\vec{X}_1|\vec{X}_2) \gg R_i^{(k)}(\vec{X}_1) \tag{5.48}$$

[8] in case such a relationship exists

holds. Accordingly, Arnhold introduces the interdependence measure

$$S^{(k)}(\vec{X}_1|\vec{X}_2) = \frac{1}{n}\sum_{i=1}^{n}\frac{R_i^{(k)}(\vec{X}_1)}{R_i^{(k)}(\vec{X}_1|\vec{X}_2)} \tag{5.49}$$

which by construction is $0 < S^{(k)}(\vec{X}_1|\vec{X}_2) \leq 1$. Therefore, this is a normalized quantifier for the interdependence of $\vec{X}_1$ and $\vec{X}_2$. This interdependence becomes maximal, when $S^{(k)}(\vec{X}_1|\vec{X}_2) \to 1$. The opposite dependence $S^{(k)}(\vec{X}_2|\vec{X}_1)$ is designed in analogy. By comparing $S^{(k)}(\vec{X}_1|\vec{X}_2)$ and $S^{(k)}(\vec{X}_2|\vec{X}_1)$, the more 'active' channel of the system can be determined. If $S^{(k)}(\vec{X}_1|\vec{X}_2) > S^{(k)}(\vec{X}_2|\vec{X}_1)$, i.e. $\vec{X}_1$ depends more on $\vec{X}_2$ than the other way around, $\vec{X}_2$ is called more active than $\vec{X}_1$. Still, this does not imply any causal interpretation in general.
Another measure is introduced by comparing $R_i^{(k)}(\vec{X}_1|\vec{X}_2)$ to the mean squared distances to random points

$$R_i(\vec{X}_1) = \frac{1}{n-1}\sum_{j\neq i}(\vec{x}_{1,i} - \vec{x}_{1,j})^2 \tag{5.50}$$

By using the geometrical average, a measure is

$$H^{(k)}(\vec{X}_1|\vec{X}_2) = \frac{1}{n}\sum_{i=1}^{n}\frac{R_i(\vec{X}_1)}{R_i^{(k)}(\vec{X}_1|\vec{X}_2)} \tag{5.51}$$

$H^{(k)}(\vec{X}_1|\vec{X}_2) = 0$ suggests (but does not proof) that $\vec{X}_1$ and $\vec{X}_2$ are completely independent, while $H^{(k)}(\vec{X}_1|\vec{X}_2) > 0$, if nearness in $\vec{X}_2$ implies nearness in $\vec{X}_1$ for equal time partners as well. $H^{(k)}(\vec{X}_1|\vec{X}_2) < 0$ holds, if close pairs in $\vec{X}_2$ correspond mainly to distant pairs in $\vec{X}_1$. This is very unlikely, but not impossible. The asymmetry under the exchange $\vec{X}_1 \leftrightarrow \vec{X}_2$ is the main difference between the discussed quantifiers and mutual information. For further discussion of the measures, the reader is referred to [48], and to [44]. Schmitz summarizes the theoretical findings of [48] in [1].

Chapter 6

Application: Diesel Engine

In this chapter, the presented methods are applied on time series measured in a diesel combustion engine. First, the manifold pressure p measured in hPa (hectopascal), and the emitted NO_X particles measured in ppm (parts per million), are treated as univariate separate time series. Secondly, the dependency between them is tested. A short description of a diesel engine, and the measurement procedure is provided as well. Both the material and the physical data have been kindly provided by the *Linz Center of Mechatronics GMBH*, and the *Institute for Design and Control of Mechatronical Systems*.

6.1 The Engine

This section is supposed to give a basic overview of the functionality of a diesel engine. For further description, the reader is referred to [55] and [56].

Basics. A diesel engine is an internal combustion engine, that initiates ignition to burn fuel by using the heat of compression. This differs from a petrol engine, that ignites a mixture of fuel and air by a spark plug. The fuel is injected into the combustion chamber after a certain pressure and heat[1] are reached. There are several different possibilities of fuel injection[2], but there is no necessity to go into detail. The more modern the engine, the better distributed are the injected fuel–droplets. Due to the heat of the compressed air, the fuel inflames, until all droplets are burned, and the process starts from beginning. The combustion in a Diesel engine results, next to several emission gases, in Diesel particulate matter (DPM). DPM is commonly known as black soot from exhaust, which consists of unburned carbon compounds. Black soot originiates in not fully–atomized fuel due to local low temperatures. These local low temperatures occur at the cylinder walls,

[1] typically about $40.000hPa$ and $550°C$

[2] pre–chamber injection

and at the outside of large fuel droplets. The mixture has less air to burn there, and some of the fuel turns into a carbon deposit. Some modern diesel engines feature diesel particulate filters, which catch a large part of the black soot, and are automatically regenerated by burning the particles when they are saturated.

The main particulate fraction of diesel exhaust consists of small particles. Because of their small size, inhaled particles may easily penetrate deep into the lungs. Also, the rough surfaces of these particles makes it easy for them to bind with other toxins in the environment. Thus, the hazards of particle inhalation increase, and can cause serious health issues. Many people suffering from soot–related health problems are close to diesel–powered engines most of their time, e.g. truckers, or railroad workers. In general population, adverse health effects have been observed as well [57] [58].

NO_X particles in exhaust are often examined instead of soot particles, because latter are very difficult to measure. The examination of dependencies within an engine may give an idea of how to make the engine more efficient, and to reduce the NO_X exhaust. Obviously, this cannot be achieved by using only manifold pressure observations. This thesis is rather supposed to give an idea of the capabilities of the presented methods.

Measurements. The test block used to simulate a diesel engine under realistic conditions consists of two parts: a dynamometer, which simulates the load, and a four–cylinder two–liter diesel combustion engine with *common rail injection*, variable geometry turbocharger, and exhaust gas recirculation (EGR). The common rail is a tube, that supplies each computer–controlled injector. EGR reduces the amount of NO_X that the combustion generates by recirculating a portion of an engine's exhaust gas back to the engine cylinders. The test block is dynamic, hence it is possible to simulate dynamic processes like changes in the load, or jumps in engine speed at shifts. Sensors measure different physical quantities in particular points, e.g.

- Temperature
- Air–pressure, e.g. manifold pressure p.
- Torsional moment
- Engine speed
- Airmass
- Exhaust gases, e.g. NO_X, HC, CO
- Angle of gas pedal

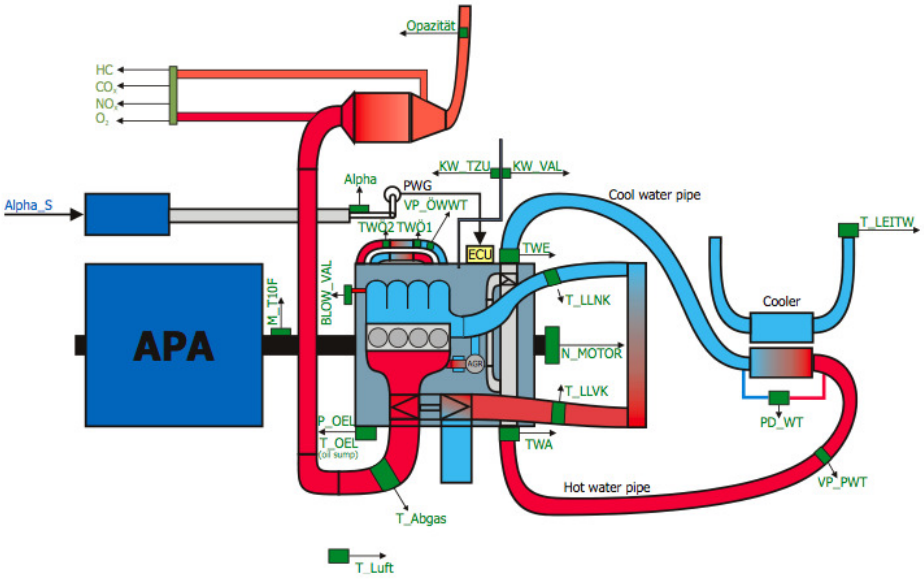

Figure 6.1: Sensors on the test block in Linz.

- Fuel–weight

The quantities applied are manifold pressure p, and number of NO_X particles. The data are measured in different setups, that simulate different states of the engine. The sampling time is $10ms$, and the sample sizes are in the order of 7×10^3 to 14×10^4, depending on the setup.

6.2 Procedure and Notations

This section is supposed to describe the applied test procedure, and to provide a list of notations and debreviations that are used in the next section.

Procedure. The manifold pressure and NO_X data are equidistantly measured with frequency $100Hz$[3]. Some of the figures show a different scale for better visibility of key aspects. The goal of this application is the *detection* of nonlinear dependencies in this two–channel system. At first, each channel is exmined and tested individually by means of univariate surrogate data. The observation of nonlinear behavior in a channel justifies multivariate methods, that are used afterwards. A list of algorithms for the construction of surrogates is given in the following paragraph, as well as the corresponding

[3] i.e. the data is observed every $10ms$.

null hypotheses. In each engine setup, there are at least two surrogate algorithms addressed. In this way, all methods are displayed through the four setups. A table informs about all test results in each section respectively. 1 signifies that a null hypothesis is approved, 0 indicates its rejection. As test statistic for univariate testing, an indicator of time reveral asymmetry T^{rev} is used. T^{rev} is an easy computable, though powerful quantifier (see 4.44). The test results are only valid in correspondence with this statistic, another quntifier may lead to different findings. Then the data sets are treated as bivariate system, in order to examine the relationship between channels. This is accomplished by bivariate surrogates. The test statistic that is used is the cross–correlation sum (see 5.20). As before, there is a table in the end of each section respectively, which contains the test results. The number of surrogates used for both univariate and multivariate testing is 39. This amount corresponds to a significance level $\alpha = 0.05$ for a two–sided test.(see section 3.3) Each of the engine setups displays a certain state of the engine, and can easily be interpreted as such by trained eye. The method of surrogates is data–driven only, hence any assumptions about these states are of no consideration.

Notations. The next sections include notations and abbreviations that require explanation. The following list provides both the tested null hypotheses and the algorithms, that are used to construct consistent surrogates. The range goes from highly restrictive to more flexible, for both univariate and multivariate data. The null hypotheses describe which underlying process is assumed.

$H_0^{(1u)}$**:** a linear Gaussian wide sense stationary process.

Methods: Unwindowed Fourier transform algorithm (FT)

$H_0^{(2u)}$**:** a monotonic nonlinear transformation of a linear Gaussian wide sense stationary process.

Methods: Amplitude adjusted Fourier transform algorithm (AAFT)

$H_0^{(3u)}$**:** a linear Gaussian wide sense stationary process measured by a instantaneous invertible function that does not depend on time.

Methods: Iterative amplitude adjusted Fourier transform algorithm (IAAFT)

$H_0^{(4u)}$**:** a linear wide sense stationary process.

Methods: Applied simulated annealing (SA)

$H_0^{(5u)}$**:** a composition of segment–wise linear Gaussian wide sense stationary process measured by a instantaneous invertible function that does not depend on time.

Methods: Iterative amplitude adjusted Fourier transform algorithm on stationary segments ($\mathrm{IAAFT}_{\mathrm{seg}}$)

$H_0^{(6u)}$**:** a linear wide sense stationary process with drifting mean and variance.

Methods: Applied simulated annealing ($\mathrm{SA}_{\mathrm{drift}}$)

$H_0^{(7u)}$**:** a linear Gaussian process.

Methods: Permutation of detail coefficients of the discrete wavelet transform ($\mathrm{DWT}_{\mathrm{p}}$)
Cyclic shift of detail coefficients of the discrete wavelet transform ($\mathrm{DWT}_{\mathrm{c}}$)
Block–wise shift of detail coefficients of the discrete wavelet transform ($\mathrm{DWT}_{\mathrm{b}}$)

$H_0^{(8u)}$**:** a linear Gaussian process measured by a instantaneous invertible function that does not depend on time.

Methods: IAAFT on detail coefficients of the maximum overlap discrete wavelet transform (WIAAFT)
IAAFT on (pinned) detail coefficients of the maximum overlap discrete wavelet transform (PWIAAFT)

$H_0^{(1b)}$**:** a bivariate linear Gaussian wide sense stationary process.

Methods: Biivariate unwindowed Fourier transform algorithm ($\mathrm{FT}_{\mathrm{bv}}$)

$H_0^{(2b)}$**:** a bivariate linear Gaussian wide sense stationary process measured by a instantaneous invertible function that does not depend on time.

Methods: Bivariate iterative amplitude adjusted Fourier transform algorithm ($\mathrm{IAAFT}_{\mathrm{bv}}$)

$H_0^{(3b)}$**:** a bivariate linear wide sense stationary process.

Methods: Bivariate applied simulated annealing ($\mathrm{SA}_{\mathrm{bv}}$)

$H_0^{(4b)}$**:** a bivariate linear Gaussian process.

Methods: Bivariate permutation of detail coefficients of the discrete wavelet transform ($\mathrm{DWT}_{\mathrm{bv}}$)

6.3 Setup 1

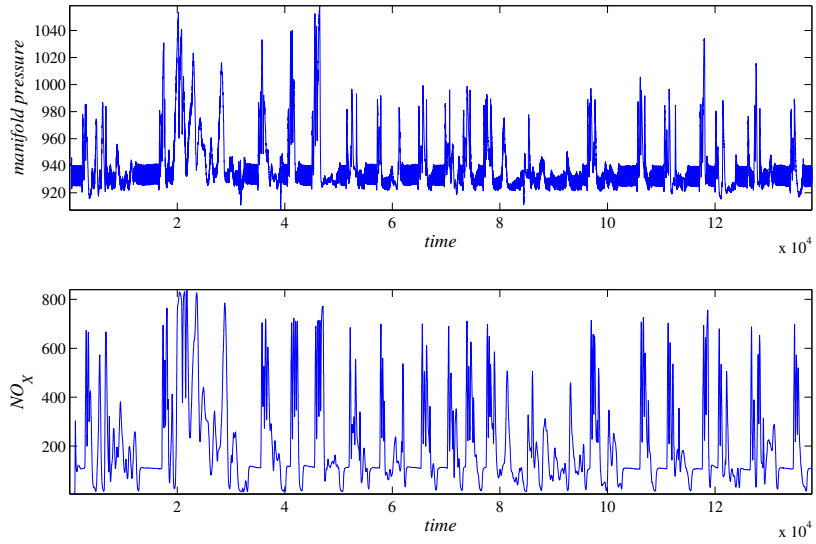

Figure 6.2: Setup 1: both the manifold pressure data (top) and the NO_X particle data (bottom) show evidence for nonstationarity, that may be difficult to distinguish from nonlinear dependencies.

6.3.1 Univariate Testing

The measurements of the first setup are displayed in figure 6.2. The data channels have length 13801, and are simultaneously measured. Comparatively high peaks in both panes are indicators for either nonlinearity or nonstationarity. There is only small jump discontinuity between the first and the last data points. Hence, there is no need to shorten the series. The unwindowed Fourier method is used to create surrogates for the manifold pressure channel. Figure 6.3 displays a typical example of a (centered) FT surrogate. Another surrogate, created by applied simulated annealing, is shown in figure 6.4. Whereas the FT surrogate does not reproduce the characteristics[4] of X_1 well, the SA surrogate seems to be more suitable. According to 4.25, the cooling parameters are set as follows: the maximum lag for the estimated auto–covariance equals 100, more long–term linear dependencies are neglected. The choice $a = 2$, $b = 1$, and $d = \frac{1}{50}$ signifies that half of the initial temperature is reached after $\frac{1}{50}$ of total steps. In this way, rather fast annealing is simulated. Truncation conditions are $T = 0.000001$, $E = 0.000001$, and

[4] e.g. high peaks

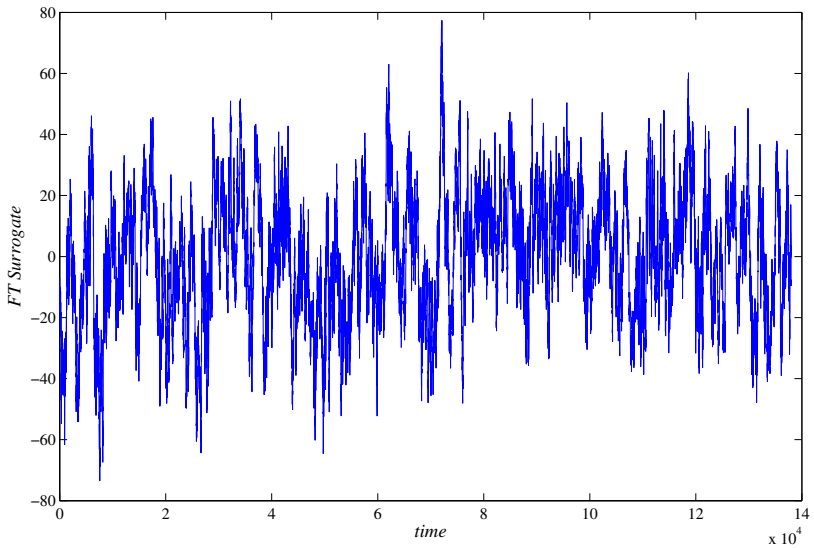

Figure 6.3: A typical unwindowed FT surrogate is displayed. Because the surrogate does not preserve basic characteristics of the manifold pressure data, the FT algorithm does not seem to be the right approach.

total number of temperature decrements equals $10 * n$. CPU time for a surrogate according to this annealing scheme is $\approx 3h$. Although this is a rather fast cooling scheme, good results cannot be reached within shorter time. Thus, this method is unattractive when good results can be accomplished by other algorithms as well. Figure 6.5 displays the test result of FT surrogates. The test statistic differs significantly from the empirical distribution, which is computed on 39 surrogates. The null hypothesis, that X_1 is realization of a linear Gaussian wide sense stationary process, can be rejected.

Figure 6.6 displays an AAFT surrogate for the NO_X channel. As before, the surrogate does not seem to feature the overall characteristics of X_2. The test statistic calculated on X_2 is much lower than its estimated distribution (see 6.7). The null hypothesis, that X_2 is realization of a monotonic nonlinear transformation of a linear Gaussian wide sense stationary process is rejected. This result provides no guarantee for nonlinearity. Any other condition, like stationarity, could be unmet as well.

6.3.2 Bivariate Testing

A bivariate FT surrogate is displayed in figure 6.8. Identical randomized phases are added to the Fourier phases of both channel respectively. In

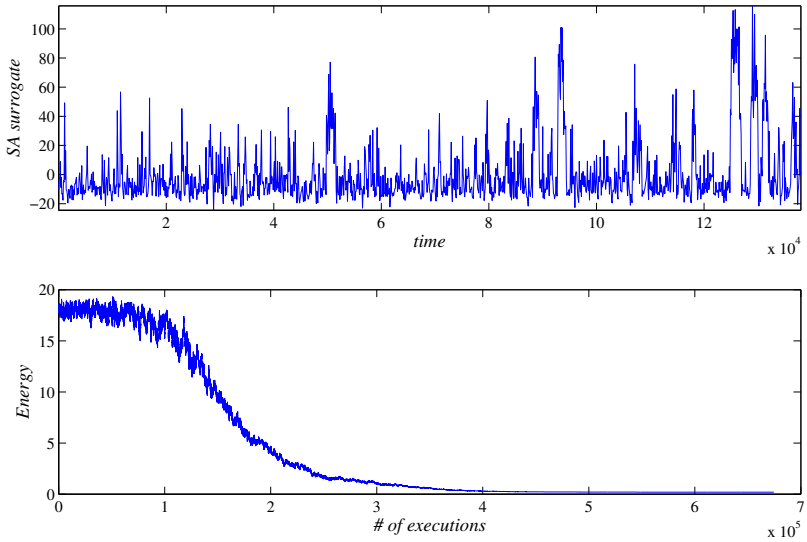

Figure 6.4: A surrogate constructed by applied simulated annealing is shown on top, and the corresponding energy function is displayed on the bottom. The basic data structure is better preserved than by the surrogate displayed in figure 6.3. The price for achieving this is a high computing time. In order to reach an acceptable energy, lots of algorithm steps have to be taken.

	Manif. Pr.	**NO_X**	**Procedure**
$H_0^{(1u)}$	0	0	FT
$H_0^{(2u)}$	0	0	$AAFT$
$H_0^{(3u)}$	1	0	$IAAFT$
$H_0^{(4u)}$	1	0	SA
$H_0^{(5u)}$	1	1	$IAAFT_{seg}$
$H_0^{(6u)}$	1	1	SA_{drift}
$H_0^{(7u)}$	0	0	DWT_p, DWT_c, DWT_b
$H_0^{(8u)}$	1	1	$(P)WIAAFT$

Table 6.1: Test results for the respective univariate channel in setup 1 demonstrate the difficulties of choosing the right surrogate method. The results of the three standard methods cannot be seen as indicators for nonlinearity, because their null hypotheses include stationarity as well. Segment–wise IAAFT, SA with moving average in the cost function, and (P)WIAAFT surrogates approve the corresponding hypotheses.

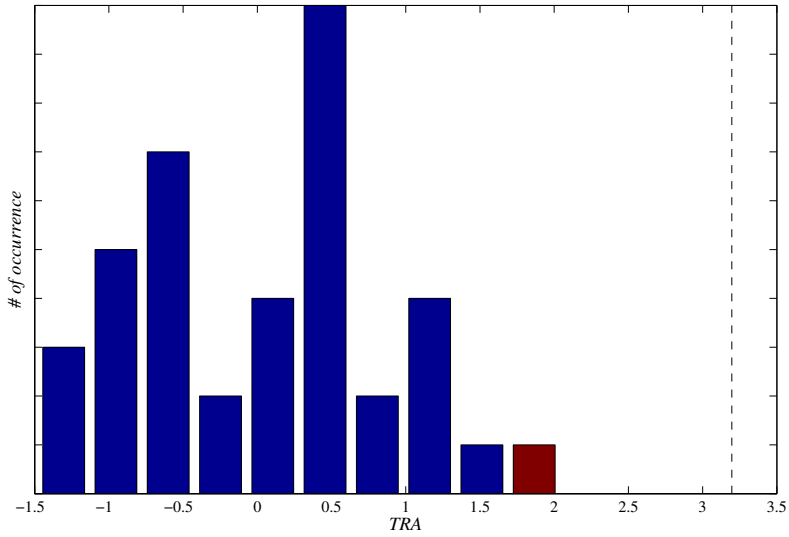

Figure 6.5: The empirical distribution of time reversal asymmetry, which is computed on 39 FT surrogates, is shown. The red bar indicates the 'region' of T^{rev} in the histogram. The bar furthest to the left of the histogram includes values up to $-\infty$, and the bar furthest to the right includes values up to ∞. The dotted line marks the actual value of T^{rev}. In this case, T^{rev} is the largest value, hence the null hypothesis $H_0^{(1u)}$ is rejected.

	Result	**Procedure**
$H_0^{(1b)}$	0	FT_{bv}
$H_0^{(2b)}$	0	$IAAFT_{bv}$
$H_0^{(3b)}$	0	SA_{bv}
$H_0^{(4b)}$	0	DWT_{bv}

Table 6.2: Test results for the bivariate system in setup 1 are distinct. None of the null hypotheses is approved, and nonlinear structures in the data can be assumed.

this way, linear structure within and between the surrogate channels are preserved. Any other relations are not reproduced on the surrogates. This method corresponds to the null hypothesis, that X is realization of an underlying bivariate linear Gaussian wide sense stationary stochastic process. Rejection of the null hypothesis (see figure 6.9) can be caused by occurring nonstationarities in data.

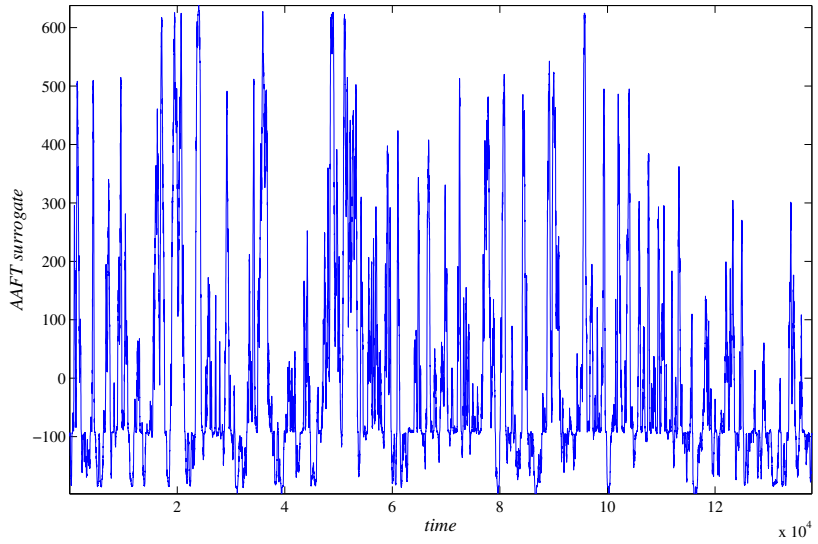

Figure 6.6: A typical surrogate created by AAFT.

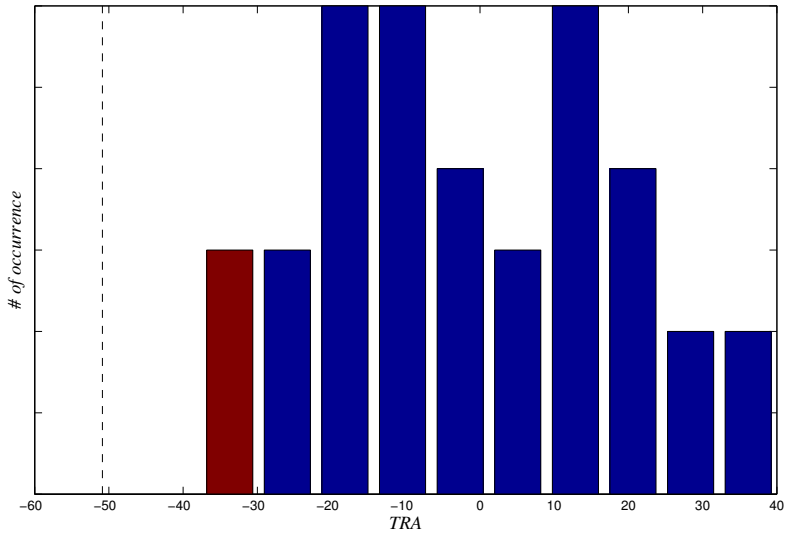

Figure 6.7: Time reversal asymmetry does not correspond to its empirically estimated distribution.

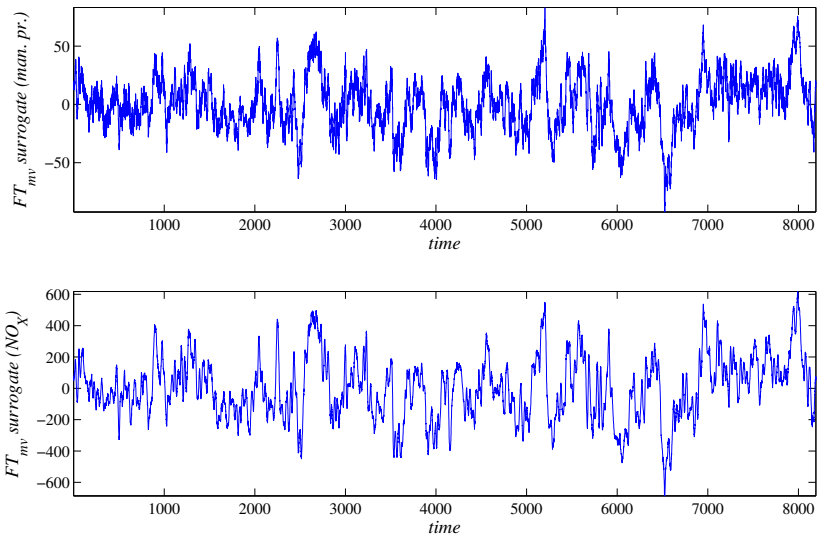

Figure 6.8: A bivariate FT surrogate features linear characteristics in and between data channels. Any other structures are not preserved.

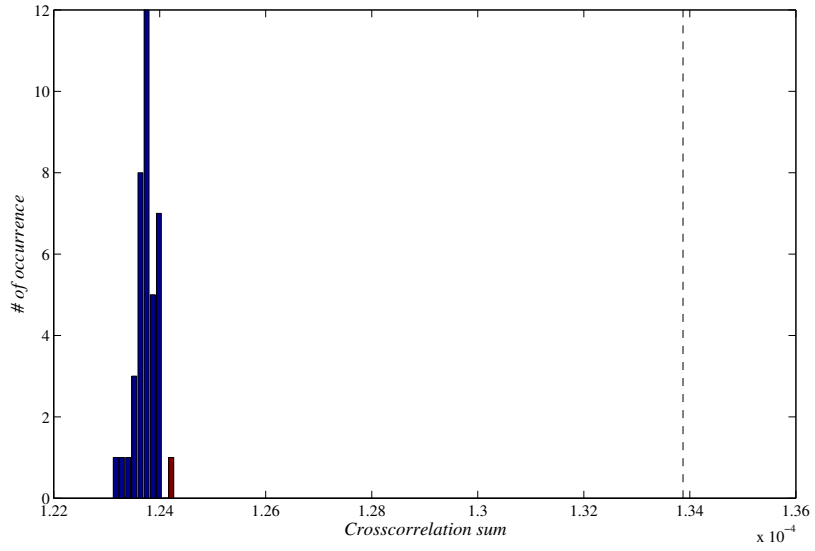

Figure 6.9: Cross–correlation sum differs significantly from the empirical distribution, that is estimated by bivariate FT surrogates. The null hypothesis is rejected.

6.4 Setup 2

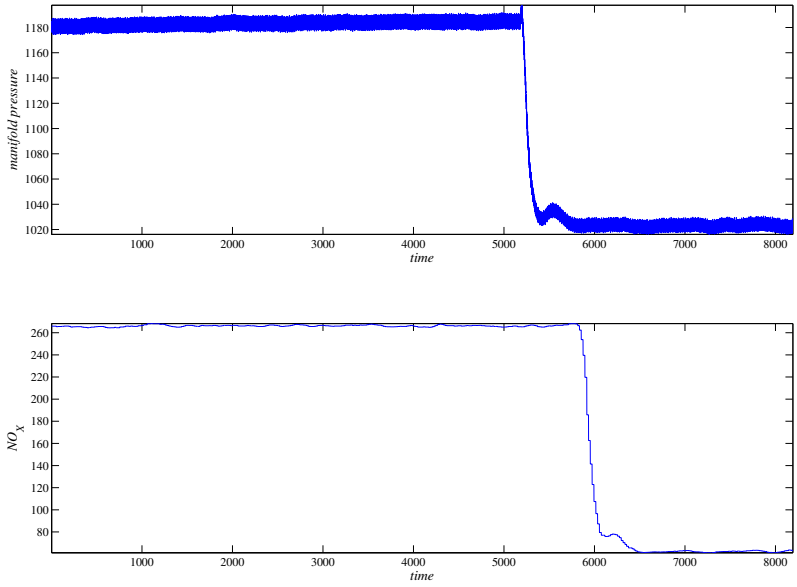

Figure 6.10: Data in setup 2 display an abrupt change, that indicates a change in activity of the engine. This nonstationary behavior may lead to a falsification of test results.

6.4.1 Univariate Testing

Data channels in the second engine setup consist of 2^{13} measurements. Both channels experience a structure break, manifold pressure at $\approx$ 5300, and NO_X at $\approx$ 6000 with a delay (see figure 6.10). The break may indicate a change in the engine's activity, e.g. a change into another gear. Anyway, most methods have difficulties to preserve structure breaks on surrogates. Figure 6.11 displays a typical surrogate, that is created by a cyclic shift of detail coefficients of data. The jump manifests itself both as a small differently located jump and as other spurious remnants in the surrogate. IAAFT surrogates on segments with length 1000 seem to preserve characteristics of X_1 well. Figure 6.12 shows a typical example for an $\text{IAAFT}_{\text{seg}}$ surrogate. The break in structure has similar location, and behavior on other segments is preserved as well. A disadvantage of this procedure is the loss of all dependencies between segments. Nonetheless, if there are only few structure breaks in data, then the $\text{IAAFT}_{\text{seg}}$ algorithm is an appropriate way to cre-

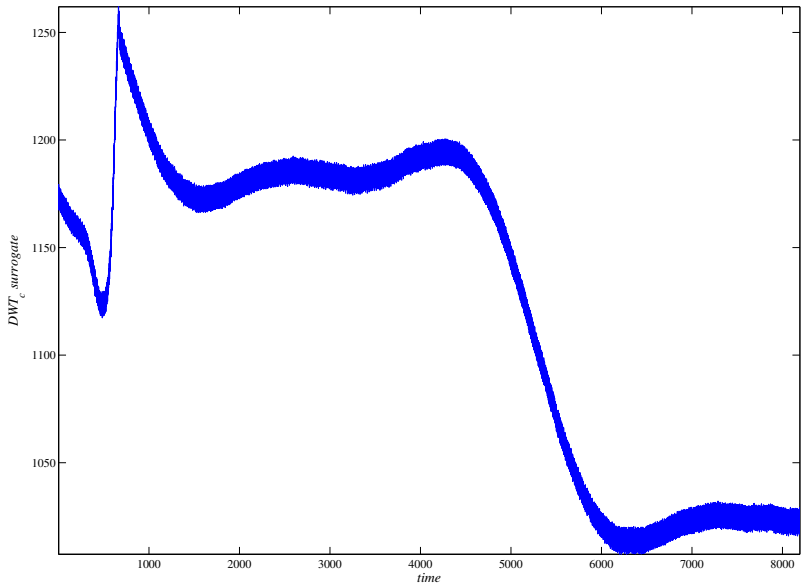

Figure 6.11: A typical surrogate, created by a cyclic shift of the detail coefficients, is displayed. There is evidence for nonstationarity in the surrogate as well, but elsewhere.

ate surrogates.
Figure 6.13 pictures test results of $\text{IAAFT}_{\text{seg}}$ surrogates. The null hypothesis, that a segment–wise linear Gaussian wide sense stationary process is underlying, is approved.

The surrogate depicted in figure 6.14 is constructed by the IAAFT algorithm. The corresponding null hypothesis says that the underlying process is linear, Gaussian, wide sense stationary, and measured by an instantaneous invertible function, that does not depend on time. The structure break in data is reproduced as two jumps in the surrogate. Clearly, this is not intended. The test result is displayed in figure 6.15.

6.4.2 Bivariate Testing

The bivariate surrogate in figure 6.16 is constructed by the bivariate IAAFT algorithm. This method has the same difficulties as the univariate IAAFT method, and is not able to preserve structure breaks. Due to a minimal distance between the random Fourier phases of X_1 and X_2, both channels show a similar behavior. Null hypothesis $H_0^{(2b)}$ is rejected (see figure 6.17),

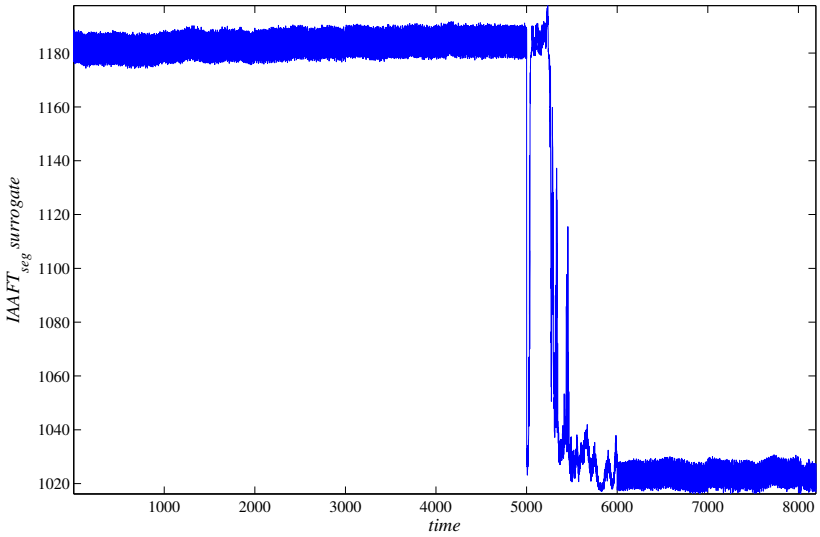

Figure 6.12: In order to reproduce the jump in data, segment–wise surrogates are created and fit together. In this way, the resulting surrogates indicate nonstationary behavior similar to that of the original data.

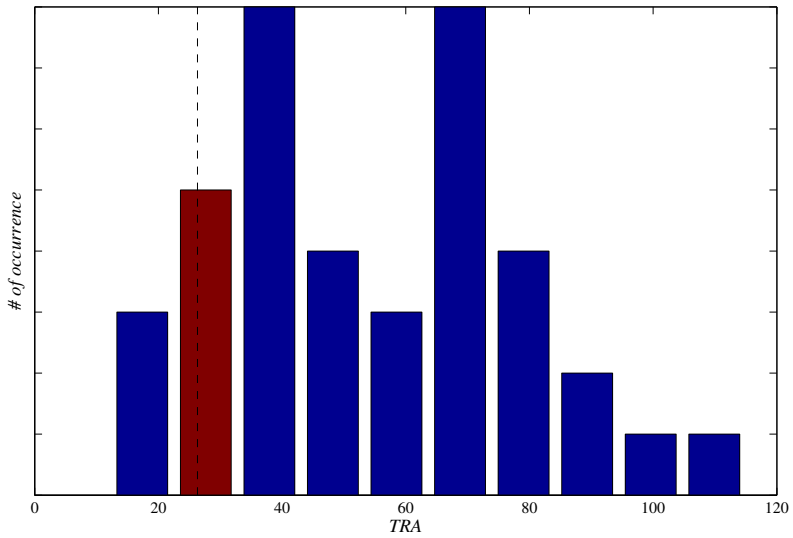

Figure 6.13: TRA test statistic corresponds to the distribution, that is empirically estimated by 39 $\text{IAAFT}_{\text{seg}}$ surrogates. The null hypothesis, that includes a drifting mean, is approved.

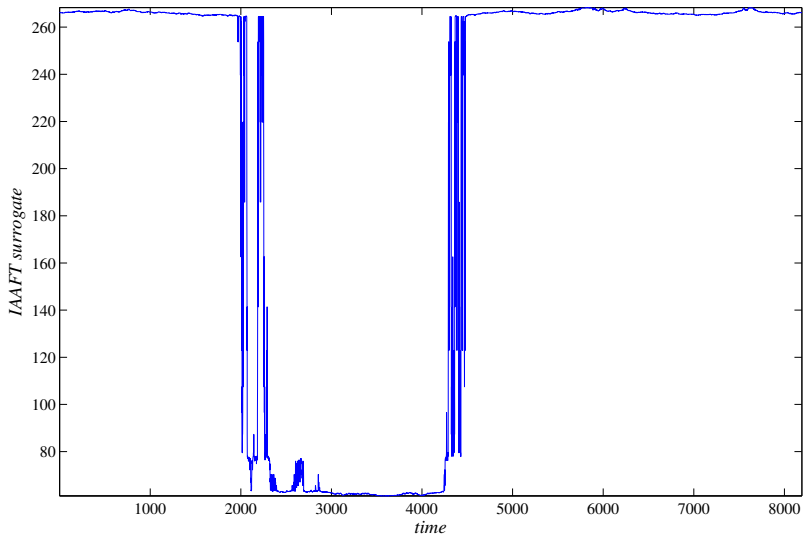

Figure 6.14: The IAAFT algorithm is not able to preserve nonstationarities occuring in NO_X channel in setup 2.

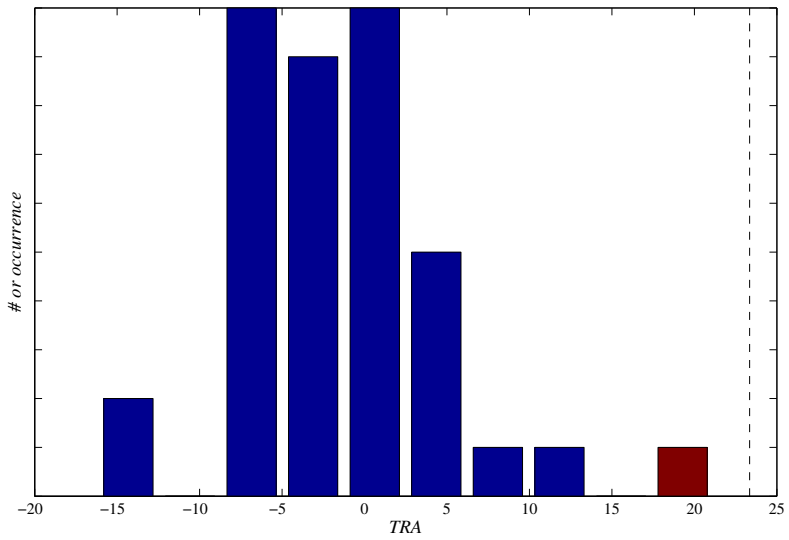

Figure 6.15: Test result for 39 AAFT surrogates: the null hypothesis of an underlying linear Gaussian wide sense stationary process measured by an instantaneous invertible function, that does not depend on time, is rejected.

	Manif. Pr.	$\mathbf{NO_X}$	**Procedure**
$H_0^{(1u)}$	0	0	FT
$H_0^{(2u)}$	0	0	$AAFT$
$H_0^{(3u)}$	1	0	$IAAFT$
$H_0^{(4u)}$	0	1	SA
$H_0^{(5u)}$	1	1	$IAAFT_{seg}$
$H_0^{(6u)}$	0	1	SA_{drift}
$H_0^{(7u)}$	0	0	DWT_p, DWT_c, DWT_b
$H_0^{(8u)}$	1	1	$(P)WIAAFT$

Table 6.3: Test results for univariate channels in setup 2 is amiguous. Only the null hypotheses that correspond to $\text{IAAFT}_{\text{seg}}$ and (P)WIAAFT surrogates are approved on both channels. Former do not consider any correlations between segments, and this result has to be questioned. According to these results, nonlinear bivariate methods should be at least considered.

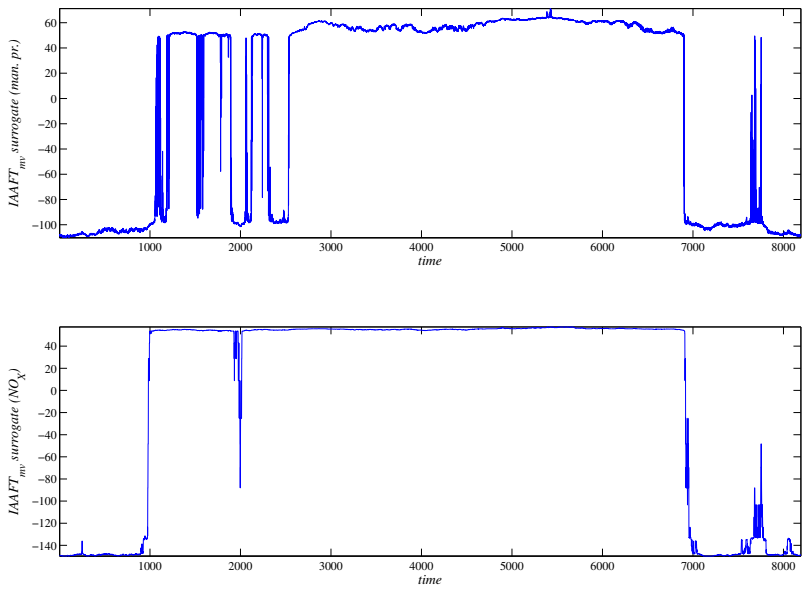

Figure 6.16: A typical bivariate surrogate of setup 2 is shown, which is created by the bivariate IAAFT method.

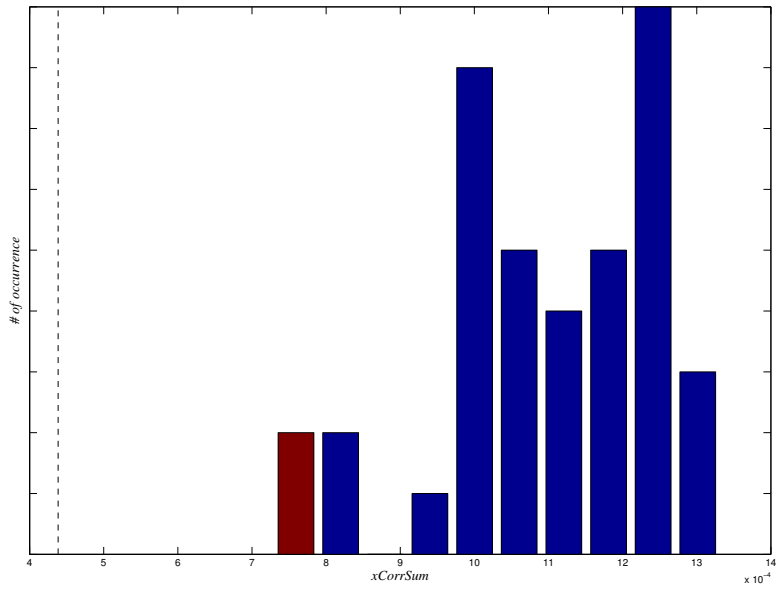

Figure 6.17: Test result for IAAFT surrogates of setup 2.

	Result	**Procedure**
$H_0^{(1b)}$	0	FT_{bv}
$H_0^{(2b)}$	0	$IAAFT_{bv}$
$H_0^{(3b)}$	1	SA_{bv}
$H_0^{(4b)}$	0	DWT_{bv}

Table 6.4: Test result of setup 2: only the null hypothesis that corresponds to simulated annealing with integrated cross–correlation is approved. Note, that the break in data behavior may distort this result.

and a time independent instantaneous invertible function is not suited to explain data behavior sufficiently.

6.5 Setup 3

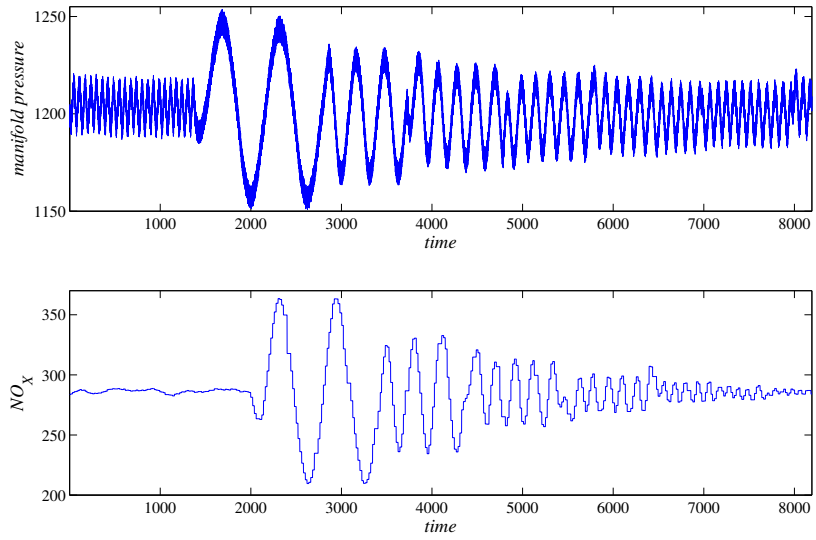

Figure 6.18: Both manifold pressure and NO_X particles show evidence of an indicated oscillation. In fact, this sinusoidal oscillation with increasing frequency is artificially induced to the system.

6.5.1 Univariate Testing

Both channels displayed in figure 6.18 show evidence for an oscillation with increasing frequency. A random permutation of detail coefficients of X_1 is used to construct surrogates like that in figure 6.19. This method is suited to preserve main characteristics, such as trend, but fails to preserve details at higher frequencies. Time reversal asymmetry deviates from its estimated distribution. Hence, an underlying linear Gaussian process is not qualified to explain the data behavior adequately.

Figure 6.21 shows a surrogate, that is created by a cyclic shift of detail coefficients at each scale individually. This method is able to reproduce characteristics of X_2 on a surrogate well, but with a shift in time. Nevertheless, a very similar look to the original data reached. In this case, the null hypothesis, that a linear Gaussian process is underlying, is approved (see figure 6.22). An elaborate choice of surrogate algorithm is crucial for qualitative testing.

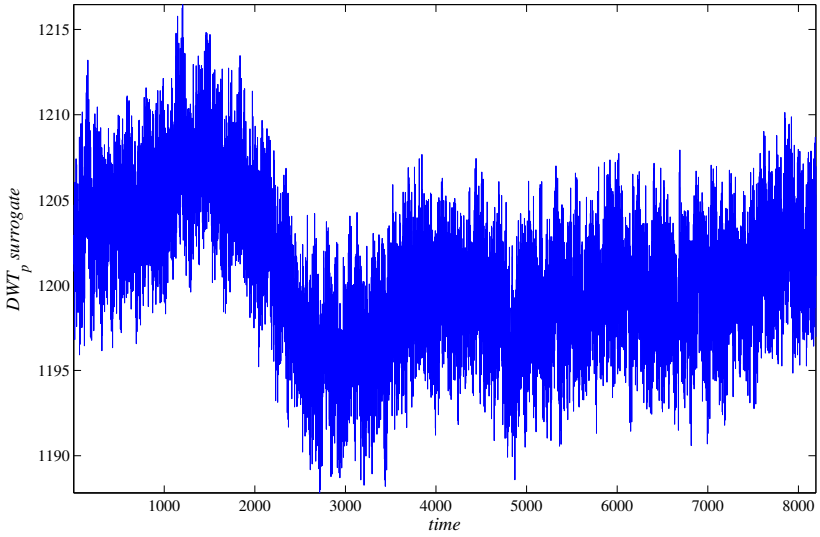

Figure 6.19: A random permutation of detail coefficients at each scale individually is suited for preserving the overall trend of data, but not for preserving more detailed characteristics, like high frequency oscillations in setup 3.

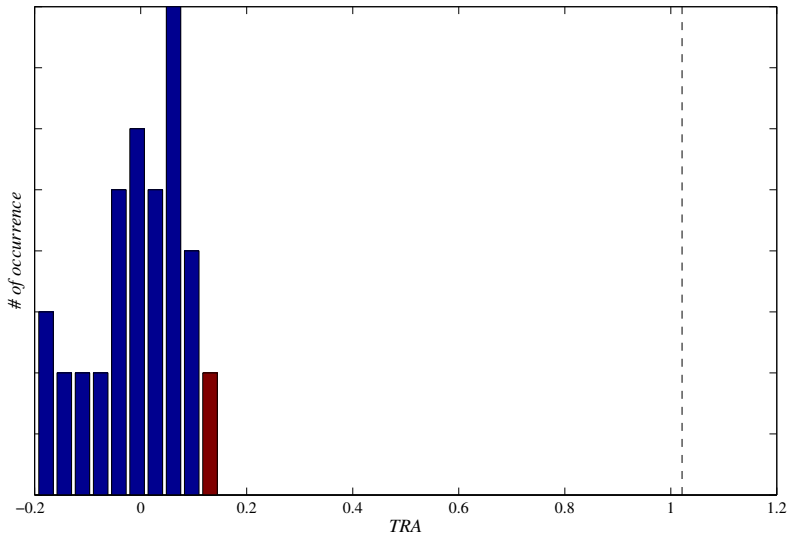

Figure 6.20: Test result for $\mathrm{DWT_p}$ surrogates of setup 3.

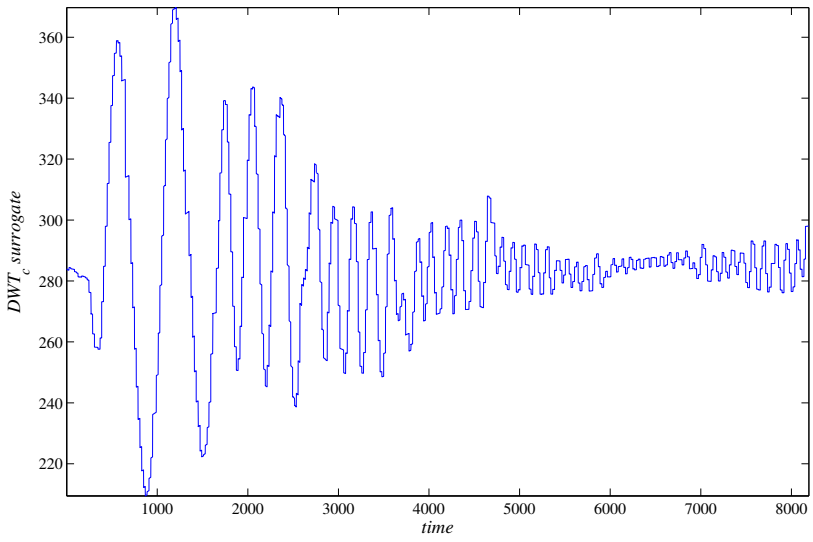

Figure 6.21: A cyclic shift of detail coefficients at each scale individually is suited to preserve oscillations in setup 3. Result is very look–alike surrogates.

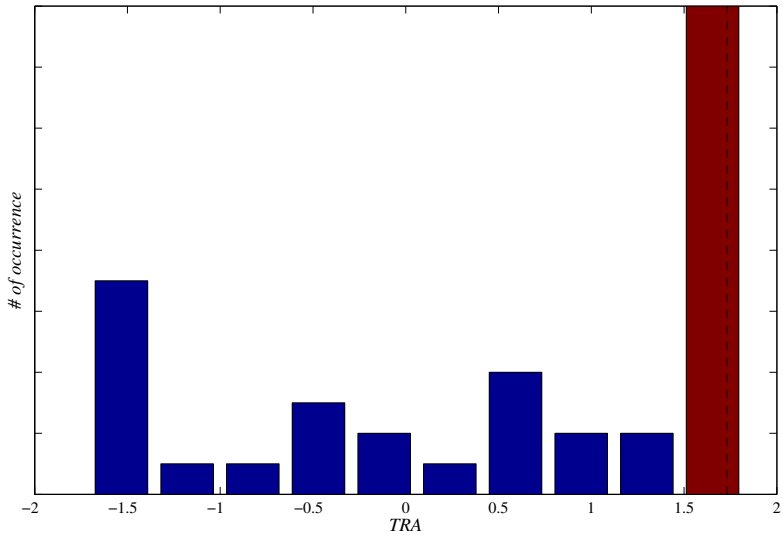

Figure 6.22: The TRA quantifier is in line with its estimated distribution. Null hypothesis $H_0^{(7u)}$ is approved.

	Manif. Pr.	$\mathbf{NO_X}$	**Procedure**
$H_0^{(1u)}$	1	1	FT
$H_0^{(2u)}$	1	1	$AAFT$
$H_0^{(3u)}$	1	1	$IAAFT$
$H_0^{(4u)}$	1	1	SA
$H_0^{(5u)}$	1	1	$IAAFT_{seg}$
$H_0^{(6u)}$	1	1	SA_{drift}
$H_0^{(7u)}$	0	0	DWT_p, DWT_c, DWT_b
$H_0^{(8u)}$	1	1	$(P)WIAAFT$

Table 6.5: Test results for setup 3 show a tendency towards approving the range of null hypotheses. Even the most restrictive of them is accepted. Only DWT algorithms lead to a rejection. The oscillations in data may bias these results.

	Result	**Procedure**
$H_0^{(1b)}$	1	FT_{bv}
$H_0^{(2b)}$	1	$IAAFT_{bv}$
$H_0^{(3b)}$	1	SA_{bv}
$H_0^{(4b)}$	0	DWT_{bv}

Table 6.6: The bivariate surrogates for setup 3 display an approval of most null hypotheses. The result is consistent with univariate testing, where the DWT algorithms lead to a rejection of the corresponding H_0 as well.

6.5.2 Bivariate Testing

The bivariate extension of $\mathrm{DWT_p}$ algorithm preserves overall trends both in and between channels. However, surrogates do not show the same oscillations as X (see figure 6.23). This may be a possible explanation for rejection of the null hypothesis (see figure 6.24). A linear process may not be able to explain the oscillations sufficiently.

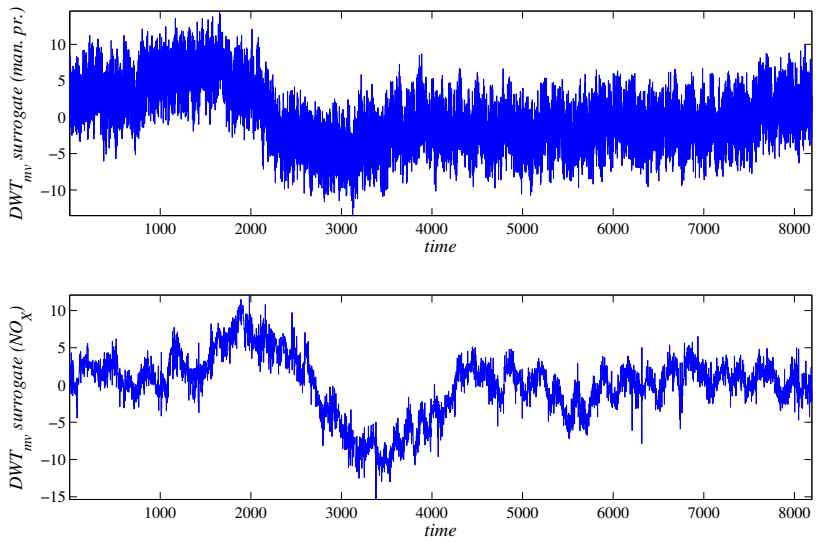

Figure 6.23: The bivariate DWT algorithm preserves the overall trend in data, but is not able to reproduce fine oscillations.

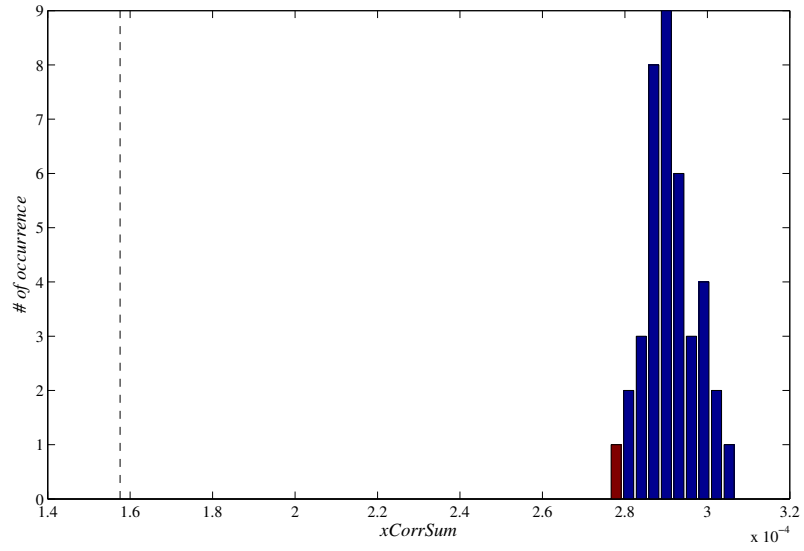

Figure 6.24: Null hypothesis $H_0^{(4b)}$ is rejected.

6.6 Setup 4

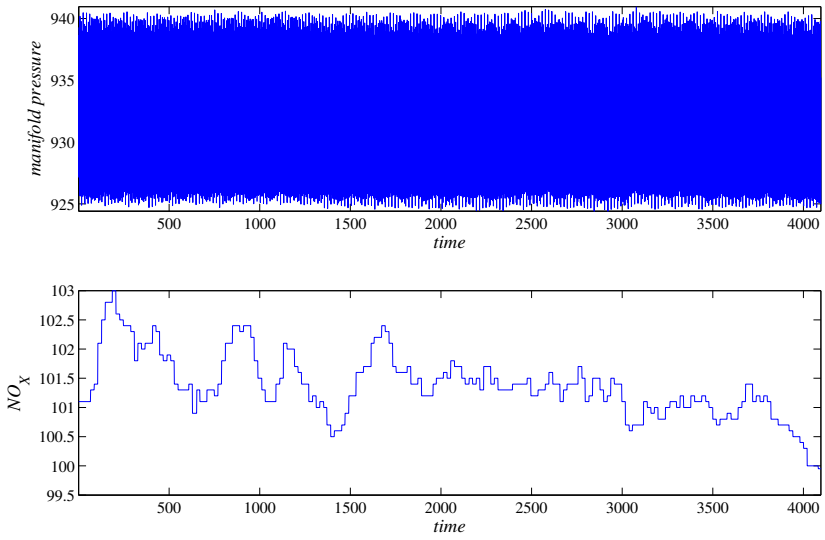

Figure 6.25: The manifold pressure is highly vibrating in channel 4, whereas NO_X values seem to experience a drifting mean.

6.6.1 Univariate Testing

In setup 4, channels show differing behavior. X_1 is a highly vibrating signal, that does not show any irregularities in mean, or variance. X_2 shows a drifting mean, i.e. a decreasing relative amount of NO_X ppm (see figure 6.25). Figure 6.26 displays a PWIAAFT surrogate for X_1. Since the time series has no significant peaks or irregularities, every 15^{th} measurement is pinned. In this way, surrogates with the same vibrations as X_1 are constructed. The TRA clearly corresponds to its empirically estimated distribution (see figure 6.27), hence this data series a linear process seems to accurately describe the data structure .

As mentioned, NO_X channel shows a drifting mean, that has to be reproduced on surrogates, in order to get convincing test results. This is accomplished by introducing a moving average of length 50, and an overlap of 10 into the cost function of applied simulated annealing. A slow cooling scheme is chosen: 1000 accepted annealing steps are needed, before temperature decrement happens. Truncation criterion is $T = 0.01$. A typical surrogate can be seen in figure 6.28. The corresponding null hypothesis, that

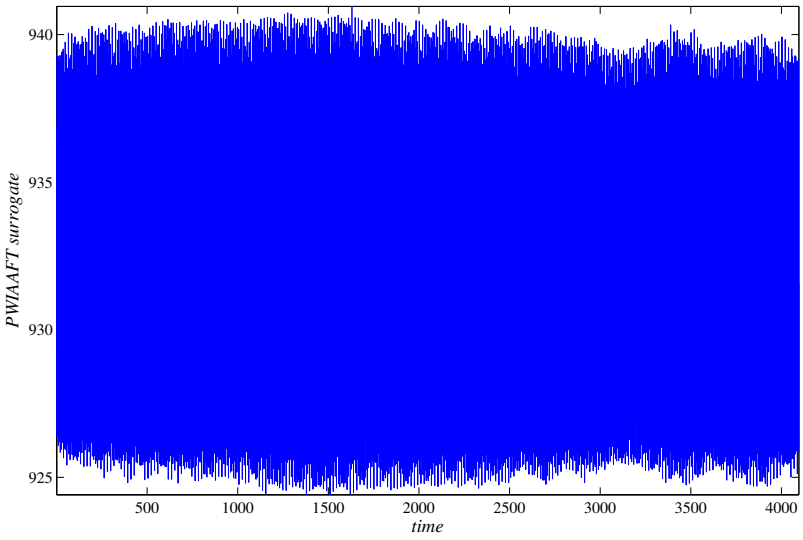

Figure 6.26: The PWIAAFT algorithm, with every 15^{th} measurement pinned, produces a surrogate similar to X_1.

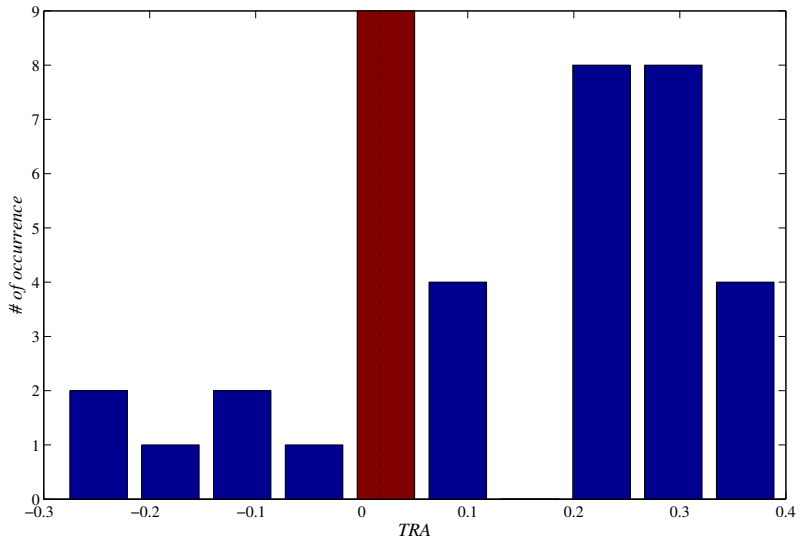

Figure 6.27: The TRA corresponds to its empirically distribution, estimated by 39 PWIAAFT surrogates.

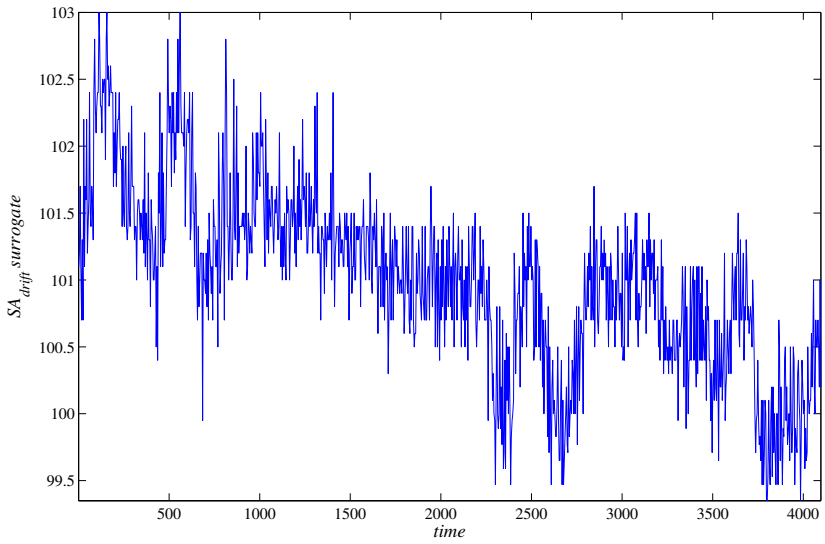

Figure 6.28: The implementation of a moving average in the energy function of simulated annealing enables to reproduce a drifting mean on a surrogate.

	Manif. Pr.	$\mathbf{NO_X}$	**Procedure**
$H_0^{(1u)}$	1	1	FT
$H_0^{(2u)}$	1	1	$AAFT$
$H_0^{(3u)}$	1	1	$IAAFT$
$H_0^{(4u)}$	1	0	SA
$H_0^{(5u)}$	1	0	$IAAFT_{seg}$
$H_0^{(6u)}$	1	1	SA_{drift}
$H_0^{(7u)}$	1	0	DWT_p, DWT_c, DWT_b
$H_0^{(8u)}$	1	1	$(P)WIAAFT$

Table 6.7: Test results of setup 4. There is clear evidence for linearity in the first channel. X_1 does not show irregular nonstationarities that could bias the result. The outcome of the tests for X_2 are different. SA, $\text{IAAFT}_{\text{seg}}$, and DWT algorithms lead to a rejection of the corresponding null hypotheses. Reason could be the drifting mean.

the underlying process is linear with drifting mean, is approved (see figure 6.29)

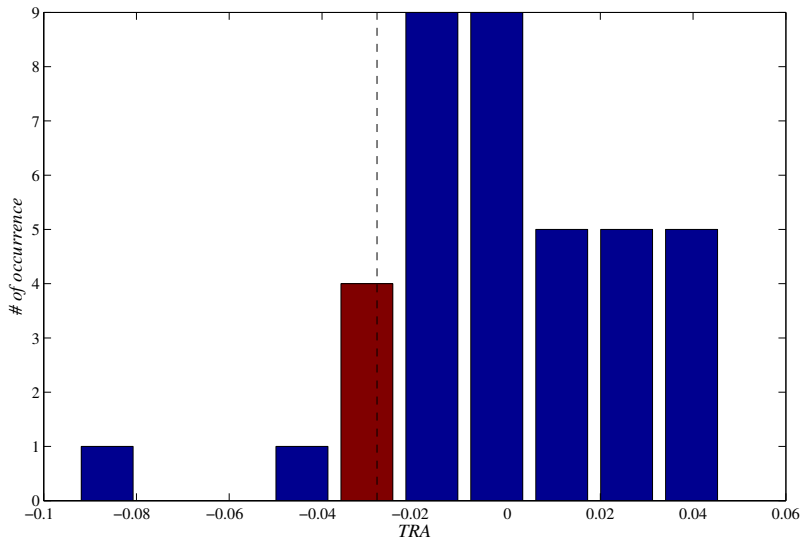

Figure 6.29: The null hypothesis including a drifting mean is approved.

	Result	**Procedure**
$H_0^{(1b)}$	1	FT_{bv}
$H_0^{(2b)}$	1	$IAAFT_{bv}$
$H_0^{(3b)}$	0	SA_{bv}
$H_0^{(4b)}$	1	DWT_{bv}

Table 6.8: Most of the null hypotheses are approved by the bivariate surrogates of setup 4. Hence, an underlying linear dynamic seems feasible. However, these findings have to be treated cautiously.

6.6.2 Bivariate Testing

Bivariate surrogates are created by implementing estimated cross–covariances into the energy function of simulated annealing (see figure 6.30). Reproducing only the linear characteristics of X does not seem to capture the structure of the bivariate system very well. Cross–correlation sum cleary differs from its empirically estimated distribution. This is depicted in figure 6.31.

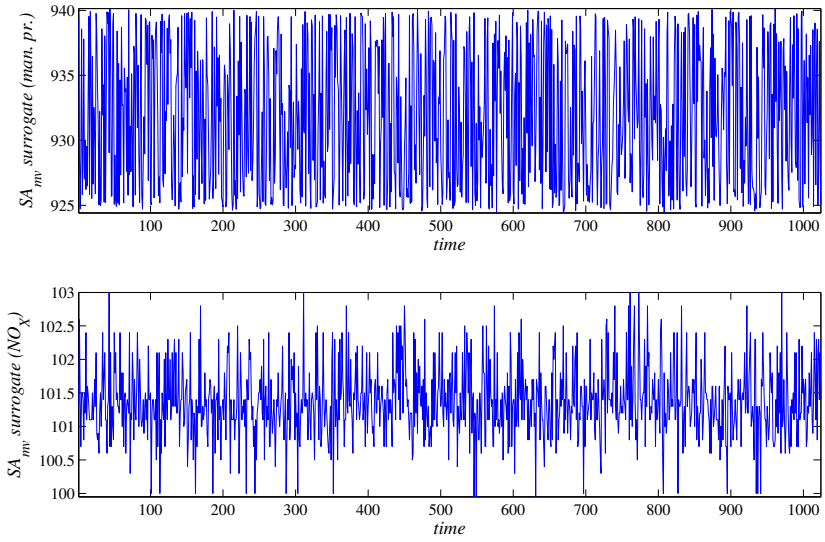

Figure 6.30: The lower channel of a bivariate SA surrogate differs from X_2, whereas the upper channel and X_1 show similar behavior.

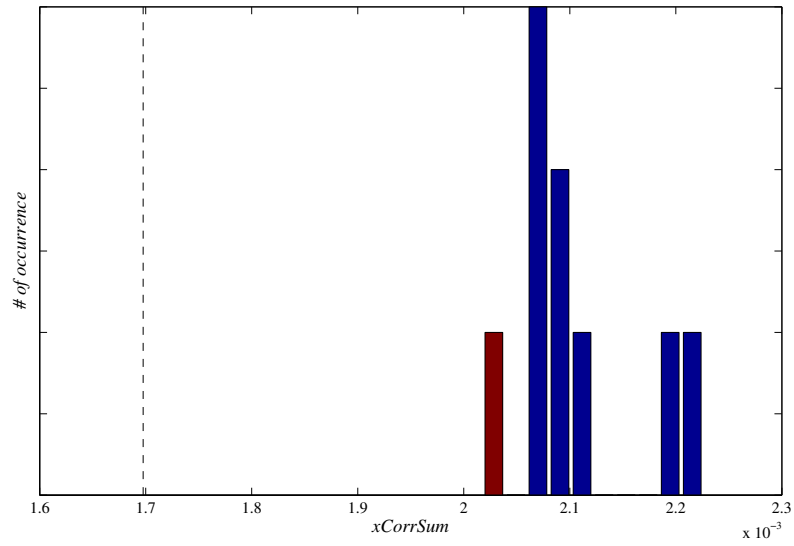

Figure 6.31: Cross–correlation sum calculated on X differs from its estimated distribution.

Chapter 7

Conclusion

Throughout this thesis, different techniques to create surrogates for testing of nonlinearity have been presented. First, standard Fourier–based methods have been discussed. They are based on destruction of both nonlinear and nonstationary dependencies through Fourier phase randomization. Then, a more sophisticated method has implemented an additional Fourier–amplitude adjustment step, in order to assimilate the point–wise distribution of a surrogate. This allows to test the possibility of a nonlinear function that measures a linear process. Afterwards, a technique coming from the field of thermo–dynamics has been introduced: applied simulated annealing. This method is more flexible than the standard methods, and enables reproduction of certain characteristics with arbitrary exactness. The price for exactness is a high computing time, and a vast amount of parameters, that have to be fine–tuned. In order to exclude a bias in test results, that comes from nonstationarity, other methods have been discussed. The most important class are wavelet–based surrogates. Detail coefficients of the (maximal ovelap) discrete wavelet transform are settled in the time–phase plane. Their manipulation preserves both linear characteristics and certain nonstationarities in data. The difficulty is to decide whether linear characteristics are reproduced well enough. Extensions of these methods have been discussed as well, together with a list of corresponding null hypotheses that range from highly restrictive to acceptably general. Basic test statistics have been addressed as well. Which of them should be used has to be decided each time individually. All these methods are powerful in statistical hypothesis testing, and have been applied on different data of a diesel combustion engine. The difficulty to choose the right surrogate algorithm has been demonstrated. Most of the difficulties origin in nonstationary behavior of the time series. Considering growing CPU power, simulated annealing may be a valuable tool in future nonlinearity testing. This method could help to understand and interpret nonlinear systems with different channels influencing each other in a better way. In context of combustion engines, statistical techniques, like the

method of surrogate data, are valuable tools for aiming goals like decreasing pollution, and increasing efficiency.

Appendix A

Algorithms and Methods

Appendix A provides an overview of important algorithms to create surrogate data, and some useful MATLAB® source code extracts as well.

A.1 (Un)Windowed Fourier Transform Algorithm

Algorithm A.1: (W)FT

1: Insert data $X \in \mathbb{R}^{(1\times n)}$
   ▷ if not $x_1 \approx x_n$, windowing should be applied

2: $X \overset{DFT}{\longrightarrow} \mathcal{F}(X)$
   ▷ DFT... Discrete Fourier transform

3: $\mathcal{F}(X) \longrightarrow \tilde{\mathcal{F}}(X)$: replace ϕ_k by random numbers in $[0, 2\pi)$
   ▷ $k = -\lfloor(\frac{n-1}{2})\rfloor, \ldots, \lfloor(\frac{n}{2})\rfloor$
   ▷ ϕ_k... Fourier phases

4: $\tilde{\mathcal{F}}(X) \overset{IDFT}{\longrightarrow} S$
   ▷ IDFT... Inverse discrete Fourier transform

5: **return**

MATLAB® computes the (inverse) discrete Fourier transform by common (inverse) fast Fourier transform algorithm. The function `fft` returns a $1 \times n$–dimensional vector of complex numbers. The corresponding Fourier amplitudes are computed by `amp=abs(fft(X))`. The sequence `tmp=2*pi*rand(1,n/2)` returns a row vector of $n/2$ pseudo–random numbers in $[0, 2\pi)$. Alternatively, if n is odd, `tmp=2*pi*rand(1,(n-1)/2)` has to be used. In order to get a real–valued surrogate, the randomized phases have to be symmetrized. This can be achieved by the sequence `phase=[tmp,-1*fliplr(tmp)]`, where `fliplr(tmp)` returns the phases with columns flipped about a vertical axis.

Finally, the sequence `ifft(amplitude.*(cos(phase)+1i*sin(phase)))` returns the surrogate.

A.2 Amplitude Adjusted FT Algorithm

Algorithm A.2: AAFT

1: Insert data $X \in \mathbb{R}^{(1\times n)}$

2: create Gaussian distributed random vector $Y \in \mathbb{R}^{(1\times n)}$

3: $Y \longrightarrow \tilde{Y}$: rescale Y according to X
▷ match indices according to the ascending order of X

4: $\tilde{Y} \overset{DFT}{\longrightarrow} \mathcal{F}(\tilde{Y})$
▷ DFT... Discrete Fourier transform

5: $\mathcal{F}(\tilde{Y}) \longrightarrow \tilde{\mathcal{F}}(\tilde{Y})$: replace ϕ_k by random numbers in $[0, 2\pi)$
▷ $k = -\lfloor(\frac{n-1}{2})\rfloor, \ldots, \lfloor(\frac{n}{2})\rfloor$
▷ ϕ_k... Fourier phases

6: $\tilde{\mathcal{F}}(\tilde{Y}) \overset{IDFT}{\longrightarrow} \widehat{Y}$
▷ IDFT... Inverse discrete Fourier transform

7: $X \longrightarrow S$: rescale X according to $\widehat{Y}$
▷ match indices according to the ascending order of $\widehat{Y}$

8: **return**

Creating a vector of n Gaussian distributed random numbers can easily be done by the MATLAB® function `randn(1,n)`. A rescaled vector of X is computed by the sequence `[B,ind]=sort(X)`. `ind`$\in \mathbb{N}^{(1\times n)}$ is a permutation vector of the corresponding indices of X with `B=X(ind)`.

A.3 Iterative AAFT Algorithm

Algorithm A.3: IAAFT

1: Insert data $X \in \mathbb{R}^{(1\times n)}$

2: $X \longrightarrow \tilde{X}$: sort X in ascending order

3: $X \longrightarrow S^{(0)}$: random permutation of X
▷ Initializing

4: **while** $S^{(i+1)} \neq S^{(i)}$ **do**

5: $S^{(i)} \overset{DFT}{\longrightarrow} \mathcal{F}(S^{(i)})$
▷ DFT... Discrete Fourier transform

6: $\mathcal{F}(S^{(i)}) \longrightarrow \widehat{\mathcal{F}}(S^{(i)})$: replace Fourier amplitudes by A_k
▷ $k = -\lfloor(\frac{n-1}{2})\rfloor, \ldots, \lfloor(\frac{n}{2})\rfloor$
▷ A_k... Fourier amplitudes of X

7: $\widehat{\mathcal{F}}(S^{(i)}) \overset{IDFT}{\longrightarrow} \tilde{S}^{(i)}$
▷ IDFT... Inverse discrete Fourier transform

8: $\tilde{S}^{(i)} \longrightarrow S^{(i+1)}$: rescale $S^{(i)}$ according to $\tilde{X}$
▷ match indices according to the ascending order of $\tilde{X}$

9: $i \leftarrow i + 1$

10: **end while**

11: **return**

The replacement of the Fourier amplitudes by A_k can be achieved by the sequence `A_k.*(fft(X)./abs(fft(X)))`. All other MATLAB® commands that were used for the IAAFT algorithm are already mentioned in the previous descriptions.

A.4 Applied Simulated Annealing

Algorithm A.4: SA

Insert data $X \in \mathbb{R}^{(1\times n)}$

$X \longrightarrow S^{(0)}$: random permutation of X
▷ Initializing

Calculate $E(S^{(0)})$ and $T^{(0)}$
▷ Initial energy and temperature

repeat

  $S^{(i)} \longrightarrow \tilde{S}^{(i)}$: $s_j^{(i)} \leftrightarrow s_k^{(i)}$
  ▷ Switch indices $j, k \in 1, \ldots, n$, $j \neq k$

  Calculate $E(\tilde{S}^{(i)})$
  ▷ Energy of the surrogate candidate

  if $E(\tilde{S}^{(i)}) \leq E(S^{(i)})$ **then**

    $S^{(i+1)} \leftarrow \tilde{S}^{(i)}$ ▷ Configuration accepted

  else

    if $e^{(-\frac{E(\tilde{S}^{(i)})-E(S^{(i)})}{T^{(i)}})} > r \in [0,1)$ **then**

      $S^{(i+1)} \leftarrow \tilde{S}^{(i)}$ ▷ Configuration accepted

    else

      $S^{(i+1)} \leftarrow S^{(i)}$ ▷ Configuration rejected

    end if

  end if

  if cooling condition fulfilled **then**

    $T^{(i+1)} \leftarrow \alpha T^{(i)}$ ▷ Decreasing the temperature

  else

    $T^{(i+1)} \leftarrow T^{(i)}$

  end if

  $i \leftarrow i + 1$

until break condition

return

The estimated auto–covariance is computed by `xcov(X)` with MATLAB®. In order to increase efficiency of the code, only elements of `xcov` that are affected by a change of indices could be calculated as well.

A.5 Discrete Wavelet Transform Algorithm

Algorithm A.5: DWT

1: Insert data $X \in \mathbb{R}^{(1\times n)}$

2: $X \overset{DWT}{\longrightarrow} \mathcal{W}(X)$
▷ DWT... Discrete wavelet transform

3: $\mathcal{W}(X) \longrightarrow \tilde{\mathcal{W}}(X)$: manipulate $d_{j,\cdot}$
▷ $d_{j,\cdot}$... detail coefficients at scale j

4: $\tilde{\mathcal{W}}(X) \overset{IDWT}{\longrightarrow} S$
▷ IDWT... Inverse discrete wavelet transform

5: **return**

Several ways to compute the (inverse) discrete wavelet transform are implemented in MATLAB®. For the purpose of this thesis, the functions `dwt` and `idwt` from the Wavelet Toolbox™ were used. The `dwt` command performs a wavelet decomposition with respect to a particular wavelet, e.g. `'db6'` for Daubechies 6 tap wavelet.

A.6 (Pinned) Wavelet IAAFT Algorithm

Algorithm A.6: (P)WIAAFT

1: Insert data $X \in \mathbb{R}^{(1\times n)}$

2: $X \overset{MODWT}{\longrightarrow} \mathcal{W}(X)$
▷ MODWT... Maximal overlap discrete wavelet transform

3: $\mathcal{W}(X) \overset{IAAFT}{\longrightarrow} \tilde{\mathcal{W}}(X)$
▷ Create IAAFT–surrogate for $d_{j,\cdot}$ at each scale j
▷ $d_{j,\cdot}$... detail coefficients at scale j

4: $\tilde{\mathcal{W}}(X) \overset{IMODWT}{\longrightarrow} \tilde{S}$
▷ IDWT... Inverse maximal overlap discrete wavelet transform

5: $X \longrightarrow S$: rescale X according to $\tilde{S}$
▷ match indices according to the ascending order of $\tilde{S}$

6: **return**

A.7 Bivariate Fourier Transform Algorithm

Algorithm A.7: BFT

1: Insert data $X \in \mathbb{R}^{(2\times n)}$
▷ if not $X(:,1) \approx X(:,n)$, windowing should be applied

2: $X \overset{DFT}{\longrightarrow} \mathcal{F}(X)$
▷ DFT... Discrete Fourier transform on each row

3: $\mathcal{F}(X) \longrightarrow \tilde{\mathcal{F}}(X)$: replace $\phi_(\ :,k)$ by the same random numbers in $[0, 2\pi)$
▷ $k = -\lfloor(\frac{n-1}{2})\rfloor, \ldots, \lfloor(\frac{n}{2})\rfloor$
▷ $\phi_(\ :,k)$... Fourier phases

4: $\tilde{\mathcal{F}}(X) \overset{IDFT}{\longrightarrow} S$
▷ IDFT... Inverse discrete Fourier transform on each row

5: **return**

A.8 Bivariate Iterative AAFT Algorithm

Algorithm A.8: BIAAFT

Insert data $X \in \mathbb{R}^{(2\times n)}$
for $m = 1$ To 2 **do**
    $X(m,:) \longrightarrow \tilde{X}(m,:)$: sort $X(m,:)$ in ascending order
    $X(m,:) \longrightarrow S^{(0)}(m,:)$: random permutation of $X(m,:)$
▷ Initializing
    while $S^{(i+1)}(m,:) \neq S^{(i)}(m,:)$ **do**
        $S^{(i)}(m,:) \overset{DFT}{\longrightarrow} \mathcal{F}(S^{(i)}(m,:))$
▷ DFT... Discrete Fourier transform
        $\mathcal{F}(S^{(i)}(m,:)) \longrightarrow \widehat{\mathcal{F}}(S^{(i)}(m,:))$: replace Fourier amplitudes by $A(m,k)$
▷ $k = -\lfloor(\frac{n-1}{2})\rfloor, \ldots, \lfloor(\frac{n}{2})\rfloor$
▷ $A(m,k)$... Fourier amplitudes of $X(m,:)$ that fulfill equation 5.12
        $\widehat{\mathcal{F}}(S^{(i)}(m,:)) \overset{IDFT}{\longrightarrow} \tilde{S}^{(i)}(m,:)$
▷ IDFT... Inverse discrete Fourier transform
        $\tilde{S}^{(i)}(m,:) \longrightarrow S^{(i+1)}(m,:)$: rescale $S^{(i)}(m,:)$ according to $\tilde{X}(m,:)$
▷ match indices according to the ascending order of $\tilde{X}(m,:)$
        $i \leftarrow i + 1$
    end while
end for
return

A.9 Bivariate Applied Simulated Annealing

Algorithm A.9: BSA

Insert data $X \in \mathbb{R}^{(2\times n)}$
$X \longrightarrow S^{(0)}$: random permutation of columns of X
▷ Initializing
Calculate $E(S^{(0)})$ and $T^{(0)}$
▷ Initial energy and temperature
repeat
    $S^{(i)} \longrightarrow \tilde{S}^{(i)}$: $s_j^{(i)} \leftrightarrow s_k^{(i)}$
    ▷ Switch indices $j, k \in 1, \ldots, n$, $j \neq k$
    Calculate $E(\tilde{S}^{(i)})$
    ▷ Energy of the surrogate candidate
    if $E(\tilde{S}^{(i)}) \leq E(S^{(i)})$ **then**
        $S^{(i+1)} \leftarrow \tilde{S}^{(i)}$ ▷ Configuration accepted
    else
        if $e^{(-\frac{E(\tilde{S}^{(i)})-E(S^{(i)})}{T^{(i)}})} > r \in [0,1)$ **then**
            $S^{(i+1)} \leftarrow \tilde{S}^{(i)}$ ▷ Configuration accepted
        else
            $S^{(i+1)} \leftarrow S^{(i)}$ ▷ Configuration rejected
        end if
    end if
    if cooling condition fulfilled **then**
        $T^{(i+1)} \leftarrow \alpha T^{(i)}$ ▷ Decreasing the temperature
    else
        $T^{(i+1)} \leftarrow T^{(i)}$
    end if
    $i \leftarrow i + 1$
until break condition
return

In the bivariate case, the cross–covariance has to be computed as well. Again, this is done by the `xcov` command, but with both rows as input: `xcov(X(1,:),X(2,:))`.

A.10 Bivariate Discrete Wavelet Transform Algorithm

Algorithm A.10: BDWT

1: Insert data $X \in \mathbb{R}^{(2\times n)}$

2: $X \overset{DWT}{\longrightarrow} \mathcal{W}(X)$
▷ DWT... Discrete wavelet transform on each row

3: $\mathcal{W}(X) \longrightarrow \tilde{\mathcal{W}}(X)$: manipulate $d_{j,\cdot}$
▷ $d_{j,\cdot}$... detail coefficients at scale j for each row
▷ Identical manipulation of $d_{j,\cdot}(1,:)$ and $d_{j,\cdot}(2,:)$

4: $\tilde{\mathcal{W}}(X) \overset{IDWT}{\longrightarrow} S$
▷ IDWT... Inverse discrete wavelet transform on each row

5: **return**

Bibliography

[1] A.Schmitz *Erkennung von Nichtlinearitäten und wechselseitigen Abhängigkeiten in Zeitreihen*, NIC–Serie, Band 6 (2001)

[2] S.Kirkpatrick, C.D.Gelatt Jr., M.P.Vecchi *Optimization by Simulated Annealing*, Science Vol. 220:p.671 (1983)

[3] N.Metropolis, A.W.Rosenbluth, M.N.Rosenbluth, A.H.Teller, E.Teller *Equation of State Calculations by fast Computing Machines*, Journ. of chem. Phys. Vol.21, Nr.6:p.1087 (1953)

[4] R.V.V.Vidal (Ed.) *Applied Simulated Annealing*, Lect. Notes in Economics and Math. Systems Vol.396 (1993)

[5] C.W.J.Granger, T.Teräsvirta *Modelling Nonlinear Economic Relationships*, Oxford University Press (1993)

[6] R.Luukkonen, P.Saikkonen, T.Teräsvirta *Testing Linearity in Univariate Time Series Models*, Scand. J. Statist. 15:p.161–175 (1988)

[7] M.D.Huang, F.Romeo, A.Sangiovanni-Vincentelli *An Efficient General Cooling Schedule for Simulated Annealing*, IEEE International Conference on Computer Aided Design p.381–384 (1989)

[8] G.Bhanot *The metropolis algorithm*, Rept. Prog. Phys 51:p.429 (1988)

[9] J.Theiler, S.Eubank, A.Longtin, B.Galdrikian, J.D.Farmer *Testing for nonlinearity in time series: the method of surrogate data*, Physica D 58:p.77–94 (1992)

[10] T.Schreiber, A Schmitz *Improved surrogate data for nonlinearity tests*, Phys. Rev. Letter. 77, nr.4:p.635–638 (1996)

[11] M.Deistler, W.Scherrer *The Prague Lectures, Econometrics II*, p.16–18 (1994)

[12] B.Efron *Bootstrap Methods: Another Look at the Jackknife*, Annals of Statistics Vol.7:p.1–26 (1979)

[13] J.Fox *Bootstrapping Regression Models*, (2002)

[14] F.H.C.Marriott *Barnard's Monte Carlo tests: How many simulations?*, Applied Statistics 28:p.75–77 (1978)

[15] J.Theiler, D.Prichard *Constrained-realization Monte-Carlo method for hypothesis testing*, Physica D Vol.94, Issue 4:p.94–221 (1996)

[16] W.Lu, N.Vaswani *The Wiener-Khinchin Theorem for Non-wide Sense stationary Random Processes*, Department of Electrical and Computer Engineering, Iowa State University, Ames, IA (2009)

[17] I.Kuščer, I.Vidav, C.D.Boley *Role of the Wiener-Khinchin theorem in statistical mechanics*, Zeitschrift für Angewandte Mathematik und Physik Vol.29, Nr.1:p.23–34 (1978)

[18] D.Prichard *The correlation dimension of differenced data*, Phys. Lett. A 191:p.245–250 (1994)

[19] T.Schreiber *Constrained randomization of time series data*, Phys. Rev. Lett. 80:p.2105–2108 (1998)

[20] A.Schmitz, T.Schreiber *Surrogate data for non-stationary signals*, Chaos in Brain? Interdisc. Workshop in Bonn, Germany:p.222–225 (1999)

[21] J.D.Scargle *Statistical aspects of of spectral analysis of unevenly spaced data*, Astrophysical Journal 263:p.835–853 (1982)

[22] N.R.Lomb *Least-squares frequency analysis of unequally spaced data*, Astrophys. and Space Sci. 39:p.447–462 (1976)

[23] V.V.Vityazev *Time Series Analysis of Unequally Spaced Data: Intercomparison Between Estimators of the Power Spectrum*, ASP Conference Series Vol. 125:p.166–169 (1997)

[24] T.Schreiber, A.Schmitz *Testing for nonlinearity in unevenly sampled time series*, Phys. Rev. E. 59(4):p.4044–4047 (1999)

[25] A.Schmitz, T.Schreiber *Surrogate time series*, Physica D 142:p.346–382 (2000)

[26] J.Theiler, P.S.Linsay, D.M.Rubin *Detecting nonlinearity in data with long coherence times*, Time series prediction: Forecasting the future and understanding the past:p.429–455 (1993)

[27] B.Efron *The Jackknife, the Bootstrap and Other Resampling Plans*, SIAM, Philadelphia PA (1982)

[28] M.M.Atiqullah *An Efficient Simple Cooling Schedule for Simulated Annealing*, ICCSA 2004, LNCS 3045:p.396–404 (2004)

[29] D.Bertsimas, J.Tsitsiklis *Simulated Annealing*, Statistical Science Vol.8, No.1:p.10–15 (1993)

[30] Z.Król *An adaptive stochastic optimization method for medical registration problem*, Journ. of Medical Inform. and Techn. Vol.7:p.79–88 (2004)

[31] H.–S.Cho, C.–H.Paik, H.–M.Yoon, H.–G.Kim *A robust design of simulated annealing approach for mixed-model sequencing*, Computers and Industrial Engineering Vol.48, No.4:p.753–764 (2005)

[32] N.Azizi, S.Zolfaghari *Adaptive temperature control for simulated annealing: a comparative study*, Computers and Operations Research 31:p.2439–2451 (2004)

[33] P.J.M.v.Laarhoven, E.H.L.Aarts *Simulated Annealing: Theory and Applications*, D.Reidel, Dordrecht:p.198 (1987)

[34] R.H.J.M.Otten, L.P.P.P.v.Ginneken *The Annealing Algorithm*, The Springer International Series in Engineering and Computer Science Vol.72:p.224 (1989)

[35] D.Kugiumtzis *Lack of Robustness of the Surrogate Data Test*, Physica D (1999)

[36] T.Schreiber, A.Schmitz *On the discrimination power of measures for nonlinearity in a time series*, Phys. Rev. E. 59:p.5443 (1997)

[37] C. Diks, J.C.v. Houwelingen, F.Takens, J.DeGoede *Reversibility as a criterion for discriminating time series*, Physical Letters A:p.201–221 (1995)

[38] M.Small, K.Judd *Detecting nonlinearity in experimental data*, International Journal of Bifurcation and Chaos Vol.8, p.1231–1244 (1997)

[39] T.Panagiotidis *Testing the assumption of Linearity*, Economics Bulletin Vol.3, No.29:p.1–9 (2002)

[40] D.Kugiumtzis *Test your surrogate data before you test for nonlinearity*, Phys. Rev. E Vol.60, No.3:p.2808–2816 (1999)

[41] A.Wolf, J.Swift, H.Swinney, J.Vastano *Determining Lyapunov exponents from a time series*, Physica D, 16:p.285–317 (1985)

[42] P.Hall, S.R.Wilson *Two guidelines for bootstrap hypothesis testing*, Biometrics Vol.47, Issue 2:p.757–762 (1991)

[43] J.Theiler, D.Prichard *Generating surrogate data for time series with several simultaneously measured variables*, Phys. Rev. Lett. 73 No.7:p.951–954 (1994)

[44] R.G.Andrzejak, A.Kraskov, H.Stögbauer, F.Mormann, T.Kreuz *Bivariate surrogate techniques: Necessity, strengths, and caveats*, Phys. Rev. E 066202:p.68 (2003)

[45] C.E.Shannon *A Mathematical Theory of Communication*, The Bell System Technical Journal, Vol.27:p.379–423 (1948)

[46] S.Kullback, R.A.Leibler *On information and sufficiency*, Annals of Mathematical Statistics 22, No.1:p.79–86 (1951)

[47] T.Schreiber *Measuring information transfer*, Phys. Rev. Lett. Vol.85, No.2:p.461–464 (2000)

[48] J.Arnhold, P.Grassberger, K.Lehnertz, C.E.Elger *A robust method for detecting interdependences: application to intracranially recorded EEG*, Physica D 134:p.419–430 (1999)

[49] G.E.P.Box, G.M.Jenkins *Time Series Analysis: Forecasting and Control*, San Francisco: Holden–Day (1970)

[50] J.D.Hamilton *Time Series Analysis*, Princeton University Press (1994)

[51] C.W.J.Granger, P.Newbold *Forecasting Economic Time Series*, (1986)

[52] R.F.Engle *Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation*, Econometrica Vol. 50, No. 4.:p.987–1007 (1982)

[53] C.W.J.Granger, A.P.Andersen *An Introduction to Bilinear Time Series Models*, Vandenhoeck and Ruprecht (1978)

[54] W.H.Press, S.T.Teukolsky, W.T.Vetterling, B.P.Flannery *Numerical Recipes in C: the art of scientific computing*, Cambridge Univ. Press (1992)

[55] J.F.Moon *Rudolf Diesel and the Diesel Engine*, London: Priory Press (1974)

[56] D.E.Thomas *Diesel: Technology and Society in Industrial Germany*, University of Alabama Press (1987)

[57] K.Steenland, D.T.Silverman, D.Zaebst *Exposure to diesel exhaust in the trucking industry and possible relationships with lung cancer*, American Journal of Industrial Medicine 21:p.887–890 (1992)

[58] I.Bruske-Holhfield, M.Mohner, W.Ahrens, et al. *Lung cancer risk in male workers occupationally exposed to diesel motor emissions in Germany*, American Journal of Industrial Medicine 36:p.405–414 (1999)

[59] J.Timmer *Power of surrogate data testing with respect to nonstationarity*, Physical Review E Vol. 58, No. 4:p.5153–5156 (1998)

[60] D.T.Kaplan *Nonlinearity and nonstationarity: the use of surrogate data in interpreting fluctuations*, Frontiers of Blood Pressure and Heart Rate Analysis (1997)

[61] P.Borgnat, P.Flandrin *Stationarization via surrogates*, Journal of Statistical Mechanics: Theory and Experiment, Issue 01:p.01001 (2009)

[62] T.Schreiber *Interdisciplinary application of nonlinear time series methods*, Phys. Rep. 308:p.1–64 (1999)

[63] L.Faes, Z.He, K.H.Chon, G.Nollo *Time–Varying Surrogate Data to Assess Nonlinearity in Nonstationary Time Series: Application to Heart Rate Variability*, Biomedical Engineering, IEEE Transactions Vol. 56, Issue 3:p.685–695 (2009)

[64] M.Small, D.Yu, R.G.Harrison *A surrogate test for pseudoperiodic time series data*, Phys. Rev. Lett. Vol.87, No.18 (2001)

[65] M.Small, C.K.Tse *Applying the method of surrogate data to cyclic time series*, Physica D 164:p.187–201 (2002)

[66] J.Theiler *On the evidence for low–dimensional chaos in an epileptic electroencephalogram*, Phys. Lett. A 196:p.335–341 (1995)

[67] C.J.Keylock *Constrained surrogate time series with preservation of the mean and variance structure*, Phys. Rev. E 73:036707 (2006)

[68] C.J. Keylock *A wavelet–based method for surrogate data generation* Physica D 225:p.219–228 (2007)

[69] M.Breakspear, M.Brammer, P.A.Robinson *Construction of multivariate surrogate sets from nonlinear data using the wavelet transform*, Physica D 182:p.1–22 (2003)

[70] I.Daubechies *The wavelet transform, time–frequency localization and signal analysis*, IEEE Trans. Inf. Theory 36, 5:p.961–1005 (1990)

[71] D.B.Percival, A.T.Walden *Wavelet Methods for Times Series Analysis*, Cambridge University Press (2000)