Dimension reduction of global models using principal component analysis

Bardoel, Stef L.

Award date:
2017

Dimension reduction of global models using principal component analysis

Master Thesis
Department of Applied Physics
Elementary Processes in Gas Discharges (EPG)

Author:
S.L. Bardoel

Supervisor:
ir. P.M.J. Koelman

Graduation committee:
dr.ir. J. van Dijk
dr.ir. S. Nijdam
dr. M. Duran Matute
dr. A. Silov
L. Vialetto, MSc. (Differ/advisor)

Eindhoven, December 2017

Abstract

Plasma-assisted conversion of CO_2 into CO provides a promising first step of the synthesis of solar fuels. Numerical simulations of a non-equilibrium CO_2 plasma can be a useful way to study the conversion. The vibrational modes of CO_2 provide an efficient path to dissociation, hence an accurate description of the transfer of vibrational energy is needed. The large number of species and reactions present in the CO_2 plasma poses a serious challenge for the simulation of the reaction kinetics due to the high computational load. Chemical reduction techniques are able to reduce the amount of species for which balance equations need to be solved. As an initial reduction study we will limit ourselves to global models, for which PLASIMO is used. Principal Component Analysis (PCA) is a chemical reduction technique that transforms correlated species densities into uncorrelated variables called the Principal Components (PCs). Reduction is achieved by solving balance equations for only the most relevant PCs instead of for all the species densities. A major challenge in the application of PCA is the quantification of the errors that are introduced by the reduction. In order to evaluate the accuracy of the PCA model, we compare results from the PCA global model with results from the full global model and a global model based on the Intrinsic Low-Dimensional Manifold method.

Contents

Chapter 1

Introduction

1.1 Motivation

Energy is one of the basic needs of people across the world, together with food, water and shelter [1]. Ever since the Industrial Revolution the energy demand has increased significantly. The availability of the main energy source at that time, wood, was limited, and therefore other energy sources, such as coal and peat/turf were used. Also other types of fossil fuels were introduced, like oil and natural gas, that helped to meet the energy demands of the modern age. Unfortunately, there are several disadvantages of the use of fossil fuels. First of all, fossil fuels are a non-renewable energy resource, because it cannot be replaced in a natural way at a rate similar to its consumption. At the current rate of oil and gas production, the fossil fuel reserves are depleting rapidly [2]. Secondly, the abundant use of fossil fuels leads to several environmental issues. This includes the production of CO_2, which is one of the greenhouse gases that are involved in global warming. Also, other harmful chemicals are released when fossil fuels are burned, such as NO_2, SO_2 and CO. These chemicals may cause serious health problems, in particular for the respiratory system. Thirdly, many European countries depend on foreign countries for a great part of the importation of fossil fuels. For these reasons, alternative energy sources have been developed in the form of renewable energy sources. Examples include wind energy, solar energy (solar cells), nuclear energy, hydropower, geothermal energy and bio energy. A drawback of many renewable energy sources is that the energy generation of most of them fluctuates due to diurnal and seasonal variations. For example, solar cells produce much more electricity during summer than during winter, and also more during the day than at night. The peaks of energy production often do not match the peaks of energy consumption during winter and early evenings. An interesting solution is to store excesses energy that could be consumed when energy production is much lower than energy consumption. One way to store energy is by production of solar fuels, which has the advantage that the modern society is adapted to the use of fuels. Solar fuels are synthesized from CO_2 in an inverse combustion reaction using solar energy. A major advantage of the use of CO_2 as an integral part of industrial processes is that CO_2 emissions are mitigated [3]. The first

Figure 1.1: The northern light.

step of the production of solar fuels is the dissociation of CO_2 into CO [4, 5], which can be done by using a plasma. The produced CO is then converted into fuels by a water gas shift reaction and methanation [6]. Section 1.3 focuses on plasma-assisted CO_2 conversion, but first we will focus on what a plasma is.

1.2 Plasma

Matter can be categorized into four fundamental states of matter. The first is the solid state, which is characterized by closely packed particles. Ions, atoms or molecules cannot move freely due to strong intermolecular forces and are arranged either in structured lattices or irregularly. The second state, liquid, is a fluid that often is almost incompressible. Intermolecular forces are still present, but it is possible for molecules to move. Gas, which is a compressible fluid, is the third state. Intermolecular interactions are small compared to the kinetic energy and distances between molecules are much larger than the size of the molecules. The fourth is the plasma state, which can be created from a gas by heating or by applying an electric field. This causes electrons to leave the atoms, so that ions and free electrons are created. Plasmas are often only partially ionized, which means that neutrals also exist in plasmas. The plasma state does not have a prominent presence on earth, but is sometimes created by sparks, lightning, or flames. The northern light, see figure 1.1, is a beautiful plasma that is created in high parts of the atmosphere of the earth. Space is more known for the existence of the plasma state of matter, such as the sun.

Collisional processes between electrons, ions and neutrals are very important in plasmas [7]. Energy transfer in elastic collisions between particles is proportional to the ratio of their individual masses. Electrons have a much lower mass than ions and neutrals, so they transfer little energy to ions and neutrals. Ions and neutrals usually have similar masses,

so they exchange energy efficiently. The uneven distribution of energy leads to a non-equilibrium plasma, where the electron temperature is much higher than the temperature of the ions and neutrals. The electron temperature can be several thousands of Kelvins, whereas the gas temperature can be close to room temperature. This type of plasma is often referred to as non-thermal plasma. When the pressure is increased, the frequency of collisions between electrons, ions and neutrals increases, so for high enough pressures the plasma will reach equilibrium, with an electron temperature that is approximately equal to the gas temperature.

1.3 Plasma-assisted CO_2 conversion

The use of non-thermal plasmas for CO_2 dissociation has been suggested to be an attractive alternative to the conventional thermal and catalytic methods. This is due to the low energy cost, its non-equilibrium properties and unique ability to initiate chemical reactions despite the low gas temperature [2, 3]. The electrons have a much higher temperature and are able to break chemical bonds of many inert molecules, such as CO_2 and produce reactive species (free radicals, excited atoms, ions an molecules) [8]. Non-thermal plasmas are therefore able to overcome the thermodynamic barriers of CO_2 dissociation at atmospheric pressure and low temperatures [2].

The energy that is required to dissociate a CO_2 molecule in the ground state is 5.5 eV. Fridman showed in [5] that the theoretical maximum energy efficiency for the dissociation is about 45 % for thermal processes. Other works, however, have shown that higher energy efficiencies can be achieved by selective excitation of the vibrational modes of CO_2, with a maximum of 80 % [4, 5, 9, 10]. The CO_2 molecule has three vibrational modes, of which the asymmetric mode provides the most efficient pathway to dissociation [5].

Several models of non-thermal CO_2 plasmas have been developed to study the processes that are involved in CO_2 dissociation. Aerts [10] *et al.* presented a CO_2 chemistry that contains 25 species and 205 reactions. Kozak *et al.* [9] extended this chemistry by adding, among others, the asymmetric vibrational modes of CO_2. This chemistry contains 72 species and thousands of reactions. All these models are Global Models, which does not resolve spatial dimensions, but only changes of species densities as a function of time. Global Models are often referred to as volume-averaged models. Ideally, in order to model a complete CO_2 reactor, spatial derivatives are included. The computational load of 2D/3D models that contain many species and thousands of reactions is extremely large and makes it impossible to run these models within a reasonable time. Clearly, there is a need for reduction techniques that reduce the chemistry of these models to an appropriate size [11].

1.4 Reduction techniques

Peerenboom *et al.* [11] categorize two types of dimension reduction for plasma models: mechanism reduction and reparametrization of the chemical state space. For mechanism

reduction, a certain mechanism that is present in the chemistry is reduced. There are multiple techniques that are used in mechanism reduction. An example of a reduced mechanism is the grouping of energy levels in state-to-state kinetic models [12, 13]. Here, the large number of rotational-vibrational energy levels are lumped into a smaller number of bins, thereby reducing the amount of species and reactions that need to be accounted for. The second reduction method that uses mechanism reduction is the Quasi Steady State Assumption (QSSA). QSSA was developed by Bates, Kingston and McWhirter [14, 15] in 1962 and called QSSA for the first time in [16]. QSSA takes advantage of the fact that many intermediate species are reactive and have small concentrations. The time scales of the reactions of these species are fast, and the QSSA considers these timescales as infinitely fast. Therefore, these species are in equilibrium. A more advanced version of QSSA is Intrinsic Low-Dimensional Manifold (ILDM) [17, 18], which automatically decouples the fast processes from the slow processes. A short explanation of ILDM is given in section 3.6. Pathway analysis, developed by Lehmann [19, 20] as an analysis tool, can also be used for mechanism reduction.

The strategy of reparametrization of the chemical state space is relatively new in the plasma modelling community, but is extensively used in combustion modelling. It takes advantage of the fact that the composition of a mixture lies on a low-dimensional manifold. It is possible due to this low-dimensional manifold to describe the composition with a limited amount of parameters. The flamelet generated manifolds is a reduction method that has been successfully used in combustion modelling [21]. Another method, which has been recently introduced by the combustion community, is Principal Component Analysis (PCA). PCA is a statistical method that transforms a set of correlated variables into a set of uncorrelated variables, called Principal Components (PCs). This transformation has properties that make PCA a very useful reduction method. Peerenboom *et al.* introduced PCA as a reduction method in the plasma community in [11]. In this work, they successfully reduced a 0D state-to-state kinetic model of CO_2 such that only two variables were needed to describe the species in the plasma. This thesis focuses on the reduction of Global Models using PCA.

1.5 Project goals

PCA has been used for some time in combustion modelling and was only very recently introduced in plasma modelling as a reduction method by Peerenboom *et al.* [11]. They showed that results obtained from the PCA global model accurately reproduced results from the full model using only two parameters. However, it is not clear how accurate the PCA reduction method is compared to other reduction methods, such as ILDM. Hence the following research question is defined:

- How accurate is the PCA global model compared to a full global model and an ILDM global model?

In order to answer this research question, the accuracy of the reduced models is deter-

mined for global models of molecular argon, argon with 78 levels and a microwave CO_2 model. According to Peerenboom *et al.* the success of PCA depends on the uniqueness of the manifold. It is unclear how should be dealt with a manifold that is not unique without using more than two parameters. Besides, Peerenboom *et al.* proposed the use of nonlinear regression to convert the two parameters to the species densities. The following subquestions are defined:

- How can we deal with a manifold that is not unique?
- How accurate is nonlinear regression for the reconstruction?

1.6 Outline of this thesis

This thesis is structured in the following way:

- Chapter 2 discusses the principles of PCA. After introducing PCA, a derivation is given of the Principal Components, including a short discussion of the PCA nomenclature. Also, two PCA algorithms are discussed and compared. This chapter finishes with the importance of scaling and a short discussion about some properties of PCA.
- Chapter 3 contains a description of the reduction of a plasma global model using PCA. Some important aspects of the PCA global model are highlighted. Also, a short description of another reduction method is given: ILDM.
- Chapter 4 shows the verification of the developed PCA global model with results from the work of Peerenboom *et al.*
- Chapter 5 gives a comparison of the accuracy of the reduced PCA and ILDM global models with respect to the full simulation. This is done for a molecular argon model, an argon model with 78 levels and a CO_2 microwave model.
- Chapter 6 considers the use of nonlinear regression in PCA. This is an alternative for the lookup tables that were used by Peerenboom *et al.*
- Chapter 7 summarizes the main conclusions of this work.

Chapter 2

Principal Component Analysis as a statistical tool

2.1 Introduction

Principal Component Analysis is one of the oldest and best known techniques of multivariate analysis [22]. PCA is able to extract useful information from complex and confusing data sets that are often present in studies. Data sets often contain many variables, which makes the analysis of the data much harder. With minimum effort, PCA provides a roadmap for the simplification of such complex data sets. After this simplification, the sometimes hidden and underlying structures of the data can be revealed. The Principal Component Analysis of a data set can be done for many objectives, such as simplification, modelling, outlier detection, variable selection, classification, prediction, unmixing and, in this work, dimensionality reduction [23]. PCA is used in many different research areas. In agriculture, MacLeod and Moller [24] used PCA to study the intensification and diversification of New Zealand agriculture since 1960. In biology, Liao *et al.* [25] pesent a modification of PCA, Network Component Analysis, that takes into account hidden regulatory signals. In chemistry, Chiang *et al.* [26] used PCA for detecting and diagonising faults in chemical processes. In climatology, Overland and Preisendorfer [27] applied PCA to study cyclone climatology. In demography, Vyas and Kumaranayake [28] used PCA to find PCA-based socio-economic status indices. In ecology, Borcard and Legendre [29] apply PCA on ecological data for outlier detection based on the Mahalanobis distance. In food research, He *et al.* [30] discriminated varieties of tea using near infrared spectroscopy, PCA and the method of back propagation. In genetics, Novembre and Stephens [31] interpret Principal Component Analyses of spatial population genetic variation. In geology, Kwarteng and Chavez [32] extracted spectral contrast in Landsat thematic mapper image data using selective PCA. In meteorology, Horel [33] used PCA to analyse the interannual variability of the northern hemisphere at hights where the pressure is 500 mbar. In oceanography, Berg *et al.* [34] studied atmospheric trace element deposition by applying PCA on data from moss samples. In psychology, Raskin and Terry [35] performed a PCA

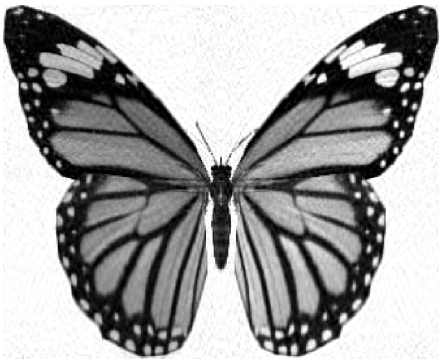
of the narcissistic personality inventory in order to examine its validity. In quality control, Jackson and Mudholkar [36] studied the accuracy of PCA for quality control applications.

Another field that often uses PCA is epidemiology. Epidemiology studies and analyzes patterns, causes and effects of health and disease conditions in populations. The probability of diseases may depend on many variables, which poses a serious challenge for identifying risk factors for these diseases. An example of a study in the field of epidemiology that uses PCA is the study performed by Navarro Silvera et al. [37]. In their work a pattern analysis of dietary and lifestyle factors in relation to risk of esophageal and gastric cancers was performed. In their analysis they considered 28 variables, consisting of 22 dietary factors and 6 lifestyle factors. PCA was used to reduce the dimensionality of the data to 6 new variables, the Principal Components (PCs). In this study, these 6 PCs are interpreted as 6 dietary or lifestyle patterns that could be linked more easily to the types of cancer. These patterns were named the Meat/Nitrite pattern, the Fruit/Vegetable pattern, the Smoking/Alcohol pattern, the Legume/Meat Alternate pattern, the GERD/BMI pattern (gastroesophageal reflux disease) and the Fish/Vitamin C pattern. It was much easier for the authors to identify potential relationships between the patterns and the esophageal and gastric cancers based on these 6 patterns after using PCA. The analyses suggested that meat/nitrite intake is associated with elevated risk of each cancer considered in this study, whereas fruit/vegetable intake reduces the risk of these cancers. GERD/obesity and smoking/alcohol were confirmed as risk factors for only one specific form of cancer.

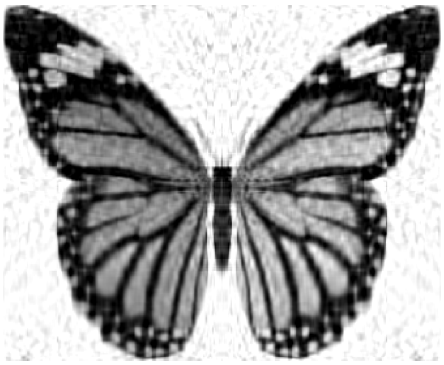
Another application of PCA is data compression. In our technological world, storage of large amounts of data has become a major challenge. Therefore, several data compression methods have been developed, and some of them use PCA, for example image compression. Images consist of a list of numbers that indicate the intensity and color of its pixels. Neighboring rows and columns of this list usually show some similarities and therefore these images have a large amount of redundancy. Therefore, the image can be described by a limited amount of Principal Components. As an example, consider the image of a butterfly, shown in figure 2.1. This image is a gray scale image described by a matrix of 272 by 335 integers ranging from 0 to 255, where each number indicates the brightness of a pixel. We are using this list of numbers as a data set on which PCA will be applied. PCA is applied on the image for four cases, where 50, 20, 10 and 5 PCs were chosen. The PCs are then converted back into the butterfly, and the resulting images in figure 2.2 are compared with the original image. The resulting image for 50 PCs shows barely any differences with the original butterfly. Some noise becomes visible in the picture of the butterfly for 20 PCs. Even the images for 10 and 5 PCs, which have a data compression rate of 33.5 and 67, still clearly resemble a butterfly.

Despite the simplicity of the method and its widespread use, it is not always clear how and why PCA works. Sometimes PCA is called a 'black box': something is put in the box and something comes out again, without having an idea what is going on inside this box. We want to get rid of this idea and unravel the mysteries that happen in the black box called PCA. Therefore, this entire chapter will be dedicated to explaining the principles of PCA and dispelling its magic. As a first step, we will derive the Principal Components. After that, the PCA nomenclature will be discussed. Next, two PCA algorithms are discussed:

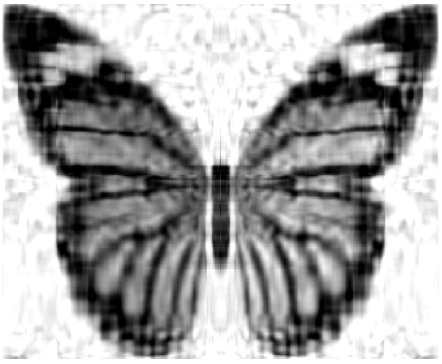
Figure 2.1: Original image of a butterfly.



(a) 50 PCs



(b) 20 PCs.



(c) 10 PCs.

(d) 5 PCs.

Figure 2.2: Compressed images by using 50, 20, 10 and 5 PCs.

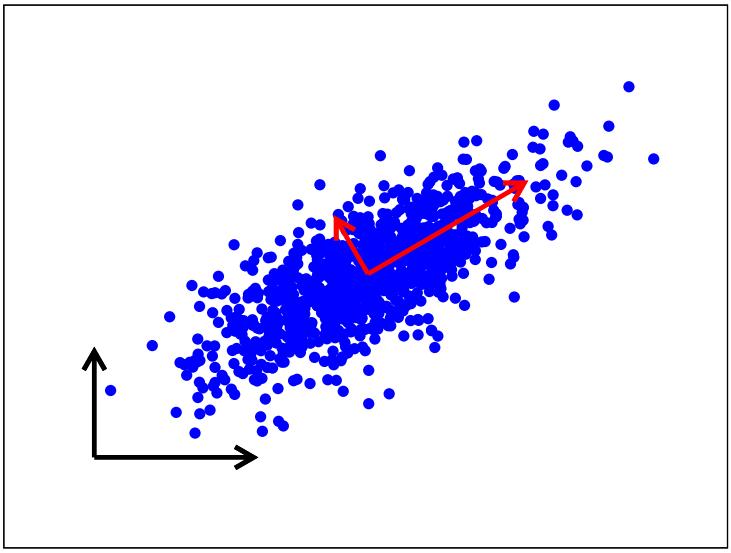


Figure 2.3: PCA applied to a data set of two variables with a Gaussian distribution. The black arrows are the axes of the original variables and the red arrows are the axes of the PCs.

the eigendecomposition of the covariance matrix and singular value decomposition of the data matrix. After that, we will look at the importance of centering and scaling. Then, the properties and assumptions of PCA are discussed. The final section contains an extension of PCA: Kernel Principal Component Analysis.

2.2 Derivation of the Principal Components

The ultimate goal of PCA is to redistribute the information of a data set, $\boldsymbol{X}$, in such a way that the first variables contain the most information, whereas the last variables contain least information. In order to achieve this goal, PCA transforms the variables of the original data set into new variables, the Principal Components (PCs). Figure 2.3 gives a graphical example of what the PCs are. The example shows a data set containing two variables that follow a Gaussian distribution. The two axes of the original variables that describe the data are given in black. The PCs represent the transformed variables, measured by the red axes. It is clear from the figure that the PC axes are much more suited for the data than the axes of the original variables.

This section will focus on this transformation: which transformation is best? In order to find the ideal transformation, we start off by imposing three requirements on the PCs:

1. *Linearity.* The PCs must be a linear combination of the original variables. The assumption of linearity is a very convenient and powerful choice. It greatly limits the amount of potential bases that can be used for the PCs.

2. *Zero-correlation.* Since one of the goals of PCA is to remove redundancy of data, the variables should have a correlation of zero.

3. *Principal.* The PCs must be sorted based the amount of information they contain. The first PC must contain as much information as possible. The second PC must contain the second highest amount of information possible. The last PC must contain least amount of information.

Let's now look at these requirements from a mathematical perspective. Our first restriction is to use a *linear transformation*, so that the PCs are a linear combination of the variables. This linear transformation is expressed by a matrix multiplication of the data $\boldsymbol{X}$ with a transition matrix:

$$\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{A}, \tag{2.1}$$

where $\boldsymbol{A}$ is the transition matrix from the original variables to the PCs and $\boldsymbol{Z}$ contains the data measured with the new basis of the PCs. This equation represents a change of basis, which has a number of interpretations:

- $\boldsymbol{A}$ is a matrix that transforms $\boldsymbol{X}$ into $\boldsymbol{Z}$.
- Geometrically, $\boldsymbol{A}$ provides a rotation and a stretch of $\boldsymbol{X}$ into $\boldsymbol{Z}$.
- The columns of $\boldsymbol{A}$, $(\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_Q)$, are a set of new basis vectors for the PCs.

The second requirement for the PCs is that they must be *uncorrelated*. The correlation of n observations of two variables $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is given by:

$$\operatorname{corr}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \frac{\operatorname{cov}(\boldsymbol{x}_1, \boldsymbol{x}_2)}{\sigma_1 \sigma_2} = \frac{(\boldsymbol{x}_1 - \mu_1)^{\mathrm{T}} (\boldsymbol{x}_2 - \mu_2)}{\sigma_1 \sigma_2 (n-1)} \tag{2.2}$$

where $\operatorname{cov}(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is the covariance between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, μ_1 and μ_2 are the mean of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, and σ_1 and σ_2 are the standard deviation of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ respectively. In this equation $(\boldsymbol{x}_1 - \mu_1)^{\mathrm{T}}$ is the transpose of $(\boldsymbol{x}_1 - \mu_1)$. Two variables have a perfectly linear relationship if the correlation equals $+1$, a negative or inverse relationship if the correlation is -1 and are linearly independent if the correlation is 0. Hence the PCs are uncorrelated if if the covariance of the PCs equals zero.

The third requirement was that the first PC must contain *most information* of the data. The second PC must be uncorrelated with the first PC and also contain the most important information of the remaining data. Finally, the last PC must be uncorrelated with all the other PCs and have minimum information. The term information is vague, and therefore it is quantified by the property of variance. This means for the first PC that its variance is to be maximized. The variance of the first PC is given by:

$$\operatorname{var}(\boldsymbol{z}_1) = \operatorname{var}(\boldsymbol{X}\boldsymbol{a}_1) = \frac{(\boldsymbol{X}\boldsymbol{a}_1)^{\mathrm{T}} \boldsymbol{X}\boldsymbol{a}_1}{n-1} = \boldsymbol{a}_1^{\mathrm{T}} \frac{\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}}{n-1} \boldsymbol{a}_1 = \boldsymbol{a}_1^{\mathrm{T}} \boldsymbol{C} \boldsymbol{a}_1. \tag{2.3}$$

Here, we see the appearance of a matrix that will become very important: the covariance matrix $\boldsymbol{C}$. This matrix combines the concepts of variance and covariance. The diagonal elements of the covariance matrix contain the variance of a variable and the off-diagonal elements are the covariance of two variables.

Now, we will derive how to calculate the Principal Components. We will start by finding the basis of the first PC $\boldsymbol{a}_1$ that maximizes the variance of the first PC $\text{var}(\boldsymbol{X}\boldsymbol{a}_1)$, given in equation (2.3). Here, a normalization constraint must be introduced, because the variance of the first PC can be maximized by simply making the elements of $\boldsymbol{a}_1$ infinitely large. The following normalization constraint is introduced in order to prevent this:

$$\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{a}_1 = 1. \tag{2.4}$$

There are also other normalization constraints that can be used, such as $\max(a_{1j}) = 1$, but these usually make the derivation of the PCs more difficult [22].

The standard procedure that is used to maximize a function subject to one or more constraint is the method of Lagrange multipliers. Consider the optimization problem of maximizing $f(x_1, \ldots, x_Q)$ subject to $g(x_1, \ldots, x_Q) = c$, with Q the amount of variables. The method uses the fact that the stationary points (extreme values and saddle points) of $f(x_1, \ldots, x_Q)$ are such that there exists a number λ, the Lagrange multiplier, such that

$$\frac{\partial f}{\partial x_i} - \lambda \frac{\partial g}{\partial x_i} = 0, \qquad i = 1, \ldots, Q, \tag{2.5}$$

at the stationary points [38]. The stationary points of f are in fact the solutions of this equation. Instead of solving equation (2.5) directly, a new function is formed first, the Lagrangian $\mathcal{L}$:

$$\mathcal{L}(\boldsymbol{x}) = f(\boldsymbol{x}) - \lambda\left(g(\boldsymbol{x}) - c\right), \tag{2.6}$$

The stationary points are found from the Lagrangian by solving:

$$\nabla_{\boldsymbol{x}}\mathcal{L} = \boldsymbol{0}, \tag{2.7}$$

with $\nabla_{\boldsymbol{x}}$ the gradient with respect to $\boldsymbol{x}$. In the derivation of the first PC, we try to maximize $\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{C}\boldsymbol{a}_1$ under the constraint $\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{a}_1 = 1$. Therefore, we define the following Lagrangian:

$$\mathcal{L}(\boldsymbol{a}_1) = \boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{C}\boldsymbol{a}_1 - \lambda_1(\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{a}_1 - 1). \tag{2.8}$$

Setting the gradient of $\mathcal{L}(\boldsymbol{a}_1)$ with respect to the elements of $\boldsymbol{a}_1$ equal to zero gives:

$$\boldsymbol{C}\boldsymbol{a}_1 - \lambda_1\boldsymbol{a}_1 = \boldsymbol{0}, \tag{2.9}$$

or in a more convenient notation:

$$(\boldsymbol{C} - \lambda_1\boldsymbol{I})\boldsymbol{a}_1 = \boldsymbol{0}, \tag{2.10}$$

where $\boldsymbol{I}$ is the identity matrix. The lagrange multiplier λ is an eigenvalue of $\boldsymbol{C}$ and the basis of the first PC $\boldsymbol{a}_1$ is an eigenvector of $\boldsymbol{C}$. We only need to decide which of the Q

eigenvectors is needed. The eigenvector of the first PC must give maximum variance, where the variance can now be simplified to

$$\text{var}(\boldsymbol{X}\boldsymbol{a}_1) = \boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{C}\boldsymbol{a}_1 = \boldsymbol{a}_1^{\mathrm{T}}\lambda_1\boldsymbol{a}_1 = \lambda_1\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{a}_1 = \lambda_1. \tag{2.11}$$

The variance of the first PC is given by the largest eigenvalue of the covariance matrix. The first PC must then be equal to the eigenvector that belongs to the largest eigenvalue of the covariance matrix.

After having found the first PC, we will continue and derive the second PC. This comes down to maximizing the variance of the second PC, where we require that the second PC is uncorrelated with the first PC. This is done by setting the covariance between the two PCs to zero:

$$\text{cov}(\boldsymbol{X}\boldsymbol{a}_1, \boldsymbol{X}\boldsymbol{a}_2) = \boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{C}\boldsymbol{a}_2 = \boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{C}\boldsymbol{a}_1 = \lambda_1\boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{a}_1 = \lambda_1\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{a}_2 = 0. \tag{2.12}$$

Thus, the normalization and uncorrelation constraints for the second PC are:

$$\begin{aligned} \boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{a}_2 &= 1, \\ \boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{a}_2 &= 0. \end{aligned} \tag{2.13}$$

We define the following Lagrangian, where we maximize the variance of the second PC under the two constraints:

$$\mathcal{L}(\boldsymbol{a}_1, \boldsymbol{a}_2) = \boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{C}\boldsymbol{a}_2 - \lambda_2(\boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{a}_2 - 1) - \mu\boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{a}_1, \tag{2.14}$$

with Lagrange multipliers λ_2 and μ. Setting the gradient of the Lagrangian with respect to $\boldsymbol{a}_2$ to zero leads to:

$$2\boldsymbol{C}\boldsymbol{a}_2 - 2\lambda_2\boldsymbol{a}_2 - \mu\boldsymbol{a}_1 = \boldsymbol{0}. \tag{2.15}$$

This equation is multiplied on the left by $\boldsymbol{a}_1^{\mathrm{T}}$, resulting in:

$$2\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{C}\boldsymbol{a}_2 - 2\lambda_2\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{a}_2 - \mu\boldsymbol{a}_1^{\mathrm{T}}\boldsymbol{a}_1 = 0. \tag{2.16}$$

The first two terms of this equation are equal to zero due to orthogonality of $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ and because $\boldsymbol{a}_1$ is normalized, this results in $\mu = 0$. Again, this gives an eigenvalue problem:

$$(\boldsymbol{C} - \lambda_2\boldsymbol{I})\boldsymbol{a}_2 = \boldsymbol{0}, \tag{2.17}$$

where λ_2 is an eigenvalue and $\boldsymbol{a}_2$ an eigenvector of $\boldsymbol{C}$. It is now relatively easy to prove that the variance of the second PC equals λ_2:

$$\text{var}(\boldsymbol{X}\boldsymbol{a}_2) = \boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{C}\boldsymbol{a}_2 = \boldsymbol{a}_2^{\mathrm{T}}\lambda_2\boldsymbol{a}_2 = \lambda_2\boldsymbol{a}_2^{\mathrm{T}}\boldsymbol{a}_2 = \lambda_2. \tag{2.18}$$

In order to find the second PC, we need to find the second highest eigenvalue (the highest was already used for the first PC) and the corresponding eigenvector. It can be proven that all other PCs are also eigenvectors of the covariance matrix, where their variances are given by the corresponding eigenvalue [22].

2.3 PCA nomenclature

One of the difficulties in the PCA literature is that there is no uniform notation of the important PCA quantities, which is also the case in other topics in multivariate analysis. Jackson [39] gives a table that contains a list of symbols that is used for PCA quantities by 34 key references in appendix D. Hardly any of the listed references use the same symbols. For example, eight different symbols are used for the eigenvalues of the covariance matrix: l, var($\boldsymbol{C}$), λ, c, d, k, $\boldsymbol{L}$ and VXF. Also, the nomenclature is very inconsistent in the PCA literature in two ways. First, the name of the method is named differently in some fields, such as the Hotelling transform in multivariate quality control [40], the discrete Karhunen-Loève transform (KLT) in signal processing [41], proper orthogonal decomposition (POD) in mechanical engineering [42], and empirical orthogonal functions (EOF) [43] or empirical eigenfunction decomposition [44] in meteorology. Moreover, there are two names of PCA that are widely used: "Principal Component Analysis" and "Principal Components Analysis". Jolliffe [22] found after searching for references to the two forms using the *Web of Science* that the singular form is the most popular, hence we use the singular form. Secondly, the nomenclature of the PCA quantities is very confusing. In this section we will focus on the nomenclature of the most important quantities: $\boldsymbol{A}$ and $\boldsymbol{Z}$. Table 2.1 shows the nomenclature that is used for $\boldsymbol{A}$ and $\boldsymbol{Z}$ by seven authors. The names differ greatly and cause great confusion, especially the usage of "Principal Components" for both $\boldsymbol{A}$ and $\boldsymbol{Z}$. Therefore, we will proceed to determine the appropriate names for the two quantities.

Jolliffe [22] notices that the vectors $\boldsymbol{a}_i$ are sometimes referred to as "Principal Components", but notes that this usage is confusing. He argues that the term "Principal Components" is preferred to be reserved for the new variables that are linear combinations of the original variables, where $\boldsymbol{a}_i$ contains the coefficients of the ith PC. This reasoning is correct: $\boldsymbol{A}$ does not contain the PCs, but it contains the coefficients that tells us how to construct the PCs from our variables. After all, the variance of the ith PC is not calculated as var($\boldsymbol{a}_i$), but as var($\boldsymbol{X}\boldsymbol{a}_i$). Jolliffe sometimes uses "loadings" as an alternative for $\boldsymbol{A}$, but this usage is slightly unfortunate, since the word loadings is sometimes used for the scaled eigenvectors of the covariance matrix [38]: $\boldsymbol{A}^* = \boldsymbol{A}\boldsymbol{L}^{1/2}$. Thus the matrix $\boldsymbol{A}$ is preferably called the coefficient matrix or weights. The naming of matrix $\boldsymbol{Z}$ is less confusing, despite the variation of names used. Most of the names include "score". The matrix $\boldsymbol{Z}$ is essentially the data set $\boldsymbol{X}$ measured by the PCs, thereby giving the scores of the data sets for the PCs.

Bro and Smilde [45] give a description of the term "Principal Component". They notice that this term is not clearly defined and is sometimes used for $\boldsymbol{A}$ or $\boldsymbol{Z}$ or both. They reserve the term Principal Component for the pair $\boldsymbol{A}$ and $\boldsymbol{Z}$, since both are closely tied together. This reasoning is intuitive, because the PCs are variables that are constructed using $\boldsymbol{A}$ and have scores $\boldsymbol{Z}$. For the above mentioned reasons, we will in this work call the matrix $\boldsymbol{A}$ the PC coefficients and $\boldsymbol{Z}$ the PC scores.

Table 2.1: Nomenclature for $\boldsymbol{A}$ and $\boldsymbol{Z}$. Jackson does not give one clear name for $\boldsymbol{A}$, but uses several descriptions like matrix of directional cosines and matrix containing characteristic vectors.

Author	$\boldsymbol{A}$	$\boldsymbol{Z}$	Reference
Peerenboom	Principal Components	PC scores	[11]
Jolliffe	Coefficients / loadings	PC scores	[22]
Abdi	Principal Components	Factor scores	[46]
Jackson	-	Z-scores	[39]
Wold	Loading vectors	Score vectors	[23]
Dunteman	Weights / loadings	Principal Components	[47]
Bro	Weights / loadings	Score vector	[45]

2.4 The PCA algorithm

This section describes the basic steps that are taken in PCA. PCA is applied on a data set, which is stored in a matrix $\boldsymbol{X}^0$ $(n \times Q)$ that contains n observations of Q variables. It is common in PCA to do two steps of data pre-processing. First, the data is centered by subtracting the mean of each variable from the data $\boldsymbol{X}^0$. The consequence of centering is that all observations will be represented as fluctuations [48]. Scaling is the second step of pre-processing for PCA. The step of scaling can be seen as making the variables dimensionless. There are multiple scaling methods available. Section 2.5 focuses on the importance of scaling more elaborately. The steps of centering and scaling are done as follows:

$$\boldsymbol{X} = (\boldsymbol{X}^0 - \bar{\boldsymbol{X}}^0)\boldsymbol{D}^{-1}. \tag{2.19}$$

Here, $\boldsymbol{X}$ contains the centered and scaled data, $\bar{\boldsymbol{X}}^0$ the mean of each variable and $\boldsymbol{D}$ is a diagonal matrix that contains the scaling coefficients of the variables. The inverse of $\boldsymbol{D}$ is written as $\boldsymbol{D}^{-1}$. Now, after the data pre-processing, PCA is applied. There are two ways of performing PCA: using eigendecompsition of the covariance of the data and singular value decomposition (SVD) of the data matrix. First, we will treat the eigendecomposition of the covariance matrix.

2.4.1 Eigendecomposition of the covariance matrix

The covariance matrix $\boldsymbol{C}$ is calculated from the processed data $\boldsymbol{X}$ in the following way:

$$\boldsymbol{C} = \frac{1}{n-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}. \tag{2.20}$$

The off-diagonal elements of the covariance matrix measure the correlation between the variables, whereas the diagonal elements give the variance of the variables. The factor $1/(n-1)$ is the usual unbiased normalization for the sample covariance matrix. One of

the important properties of the covariance matrix is that it is a symmetric matrix. This property is very convenient, because the spectral theorem guarantees symmetric matrices to be orthogonally diagonizable:

$$\boldsymbol{C} = \boldsymbol{A}\boldsymbol{L}\boldsymbol{A}^{\mathrm{T}}. \tag{2.21}$$

Here, $\boldsymbol{L}$ is a diagonal matrix that contains the eigenvalues of the covariance matrix. In section 2.2, it was shown that the variance of the principal components are given by the eigenvalues of the covariance matrix. The eigenvalues are sorted in descending order, so that the variance of the first PC is maximized and contains most of the variation of the dataset. The matrix $\boldsymbol{A}$ contains the eigenvectors of the covariance matrix as column vectors. The eigenvectors of the covariance matrix contain the principal axes of the PCs. Finally, the original dataset is projected onto the PC basis in order to obtain the PC scores:

$$\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{A}. \tag{2.22}$$

The original data can be recovered from the PC scores in a similar manner:

$$\boldsymbol{X} = \boldsymbol{Z}\boldsymbol{A}^{-1} = \boldsymbol{Z}\boldsymbol{A}^{\mathrm{T}}, \tag{2.23}$$

where we used the fact that the inverse of $\boldsymbol{A}$ is equal to its transpose. This is true for all orthogonal matrices.

2.4.2 Singular value decomposition of the data matrix

One of the disadvantages of finding the PCs from the eigendecomposition of the covariance matrix is that a loss of precision is involved in the calculation of the covariance matrix. Therefore, we will present an alternative method of finding the PCs, where we do not need to calculate the covariance matrix explicitly. This is done using singular value decomposition (SVD) of the data matrix.

In the previous section we considered the decomposition of the covariance matrix, which is a symmetric matrix. Symmetric matrices have the convenient property of being orthogonally diagonalizable. But not all matrices are symmetric and some are not square. In those cases one might look for other types of decompositions that are valid for any $(n \times Q)$ matrices. The singular value decomposition is one of these methods. The matrix factorization of SVD is done in the following way [49]:

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} \tag{2.24}$$

in which $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal matrices but not necessary the same. The columns of $\boldsymbol{U}$ are called the left singular vectors of $\boldsymbol{X}$ and has size $n \times n$. The columns of $\boldsymbol{V}$ are called the right singular vectors of $\boldsymbol{X}$ and has size $Q \times Q$. The matrix $\boldsymbol{\Sigma}$ is a matrix of the same size as $\boldsymbol{X}$, $(n \times Q)$, that contains the singular values of $\boldsymbol{X}$ on its main diagonal and zeros at the other entries. If $\lambda_1, \lambda_2, \ldots, \lambda_Q$ are the eigenvalues of $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$, then the numbers $\sigma_1 = \sqrt{\lambda_1}, \sigma_2 = \sqrt{\lambda_2}, \ldots, \sigma_Q = \sqrt{\lambda_Q}$ are called the singular values of $\boldsymbol{X}$.

In practice, the full singular value decomposition of some matrix $\boldsymbol{X}$ will look as follows:

$$\begin{aligned}\boldsymbol{X} &= \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} \\ &= \begin{pmatrix}\boldsymbol{u}_1 & \boldsymbol{u}_2 & \cdots & \boldsymbol{u}_Q \,\middle|\, \boldsymbol{u}_{Q+1} & \cdots & \boldsymbol{u}_n\end{pmatrix} \left(\begin{array}{cccc} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_Q \\ \hline \multicolumn{4}{c}{\mathbf{0}_{(n-Q)\times Q}} \end{array}\right) \begin{pmatrix}\boldsymbol{v}_1^{\mathrm{T}} \\ \boldsymbol{v}_2^{\mathrm{T}} \\ \vdots \\ \boldsymbol{v}_Q^{\mathrm{T}}\end{pmatrix}\end{aligned} \tag{2.25}$$

We see that the matrix $\boldsymbol{\Sigma}$ containing the singular values has at least $n - Q$ rows of zeros on the bottom if the data matrix $\boldsymbol{X}$ contains more observations than variables. When multiplying $\boldsymbol{U}$ with $\boldsymbol{\Sigma}$, we see that column $(Q + 1)$ upto n of $\boldsymbol{U}$ are always multiplied with these rows of zeros in $\boldsymbol{\Sigma}$. Therefore, column $(Q + 1)$ upto n as well as row $(Q + 1)$ upto n of $\boldsymbol{\Sigma}$ can be removed without losing any information. This reduced form of SVD is sometimes called the economy SVD and has the following form:

$$\begin{aligned}\boldsymbol{X} &= \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} \\ &= \begin{pmatrix}\boldsymbol{u}_1 & \boldsymbol{u}_2 & \cdots & \boldsymbol{u}_Q\end{pmatrix} \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_Q \end{pmatrix} \begin{pmatrix}\boldsymbol{v}_1^{\mathrm{T}} \\ \boldsymbol{v}_2^{\mathrm{T}} \\ \vdots \\ \boldsymbol{v}_Q^{\mathrm{T}}\end{pmatrix}\end{aligned} \tag{2.26}$$

Note that a similar reduction is possible when the data matrix contains more variables than observations. In that case column $(n + 1)$ upto Q of $\boldsymbol{V}$ and $\boldsymbol{\Sigma}$ can be safely deleted. The advantage of the economy SVD is that it is much cheaper to compute than the full SVD. The computation of the full SVD is often not necessary, in particular when $\boldsymbol{U}$ and $\boldsymbol{\Sigma}$ do not need to be known individually, but only the product $\boldsymbol{U}\boldsymbol{\Sigma}$. This will appear to be the case for PCA.

In order to find the PCs using SVD we first need to calculate the covariance matrix. The covariance matrix is in this case:

$$\begin{aligned}\boldsymbol{C} &= \frac{1}{n-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X} \\ &= \frac{1}{n-1}\left(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}\right)^{\mathrm{T}}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} \\ &= \frac{1}{n-1}\boldsymbol{V}\boldsymbol{\Sigma}^{\mathrm{T}}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} \\ &= \boldsymbol{V}\frac{\boldsymbol{\Sigma}^{\mathrm{T}}\boldsymbol{\Sigma}}{n-1}\boldsymbol{V}^{\mathrm{T}},\end{aligned} \tag{2.27}$$

where we used that $\boldsymbol{U}$ is an orthogonal matrix. Here, $\boldsymbol{\Sigma}^{\mathrm{T}}\boldsymbol{\Sigma}$ is a square $(n \times n)$ matrix and $\boldsymbol{V}$ is orthogonal. Note that this equation is similar to equation (2.21). The eigenvalues of the covariance matrix can be calculated from the singular values using $\boldsymbol{L} = \boldsymbol{\Sigma}^{\mathrm{T}}\boldsymbol{\Sigma}/(n-1)$.

The eigenvectors of the covariance matrix are just the right singular vectors of $\boldsymbol{X}$: $\boldsymbol{A} = \boldsymbol{V}$. The PC scores are calculated by projecting the data matrix onto the PC basis:

$$\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{A} = \boldsymbol{X}\boldsymbol{V} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V} = \boldsymbol{U}\boldsymbol{\Sigma}, \tag{2.28}$$

where we used that $\boldsymbol{V}$ is an orthogonal matrix. Here, we see that the usage of the economy SVD is justified, since in the process of finding the PCs we do not need to know $\boldsymbol{U}$ and $\boldsymbol{\Sigma}$ individually, but only their product.

Let us now consider an example that shows the advantage of using SVD. A well-known example that indicates the danger of calculating $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ is the Läuchli matrix [50]. The Läuchli matrix is a $(n+1) \times n$ rectangular matrix with ones on the top row and $\varepsilon > 0$ on the diagonal starting at $(2, 1)$. For $n = 3$, the Läuchli matrix takes the following form:

$$\boldsymbol{X} = \begin{pmatrix} 1 & 1 & 1 \\ \varepsilon & 0 & 0 \\ 0 & \varepsilon & 0 \\ 0 & 0 & \varepsilon \end{pmatrix}. \tag{2.29}$$

For this matrix, we will determine the principal axes ($\boldsymbol{A}$) and their variances ($\boldsymbol{L}$) analytically and compare the latter with numerical results using the eigendecomposition of $\boldsymbol{C}$ and the SVD of $\boldsymbol{X}$. For SVD, the variance of the PCs is calculated by squaring the singular values and dividing by $n-1$. Results are given in table 2.2. The covariance matrix for the Läuchli matrix is:

$$\boldsymbol{C} = \begin{pmatrix} 1+\varepsilon^2 & 1 & 1 \\ 1 & 1+\varepsilon^2 & 1 \\ 1 & 1 & 1+\varepsilon^2 \end{pmatrix}, \tag{2.30}$$

where the normalization of the covariance matrix was omitted. The eigenvectors and eigenvalues of the covariance matrix are the following:

$$\boldsymbol{A} = \begin{pmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & 0 & -\frac{2}{\sqrt{6}} \end{pmatrix}, \quad \boldsymbol{L} = \begin{pmatrix} 3+\varepsilon^2 & 0 & 0 \\ 0 & \varepsilon^2 & 0 \\ 0 & 0 & \varepsilon^2 \end{pmatrix} \tag{2.31}$$

The computation of the eigenvalues will become more difficult for small values of ε, which can be seen from the covariance matrix. Its diagonal elements will become close to 1. First, we take $\varepsilon = 10^{-5}$. For both the eigendecomposition of the covariance matrix and the SVD of the data matrix, we find correct values for the second and third eigenvalues: 10^{-10}. Differences between the two methods appear when we choose $\varepsilon = 10^{-10}$. The SVD gives the correct for the second and third eigenvalue of 10^{-20}, whereas the eigendecomposition gives incorrect values of $-2.37 \cdot 10^{-17}$ and $3.33 \cdot 10^{-16}$. These differences can be explained by looking at the condition numbers of the data matrix and of the covariance matrix. The condition number of a matrix gives an estimate of the accuracy of matrix operations. High condition numbers generally give inaccurate results, whereas low condition numbers give

Table 2.2: Results for the comparison between the eigendecomposition of the covariance matrix and the singular value decomposition of the data matrix. The data matrix in this example was the Läuchli matrix

	$\varepsilon = 10^{-5}$			$\varepsilon = 10^{-10}$		
	λ_2	λ_3	cond	λ_2	λ_3	cond
Analytical	10^{-10}	10^{-10}	-	10^{-20}	10^{-20}	-
EVD	10^{-10}	10^{-10}	$3 \cdot 10^{10}$	$-2.37 \cdot 10^{-17}$	$-3.33 \cdot 10^{-16}$	$3 \cdot 10^{20}$
SVD	10^{-10}	10^{-10}	$\sqrt{3} \cdot 10^{5}$	10^{-20}	10^{-20}	$\sqrt{3} \cdot 10^{10}$

accurate results. A matrix with an infinitely high condition number is not invertible. The condition number of matrix $\boldsymbol{A}$ is given by:

$$\kappa(\boldsymbol{A}) = \frac{\sigma_{max}(\boldsymbol{A})}{\sigma_{min}(\boldsymbol{A})}, \tag{2.32}$$

where $\sigma_{max}(\boldsymbol{A})$ and $\sigma_{min}(\boldsymbol{A})$ are the maximum and minimum singular values of $\boldsymbol{A}$. For the Läuchli matrix and its covariance matrix, we find the following condition numbers:

$$\kappa(\boldsymbol{X}) = \sqrt{1 + \frac{3}{\varepsilon^2}}, \quad \kappa(\boldsymbol{C}) = 1 + \frac{3}{\varepsilon^2}. \tag{2.33}$$

The condition number of the covariance matrix is the square of the condition number of the data matrix. This means that the condition number of the data matrix is always lower than the condition number of the covariance matrix and therefore the SVD of the data matrix is more accurate than the eigendecomposition of the covariance matrix.

2.5 The importance of scaling

Centering and scaling are two important steps of pre-processing that are often done before PCA. Centering is done because the covariance matrix is calculated for centered data. Scaling, which will be discussed in this section, is a very important step that is usually done before PCA. For some types of scaling, outlier removal is necessary.

In the derivation of the PCs, we showed that the loadings of the PCs are calculated as the eigenvalues of the covariance matrix. It is common to calculate the PCs from a data set where the variances of the variables are set to unity. This is done by scaling the data with scaling matrix $\boldsymbol{D}$ in (2.19), which has diagonal elements $D_{ii} = \sqrt{\text{var}(\boldsymbol{x}_i)} = \sigma_i$. This choice means that we use the correlation matrix instead of the covariance matrix for the calculation of the PCs:

$$\boldsymbol{C}^* = \frac{1}{n-1} \left(\boldsymbol{X}\boldsymbol{D}^{-1}\right)^{\mathrm{T}} \boldsymbol{X}\boldsymbol{D}^{-1} = \frac{1}{n-1} \boldsymbol{D}^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{D}^{-1} = \boldsymbol{D}^{-1}\boldsymbol{C}\boldsymbol{D}^{-1}. \tag{2.34}$$

The elements of the correlation matrix $\boldsymbol{C}^*$, also called Pearson correlation coefficients, are given by:

$$C^*_{ij} = \frac{C_{ij}}{\sigma_i \sigma_j} = \frac{\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j}{\sqrt{\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_i} \sqrt{\boldsymbol{x}_j^{\mathrm{T}} \boldsymbol{x}_j}} = \frac{\boldsymbol{x}_i \cdot \boldsymbol{x}_j}{\|\boldsymbol{x}_i\| \, \|\boldsymbol{x}_j\|}. \tag{2.35}$$

Here, we used a data matrix $\boldsymbol{X}$ with zero mean. This shows that all diagonal elements will be equal to 1, which confirms the standardization of the variances of the variables. All off-diagonal elements will be between -1 and 1 due to the Cauchy-Schwarz inequality.

It is important to notice that the eigenvalues and eigenvectors of the correlation matrix are generally not the same as those of the covariance matrix. The choice of using the correlation matrix instead of the covariance matrix is a definite but arbitrary decision to make all variables equally important in terms of their variance. A major argument for using the correlation matrix rather than the covariance matrix for PCA is that the results of analyses for different sets of variables are more comparable for the choice of the correlation matrix. PCA based on covariance matrices gives PCs that are very sensitive to the units of the variables. If there are large differences between the variances of the variables, then the variables with the highest variances will tend to dominate the first few PCs. This might be acceptable when all variables have the same units, but this is often not the case. We can demonstrate this with the following example.

Suppose we have two variables $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. The variable $\boldsymbol{x}_1$ can be measured in both centimeters and millimeters. Variable $\boldsymbol{x}_2$ is a variable that does not measure a length, but something else. The covariance matrices for the two cases are known and given by:

$$\boldsymbol{C}_{\mathrm{cm}} = \begin{pmatrix} 80 & 44 \\ 44 & 80 \end{pmatrix} \quad \text{and} \quad \boldsymbol{C}_{\mathrm{mm}} = \begin{pmatrix} 8000 & 440 \\ 440 & 80 \end{pmatrix}. \tag{2.36}$$

The first PC using centimeters is $0.707x_1 + 0.707x_2$, whereas the first PC using millimeters is $0.998x_1 + 0.055x_2$. The correlation matrices for the two cases are equal:

$$\boldsymbol{C}^*_{\mathrm{cm}} = \boldsymbol{C}^*_{\mathrm{mm}} = \begin{pmatrix} 1 & \frac{44}{80} \\ \frac{44}{80} & 1 \end{pmatrix}, \tag{2.37}$$

with a first PC of $0.707x_1 + 0.707x_2$. In this case the choice of units did not influence the PCs.

It is also possible to use different scaling methods besides normalizing the variances of the variables [48, 11]. These scaling methods are listed in table 2.3. Ideally, the choice of the scaling method is based on some *a priori* idea of the relative importance of the variables. In practice, it is quite rare that a set of variables suggests a certain scaling method [22]. These scaling methods will be discussed in short below [48]:

1. *Auto scaling.* This scaling method is also called unit variance scaling. Each variable is scaled with its standard deviation σ_i, so that all the variables have a standard deviation of one. The covariance matrix therefore becomes a correlation matrix.

Table 2.3: Different scaling methods. For the standard deviation of variable i we use the symbol σ_i.

Method	Scaling coefficient
auto	$D_{ii} = \sigma_i$
pareto	$D_{ii} = \sqrt{\sigma_i}$
vast	$D_{ii} = \sigma_i^2/\bar{x}_i^0$
range	$D_{ii} = \max(x_i^0 - \bar{x}_i^0) - \min(x_i^0 - \bar{x}_i^0)$
level	$D_{ii} = \bar{x}_i^0$

2. *Pareto scaling.* Pareto scaling differs slightly from auto scaling. Instead of the standard deviation it uses the square root of the standard deviation of the variables for the scaling. Consequently, the variables are scaled in such a way that their variance are set equal to its standard deviation.

3. *Vast scaling.* The word VAST is an acronym of for VAriable STability scaling. The scaling factor is given by the product of the standard deviation and the coefficient of variation, defined as $\sigma_i/\bar{x}_i^0$, with $\bar{x}_i^0$ the mean of variable i. It gives a higher importance for variables with a small relative standard deviation.

4. *Level scaling.* Level scaling uses the mean of the variables as the scaling factor. The presence of outliers could affect the scaling. Therefore, the mean should be determined after outlier removal or alternatively, a more robust estimator of the mean (e.g. the median) could be used. Level scaling gives high importance to variables with a large relative standard deviation. Parente [48] suggests to use level scaling when large relative changes of variables are of interest.

5. *Range scaling.* Range scaling uses the difference between the maximal and minimum value with respect of the mean of the variables as scaling factors. The main disadvantage of range scaling is that only two values are used to determine the range, whereas the other scaling methods take all values into account. Therefore, range scaling is more sensitive to outliers. The robustness of the range scaling method can be increased by removing outliers or by using robust estimators for maximum and minimum values of the variables.

2.6 Properties of PCA

This section focuses on some of the properties of the PCs. First, the variance of the PCs is discussed and how they relate to the original variables. After that, we have a look at the assumptions that were made in the derivation of the PCs in section 2.2 and their consequences.

We have seen in the derivation of the PCs that the eigenvalues represent the variances of the PCs. The sum of these variances is given by:

$$\sum_{i=1}^{Q} \text{var}(\boldsymbol{z}_i) = \sum_{i=1}^{Q} \lambda_i = \text{tr}(\boldsymbol{L}), \tag{2.38}$$

where $\boldsymbol{L}$ is the matrix containing the eigenvalues of the covariance matrix, as was defined in section 2.4.1. We can use the eigendecomposition of the covariance matrix to simplify this result:

$$\text{tr}(\boldsymbol{L}) = \text{tr}(\boldsymbol{A}^{\text{T}}\boldsymbol{C}\boldsymbol{A}) = \text{tr}(\boldsymbol{C}\boldsymbol{A}\boldsymbol{A}^{\text{T}}) = \text{tr}(\boldsymbol{C}) = \sum_{i=1}^{Q} \text{var}(\boldsymbol{x}_i), \tag{2.39}$$

where we used the fact that the trace is invariant under cyclic permutations. This result shows that the sum of the variances of the original variables and of the PCs are equal.

This result can be used to calculate the fraction of the total variance that is accounted for by q PCs [48]:

$$t_q = \frac{\sum_{i=1}^{q} \text{var}(\boldsymbol{z}_i)}{\sum_{i=1}^{Q} \text{var}(\boldsymbol{x}_i)} = \frac{\sum_{i=1}^{q} \lambda_i}{\sum_{i=1}^{Q} \lambda_i}, \tag{2.40}$$

with $0 \leq t_q \leq 1$. The fraction t_q is often used to decide how many PCs are to be used. The goal is to choose q as small as possible while still having a high value for t_q. One might set a threshold for t_q and choose the lowest value for q, for which t_q still exceeds this threshold.

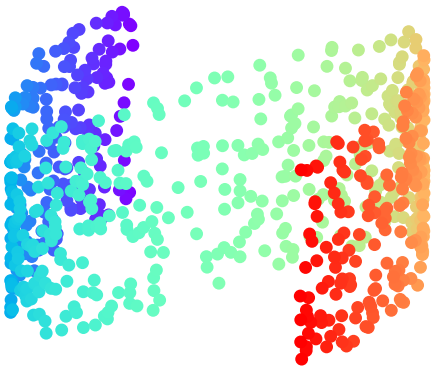
A couple of assumptions were made in the derivation of the PCs. Here, we will reflect upon these assumptions and their consequences.

1. Orthogonality. This is due to the fact that the PCs are uncorrelated. The main advantage of the orthogonality assumption is that linear algebra decomposition techniques can be used in PCA.

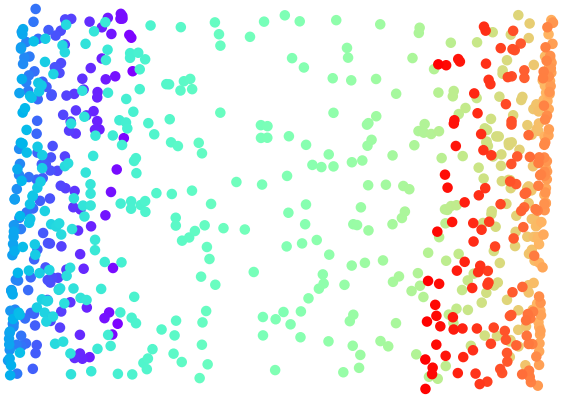
2. Importance = variance. The sorting of the PCs by importance is quantified by the variance of the PCs. This is based on the assumption that the data has a high signal-to-noise ratio (SNR). Consequently, components with larger variances correspond to interesting dynamics, the signal, and lower ones correspond to noise.

3. Mean and variance are sufficient statistics. A statistic satisfies the criterion of sufficiency when no other statistic which can be calculated from the sample provides any additional information as to the value of the parameter [51]. PCA only uses the mean (due to centering) and variance as statistics to describe the entire probability distribution. The only probability distribution function with zero mean that is described by solely the variance is the Gaussian distribution. Therefore, PCA is expected to work well on variables that follow a Gaussian distribution. Fortunately, a lot of the real world data is Gaussian distributed, thanks to the Central Limit Theorem. PCA is usually a robust solution to slight deviations from this assumption.

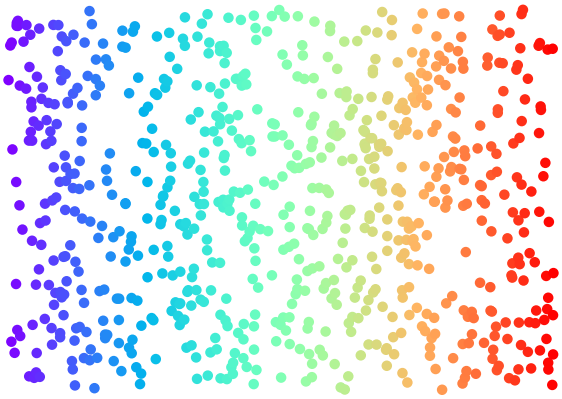
4. Linearity. The linearity of PCA is evident from the PCs, which are linear combinations of the original variables The assumption of linearity is a very powerful and convenient property. It frames the problem as a change of basis. The consequence of the assumption of linearity is that in some cases nonlinearities cannot be captured. This will be shown in the example of figure 2.4. Figure 2.4a shows the curved plane on which PCA is applied. The scores of the first two PCs are plotted in 2.4b, which shows that the colors are mixed and cannot be resolved correctly. There exist extensions of PCA that are able to deal with nonlinearities in data, but this goes beyond the scope of this work. A short text about nonlinear PCA is found in appendix A.



(a) Example



(b) PCA



(c) Nonlinear methods

Figure 2.4: Example where PCA fails. PCA is not able to recover the 2D manifold in 3D space, unlike nonlinear methods such as isomap.

Chapter 3

Principal Component Analysis as a reduction method

The plasma state of matter is known for the complex interactions between neutrals, ions and electrons. There are several kinds of plasma modelling tools available, for example, single particle (particle-in-cell), kinetic and fluid models. An example of a fluid model is the global model, which is a volume-averaged model that does not resolve spatial derivatives. In this chapter, the global model is introduced and the PCA global model derived from it. Next, some essential aspects of the PCA global model are discussed, such as the log transformation, the manifold and the backward transformation. The final section of this chapter focuses on an alternative reduction method: the Intrinsic Low Dimensional Manifold (ILDM).

3.1 The global model

The fluid equations that are used for the global model are derived from the moments of the Boltzmann equation, see for example [52]. We start with the zeroth moment of the Boltzmann equation, which results in the conservation of mass and is written as:

$$\frac{\partial n_i}{\partial t} + \nabla \cdot (n_i \vec{v}_i) = S_i, \tag{3.1}$$

where n_i is the density, $\vec{v}_i$ the mean velocity and S_i the source term of species i. In a global model, transport is often written as a transport frequency $\mathcal{F}_i$:

$$\mathcal{F}_i = \frac{1}{n_i} \nabla \cdot (n_i \vec{v}_i), \tag{3.2}$$

and thus equation (3.1) is written as:

$$\frac{\partial n_i}{\partial t} + \mathcal{F}_i n_i = S_i. \tag{3.3}$$

In this work, we will not consider transport, hence $\mathcal{F}_i$ is set to zero:

$$\frac{\partial n_i}{\partial t} = S_i. \tag{3.4}$$

It is convenient to write this equation in vector notation, since we are dealing with a system of multiple species:

$$\frac{\partial \boldsymbol{n}}{\partial t} = \boldsymbol{S}. \tag{3.5}$$

The source vector $\boldsymbol{S}$ deals with the production and destruction of species, which happens through reactions. This global model includes reactions in the following general form:

$$\sum_i a_{i,r}^{\mathrm{L}} Y_i^{\mathrm{L}} \xrightarrow{k(T_\mathrm{e},T_\mathrm{g})} \sum_i a_{i,r}^{\mathrm{R}} Y_i^{\mathrm{R}}, \tag{3.6}$$

which describes the production of species Y_i^{R} and the destruction of species Y_i^{L}. The superscripts L and R refer to respectively the left and right hand side of the reaction. The constants $a_{i,r}^{\mathrm{L}}$ and $a_{i,r}^{\mathrm{R}}$ are called the stoichiometric coefficients. Each reaction has a rate coefficient $k(T_\mathrm{e}, T_\mathrm{g})$ depends on the electron temperature T_e and the gas temperature T_g. The rate at which this reaction occurs is given by:

$$R_r(\boldsymbol{n}, T_\mathrm{e}, T_\mathrm{g}) = k(T_\mathrm{e}, T_\mathrm{g}) \prod_i n_i^{a_{i,r}^{\mathrm{L}}}, \tag{3.7}$$

where the product runs over all species i. The net source of species i is then calculated by summing over the reaction rates, multiplied by the net stoichiometric coefficients:

$$S_i = \sum_r \left(a_{i,r}^{\mathrm{R}} - a_{i,r}^{\mathrm{L}}\right) R_r, \tag{3.8}$$

or written in vector notation:

$$\boldsymbol{S} = \boldsymbol{W}\boldsymbol{R}, \tag{3.9}$$

where $\boldsymbol{W}$ is the stoichiometry matrix containing the net stoichiometric coefficients.

The first moment of the Boltzmann equation represents the conservation of momentum, but is not included in the global model because the global model is not spatially resolved. The second moment deals with the conservation of energy, which for the global model reads [53]:

$$\frac{\partial}{\partial t}\left(\frac{3}{2} n_\mathrm{e} k_\mathrm{B} T_\mathrm{e}\right) = P_\mathrm{input}(t) - Q_\mathrm{inelas} - Q_\mathrm{elas} + Q_\mathrm{extra}, \tag{3.10}$$

where n_e is the electron density, k_B the Boltzmann constant, T_e the electron temperature, P_input the input power density, Q_extra extra energy density source terms, and Q_inelas and Q_elas the energy density losses from inelastic and elastic processes between electrons and heavy particles. Note that this equation only deals with the energy of the electrons, the temperature of the heavy species is assumed to be constant in this work. More details about the global model can be found in [53].

3.2 Derivation of the PCA global model

Sutherland and Parente [54] proposed to derive PC continuity equations from the species continuity equations. In this work we focus on the reduction of global models, and a derivation of the 0D continuity equations for a global model is presented by Peerenboom *et al.* [11]. The derivation was done by applying the steps of the PCA algorithm as described in 2.4 on the species balance equations of the global model. We start with the coupled set of differential equations describing the time evolution of the species, which is given by equation (3.5). The first step of PCA is centering:

$$\frac{\partial(\boldsymbol{n} - \bar{\boldsymbol{n}})}{\partial t} = \boldsymbol{S}, \tag{3.11}$$

where $\boldsymbol{n}$ is a row-vector that contains the species densities n_i and $\bar{\boldsymbol{n}}$ contains the mean of all the species densities. Scaling is the second step:

$$\frac{\partial(\boldsymbol{n} - \bar{\boldsymbol{n}})\,\boldsymbol{D}^{-1}}{\partial t} = \frac{\partial \boldsymbol{X}}{\partial t} = \boldsymbol{S}\boldsymbol{D}^{-1}, \tag{3.12}$$

with $\boldsymbol{D}^{-1}$ the inverse of the scaling matrix and $\boldsymbol{X}$ the scaled variables. Section 2.5 explains how the scaling matrix is obtained. This equation is multiplied with the PC coefficients $\boldsymbol{A}$ to find the PC continuity equations:

$$\frac{\partial(\boldsymbol{X}\boldsymbol{A})}{\partial t} = \boldsymbol{S}\boldsymbol{D}^{-1}\boldsymbol{A}. \tag{3.13}$$

Here, the PC scores $\boldsymbol{Z}$ come into play, since $\boldsymbol{X}\boldsymbol{A} = \boldsymbol{Z}$, according to equation (2.1). This equation is often rewritten as:

$$\frac{\partial \boldsymbol{Z}}{\partial t} = \boldsymbol{S}_{\boldsymbol{Z}}, \tag{3.14}$$

where $\boldsymbol{S}_{\boldsymbol{Z}}$ is the source term of $\boldsymbol{Z}$, defined as:

$$\boldsymbol{S}_{\boldsymbol{Z}} = \boldsymbol{S}\boldsymbol{D}^{-1}\boldsymbol{A}. \tag{3.15}$$

Before solving the PC continuity equations, it is necessary to determine the PCs. In PCA, the PCs are calculated from a data set. Therefore, a training set is needed for a PCA global model. Ideally, the training set is generated using cheap 0D or 1D models that are representative for the chemistry. We include output of at least two models in the training set. For the models we use the same chemistry, but we vary a certain parameter This parameter could for example be the ionization degree or the pressure. It is important that the variation of the parameter is visible in the initial conditions.

There are three vital elements of the PCA global model that are discussed in the following sections. In the next section, we focus on the lower dimension space in which the PC continuity equations are solved: the manifold. After that, we study some methods for the reconstruction of the species densities from the PC scores. The log tranformation, which gives several improvements for the manifold and the reconstruction, is treated last.

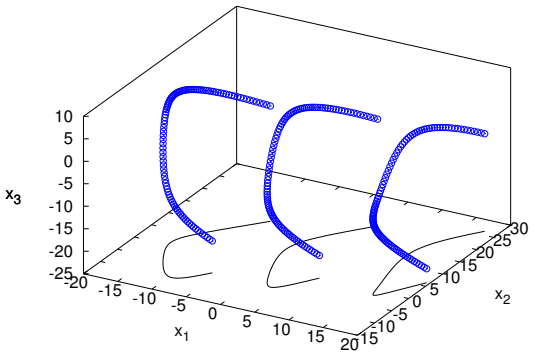
3.3 Manifold

PCA is a reduction method that uses the approach of reparametrization of the chemical state space. This approach is very common in the field of combustion modelling, but is still relatively new in plasma physics modelling. It is based on the fact that the composition often lies on a so-called low dimensional manifold. The existence of this low dimensional space enables the mixture composition to be described by only a limited amount of parameters. In the case of PCA, the parameters that describe the composition manifold are the Principal Components. The manifold will prove to be an essential part of a PCA global model. In this chapter we will use an artificial example to explain the manifold. We consider some training data, containing variables x_1, x_2 and x_3, shown in figure 3.1a. The training data is plotted in figure 3.1a (blue circles). The training data consists of three different data sets, which can be seen from the three separate trajectories. The black lines in the x_1,x_2 plane serve to better visualize the data. The centered and range-scaled data is shown in figure 3.1b.

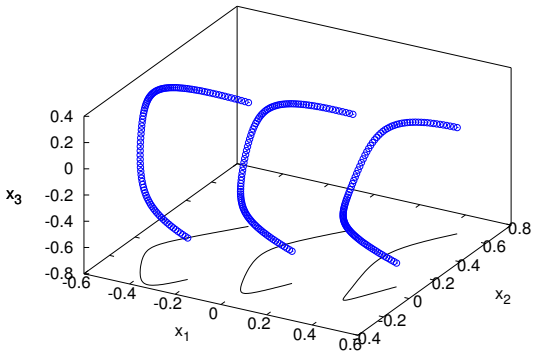
After the data pre-processing, the PC scores are calculated through a rotation of the data. The PC scores are represented by the blue circles in figure 3.1c for 'range' scaling and 3.1d for 'vast' scaling. The expressions for the scaling coefficients are given in table 2.3. The manifold is created by only taking into account $q = 2$ PCs, which is done by projecting the PC scores onto the plane spanned by the first two PCs. The manifolds for the two scaling methods are given by the green circles in figure 3.1c and 3.1d. The manifolds are clearly different, and this difference exists due to scaling. This can be explained by considering the PC scores $\boldsymbol{Z}$. The PC scores are calculated by $\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{A}$. The training set $\boldsymbol{X}$ depends on the scaling because scaling is part of the data pre-processing. Also the basis of the PCs $\boldsymbol{A}$ depends on the scaling, because $\boldsymbol{A}$ depends on the covariance matrix and the covariance matrix depends on $\boldsymbol{X}$. This shows that the PC scores, and thus the shape of the manifold, depend on the scaling. Unfortunately, it is very hard to predict the shape of the manifold prior to performing PCA.

One of the important properties of the manifold is its uniqueness. A manifold is unique if the trajectories of the PC scores do not cross. It is possible that none of the five scaling methods give a unique manifold. One way of looking at PCA is by describing it as a rotation and stretching of the training set. The reduction is done by projecting this rotated and stretched data on a q dimensional hyperplane. This projection determines whether the manifold will be unique or not. A manifold that is not unique will appear to be problematic for the reconstruction of the densities and PC sources and must therefore be evaded. One way to deal with this issue is to increase the amount of PCs. The disadvantage of using more PCs is that the reconstruction of the species densities from the PC scores becomes more expensive, which might significantly increase the duration of the simulation. This will become clear in section 3.4.

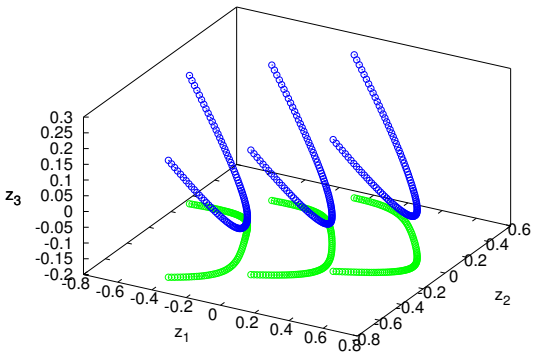
An alternative is to divide the manifold into several unique parts. An example of this is shown in figure 3.2, where the manifold for 'vast' scaling is split into three parts. These three parts are unique. One may argue that after splitting the data, the separate parts represent different training sets, and therefore PCA needs to be applied to these



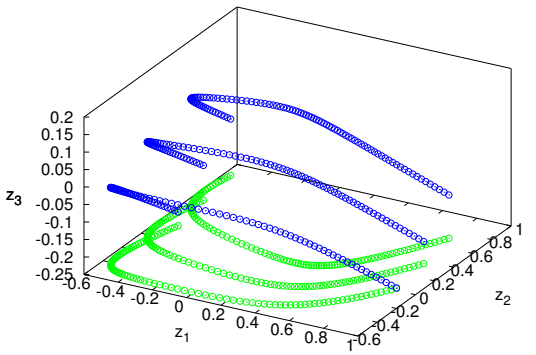
(a) Training data.



(b) Centered and range-scaled data.



(c) Manifold using range scaling.



(d) Manifold using vast scaling.

Figure 3.1: The artificial example. It shows the steps that are taken in PCA starting with the initial training set in (a), the centering and scaling in (b) and the calculation of the PC scores in (c) and (d) for two different scaling methods. The blue circles represent the training data or PC scores without dimensionality reduction. The black lines serve to better visualize the 3D data. The green circles show the reduced PC scores or manifold for $q = 2$ PCs. Figures (c) and (d) shows that the shape of the manifold depends on the scaling method.

training sets individually. The splitting is done by looking at the curvature of the manifold trajectories. The manifold is split at places where the curvature has a maximum. The amount of times that the manifold is split is specified by the user. Suppose we have a 2D curve that is parametrized by $\boldsymbol{r}(t) = (x(t), y(t))$. The curvature κ of a curve is defined as:

$$\kappa = \frac{\mathrm{d}\phi}{\mathrm{d}s}, \tag{3.16}$$

with ϕ the tangential angle and s the arc length of the curve. The curvature of $\boldsymbol{r}(t)$ can be calculated as [55]:

$$\kappa = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{3/2}}, \tag{3.17}$$

where the derivatives of x and y are taken with respect to time. In our case, we calculate the curvature of the manifold trajectories with variables $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$. Dalle [55] has shown that the curvature can be calculated very accurately by using a geometrical technique that involves the radius of curvature. The radius of curvature R is defined as the reciprocal of the curvature:

$$\kappa = \frac{1}{R}. \tag{3.18}$$

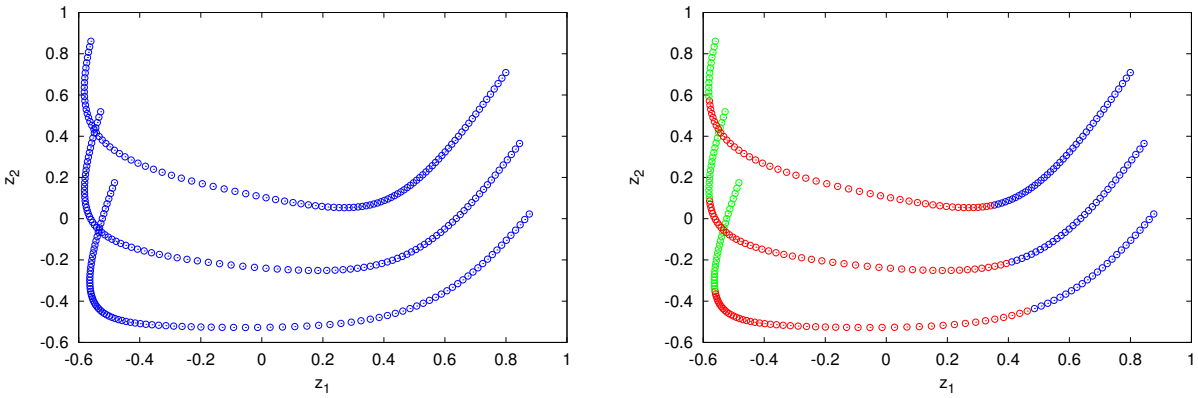
The radius of curvature at a certain point of the trajectory is approximated by calculating the radius of the circle that is constructed from the point and its neighbour points. For three points (x_1, y_1), (x_2, y_2) and (x_3, y_3) the curvature and the radius of curvature are calculated from the following expressions:

$$\begin{aligned} \kappa &= \frac{1}{R} = 4\frac{\Delta}{abc} \\ \boldsymbol{a} &= (x_1 - x_2, y_1 - y_2) \\ \boldsymbol{b} &= (x_3 - x_1, y_3 - y_1) \\ \boldsymbol{c} &= (x_3 - x_2, y_3 - y_2) \\ \Delta &= \frac{1}{2}\det\begin{pmatrix}\boldsymbol{b} & \boldsymbol{a}\end{pmatrix}, \end{aligned} \tag{3.19}$$

with $a = |\boldsymbol{a}|$, $b = |\boldsymbol{b}|$ and $c = |\boldsymbol{c}|$ [55].

3.4 Backward transformation

The backward transformation is essential for the performance of PCA as a reduction method. The transformation is used in two parts of the simulation: in the calculation of the PC source terms and the calculation of the densities from the PCs after the time integration. We will discuss three possible procedures for the backward transformation.



(a) Not split

(b) Split

Figure 3.2: The manifold for the example using 'vast' scaling. The manifold is not unique and is therefore split into separate parts in figure (b), based on the curvature of the PC scores.

3.4.1 Using matrix multiplication

The first method of reconstructing the densities and the PC sources from the PC scores is done by the matrix multiplication similar to equation (2.23). Since the PC continuity equations are solved for only the first q PCs, the reconstruction is done from only q PC scores:

$$\boldsymbol{X} \approx \boldsymbol{X}_q = \boldsymbol{Z}_q \boldsymbol{A}_q^{\mathrm{T}}, \tag{3.20}$$

where equation (2.19) shows how to undo the centering and scaling. An error is made when the densities are reconstructed from only q PCs. The fewer PCs are used, the greater the error will be. For the calculation of the PC sources, it is necessary to first calculate the sources of the species densities using equations (3.7) and (3.9). These sources of the species densities are a nonlinear function of the species densities. The error will increase after many iterations because of the nonlinearity of the sources. Consequently, the PC source terms quickly become inaccurate when using few PCs. Finally, the PC sources are calculated from the sources of the species densities through equation (3.15). In the next two subsections, we consider two alternatives for the reconstruction of the densities and the PC sources.

3.4.2 Using lookup tables

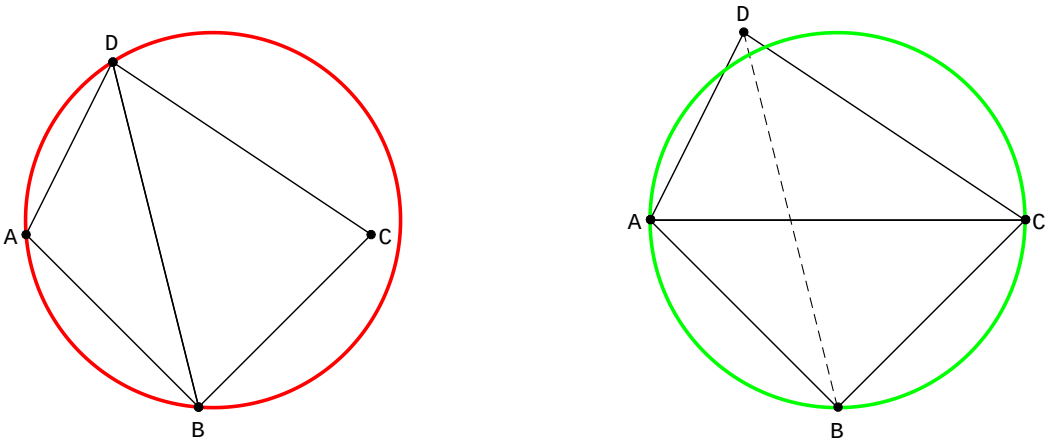
The usage of lookup tables for the reconstruction of the densities from the PC scores was proposed by Peerenboom *et al.* [11]. They used linear interpolation from 2D lookup tables to recover the densities and PC sources. In these lookup tables the densities and PC sources were stored as a function of $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$. The main advantage of the lookup tables is that $\boldsymbol{S_n}$ does not need to be evaluated. The lookup tables do unfortunately limit the

amount of PCs that can be used in the PCA global model, because the size of the lookup tables quickly becomes too large when more than 3 PCs are used. In this work we will use only two PCs for the lookup tables.

MATLAB offers two main types of data interpolation in lookup tables: interpolation for gridded data and for scattered data. The PC scores of the training set, which are the parameters of the lookup table, belong to the scattered data category. Amidror [56] discusses several scattered data interpolation methods, such as triangulation (or tetrahedrization) based methods, inverse distance weighted methods, radial basis function methods and natural neighbour methods. The MATLAB scattered data interpolation classes are all based on the approach of constructing a triangulation of the data. A triangulation is the subdivision of data into triangles by connecting the data points in a specific way. The most popular triangulation is the Delaunay triangulation, developed by Boris Delaunay in 1934 [57]. The Delaunay triangulation has geometric properties that are favourable for many interpolation applications. Delaunay triangulations try to avoid sliver triangles by maximizing the minimum angle of the triangles of the triangulation. This is done by a so-called edge flipping algorithm, which is illustrated in figure 3.3. Suppose we have four points for which we want to find the Delaunay triangulation. There are two ways ways to construct the triangulation, the first way is shown in figure 3.3a. In order to fulfill the Delaunay condition, the circumcircle of a triangle must not contain any other points. The red circle is the circumcircle of triangle ΔABD and contains the fourth point C, so the Delaunay condition is not met. This is solved by flipping the common edge BD, see figure 3.3b. The circumcircle of triangle ΔABC is given in green, which does not contain the fourth point D and thereby fulfills the Delaunay criterion. Another way of formulating the Delaunay triangulation is the requirement that, for the example in figure 3.3a, the sum of the angles of A and C must be smaller than $180°$.

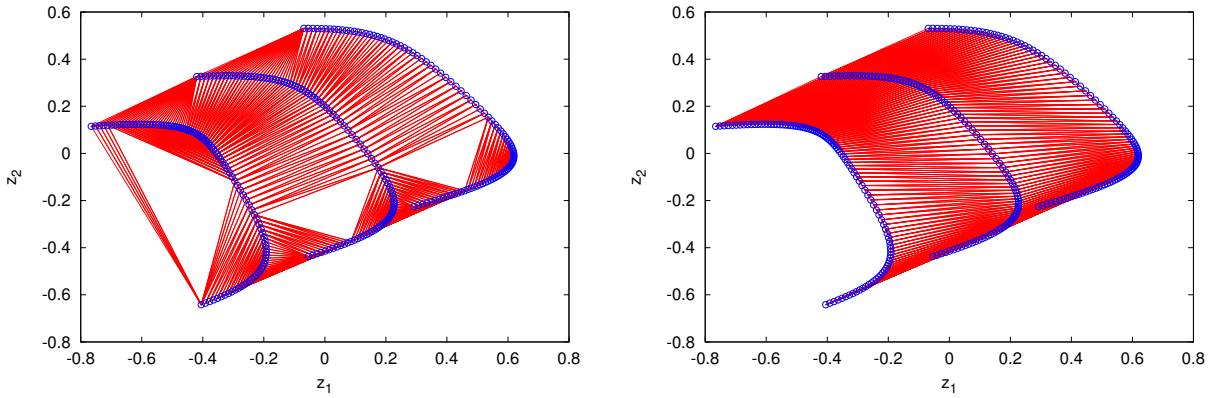
In our case, we construct a Delaunay triangulation of the manifold, see figure 3.4a for the Delaunay triangulation of the example. We see the appearance of two very large triangles between the first and second, and the second and third manifold trajectory. These large triangles are undesirable, because some points of the manifold are skipped. The appearance of these large triangles are inherent to the Delaunay triangulation, since it tries to maximize the minimum angles of the angles of the triangles. Another issue is that some triangles are formed that only contain points from one trajectory. Ideally, triangles are formed that contain two points from one trajectory and one from a different one. Thus, the Delaunay triangulation does not seem to be a viable triangulation for this study. We will consider a custom triangulation, where we force that every triangle contains two points from one trajectory and one point from a different one. This is done in an alternating way, i.e. first take two points from trajectory one and one from trajectory two, then two points from trajectory two and one point from trajectory one, and so on. This custom triangulation is plotted in figure 3.4b for the artificial example. The custom triangulation looks more suited for the manifold than the Delaunay triangulation.

In constructing the triangulation of the manifold, it is important to take into account the uniqueness of the manifold. A manifold that is not unique will likely cause problems for the lookup tables, because a certain combination of PC scores might give two possible



(a) Condition not met (b) Condition met

Figure 3.3: Illustration of the Delaunay criterion. In (a) the circumcircle of triangle ABD is given. Point D lies inside the circumcircle, so the Delaunay condition is not met. In (b) the circumcircle of triangle ABC is given, which does not contain point D, so the conditions is met.



(a) Delaunay (b) Custom

Figure 3.4: Triangulations of the manifold with range scaling. The Delaunay triangulation in (a) shows large triangles, which are undesirable. The custom triangulation in (b) does not show any large triangles.

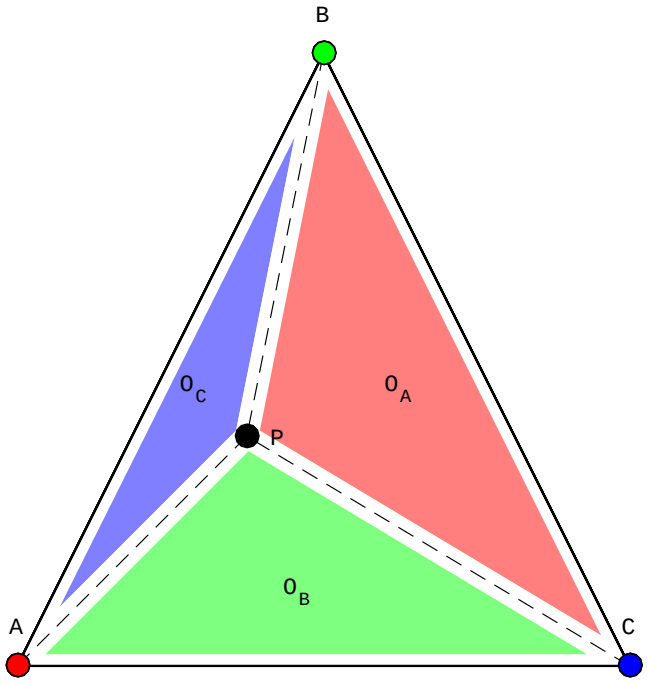


Figure 3.5: Example of the linear interpolation using a triangle ABC. Point P is looked up. The area O_A is the area of the triangle that does not include A, i.e. BCP.

values for the densities of PC sources. In this case, the manifold must be split into unique parts as was described in section 3.3.

The lookup table is ready to be used after constructing a triangulation. Suppose we want to look up a density for given PC scores z_1 and z_2. In the first step, the triangle is found in which this point lies. After that, the linear interpolation is done in this single triangle, which is illustrated by the example in figure 3.5. Suppose we have a triangle with vertices (x_A, y_A), (x_A, y_A) and (x_C, y_C) with values z_A, z_B and z_C. We want to look up the value z_P for point (x_P, y_P), which is done in two steps. Firstly, the areas of triangles ABP, ACP and BCP are calculated. After that, the value of z_P is found by:

$$z_P = \frac{z_A O_A + z_B O_B + z_C O_C}{O_A + O_B + O_C}, \tag{3.21}$$

where O_A, O_B and O_C are the triangle areas from figure 3.5. Equation (3.21) is evaluated for every single density and PC source that is looked up.

3.4.3 Using nonlinear regression

The third method for the reconstruction of the densities and PC sources is nonlinear regression, which has recently been proposed by [58, 59]. This part was done in cooperation

with Luca Vialetto, who studied this topic as a part of this master thesis under supervision of Dr Emile Carbone. The general goal of regression analysis is to find a relationship between variables, where only the values of the variables are known. In many cases a relationship needs to be found between one dependent variable and one or more independent variables. This becomes very challenging when many independent variables are present. The mathematical aim is to construct a function $\hat{f}(x_1, \ldots, x_n)$ that approximates:

$$y = f(x_1, \ldots, x_n) + \varepsilon, \tag{3.22}$$

where y is the dependent variable, $x_1, \ldots, x_n$ are the independent variables, f is a deterministic function of $x_1, \ldots, x_n$ and ε is the stochastic component. A familiar example is ordinary least squares, which is a parametric method that fits a predefined function of several parameters to some data. The disadvantage of such parametric regression is that some prior knowledge is needed about the data in order to choose an appropriate regression function. This is particularly difficult if used for the reconstruction of the densities and PC sources, because it is hard to identify in advance a functional relationship between a density and the PC scores. Therefore, we use a non-parametric regression method called MARS that determines the regression function directly from the data. MARS is an acronym for Multivariate Adaptive Regression Splines and was developed by Jerome Friedman in 1991 [60]. MARS builds a model in tho phases: the forward and the backward phase. The forward phase starts with the intercept term, which is the mean of the dependent variable. MARS then adds certain pairs of basis function (splines) that maximizes the reduction of the sum of the squared error. The added basis functions consist of a basis function that is already in the model (including the intercept) multiplied by a new basis function. In this way, it is possible to build a very accurate model by adding an excessive amount of basis functions. The large models that are build in the forward phase might be overfit. The backward phase deals with this by pruning the model to a proper size so that it is not overfitted. This is done by deleting the least important basis function, i.e. the basis function that reduces the error the least. This model is stored and the deletion process is continued until only the intercept term is remaining. From all of these pruned models the model with the lowest generalized cross-validation (GCV) is selected as the final model [61]. The GCV is calculated by:

$$\mathrm{GCV}(M) = \frac{1}{N} \sum_{i=1}^{N} \left[y_i - \hat{f}_M(x_i) \right]^2 \Bigg/ \left[1 - \frac{\mathrm{enp}(M)}{N} \right]^2, \tag{3.23}$$

where $\hat{f}_M$ is the regression function with M basis functions. The GCV criterion consists of the averaged-squared residual of the regression function to the data (numerator), together with a penalty to account for the size of the model (denominator). The effective number of parameters function is calculated as:

$$\mathrm{enp}(M) = M + \frac{1}{2} c(M - 1), \tag{3.24}$$

with $\frac{1}{2}(M-1)$ the amount of knots and c the GCV penalty per knot. The knots are related to the basis functions, which are described next.

There are two types of basis functions that MARS offers: piecewise linear and piecewise cubic functions. The piecewise linear functions have the following form:

$$b(x|s=+1,t) = \begin{cases} 0 & x \leq t \\ x-t & x > t \end{cases} \tag{3.25}$$

and

$$b(x|s=-1,t) = \begin{cases} -(x-t) & x < t \\ 0 & x \geq t \end{cases} \tag{3.26}$$

with t the knot location. The knots represent the locations of the discontinuities of the piece-wise functions. Here, $s = \pm 1$ is a parameter that determines on which side of the knot the basis function is zero. The piecewise-linear basis function can also be written in a more compact form: $b(x|s,t) = \max(s(x-t))$. The main disadvantage of the piecewise linear basis functions is that they do not have continuous first derivatives. Therefore, we also consider piecewise cubic basis functions, which have continuous first derivatives but do not have continuous second derivatives. The cubic functions are of the form:

$$C(x|s=+1,t_-,t,t_+) = \begin{cases} 0 & x \leq t_- \\ p_+(x-t_-)^2 + r_+(x-t_-)^3 & t_- < x < t_+, \\ x-t & x \geq t_+, \end{cases} \tag{3.27}$$

and

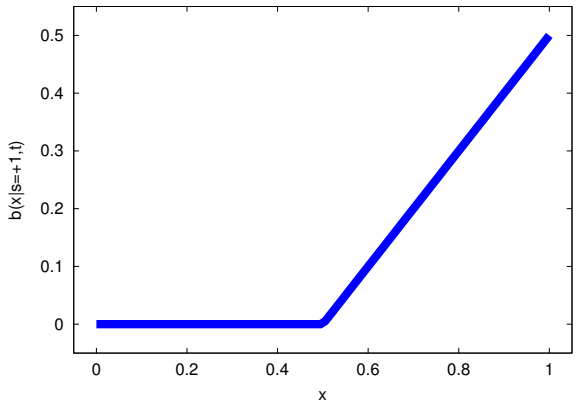
$$C(x|s=-1,t_-,t,t_+) = \begin{cases} -(x-t) & x \leq t_-, \\ p_-(x-t_-)^2 + r_-(x-t_+)^3 & t_- < x < t_+, \\ 0 & x \geq t_+, \end{cases} \tag{3.28}$$

where t_+ and t_+ are the lower and upper knot and t the central knot. The continuity of the first derivative of $C(x|s,t_-,t,t_+)$ is achieved by setting

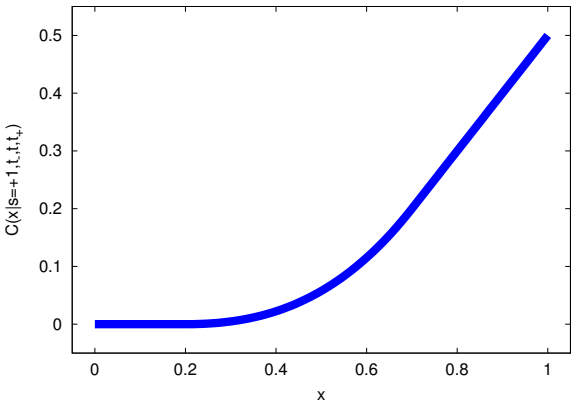
$$\begin{aligned} p_+ &= (2t_+ + t_- - 3t)/(t_+ - t_-)^2, \\ r_+ &= (2t - t_+ - t_-)/(t_+ - t_-)^3, \\ p_- &= (3t - 2t_- - t_+)/(t_- - t_+)^2, \\ r_- &= (t_- + t_+ - 2t)/(t_- - t_+)^3. \end{aligned} \tag{3.29}$$

Figure 3.6 shows an example of the piecewise linear and piecewise cubic basis functions for both values of s with the central knot $t = 0.5$ and side knots at $t_- = 0.2$ and $t_+ = 0.7$. The piecewise cubic basis functions are smooth and therefore usually preferred over the piecewise linear basis functions.

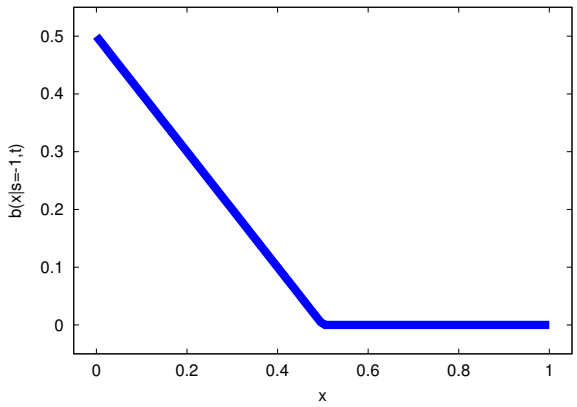
The software that was used to construct MARS models is called ARESLab, a MATLAB toolbox developed by Jekabsons [61]. The term "MARS" is a registered trademark and



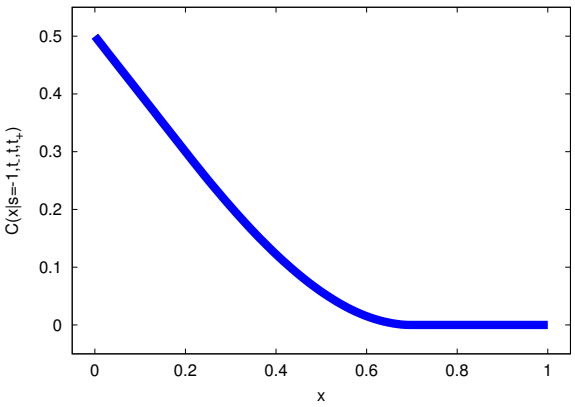
(a) Linear, $s = +1$



(b) Cubic, $s = +1$.



(c) Linear, $s = -1$.



(d) Cubic, $s = -1$.

Figure 3.6: Comparison of the piecewise linear and piecewise cubic basis functions, with central knot $t = 0.5$ and side knots $t_- = 0.2$ and $t_+ = 0.7$.

thus other names like "EARTH" or "ARES" are used instead of "MARS". For this reason, MARS models will from here on be referred to as ARES models.

In order to find a good regression function for the densities and PC sources, depending on at least two PC scores, it is important to control the size, complexity and accuracy of the ARES model. In the ARESLab toolbox, there are several parameters that can be specified by the user to determine the size, complexity and accuracy of the ARES model. The most important ones are listed below:

- *cubic*: true if piecewise cubic functions should be used, false if piecewise linear functions should be used. Piecewise-cubic functions are generally preferred for smooth data with little noise.

- *maxFuncs*: the maximum amount of basis function in the model after the forward phase. This parameter is used to set a maximum size of the ARES model in terms of the maximum amount of basis functions after the forward phase. Note that in the backward phase basis functions are removed, so the final ARES model will contain less basis functions.

- *maxInteractions*: the maximum degree of interactions between input variables. The input variables interact when basis functions are present that contain products of piecewise functions of two different variables. For example, the model $y = 0.5b(x_1|1,1) - 0.2b(x_2|1,2)$ has a degree of interaction of 1, whereas the model $y = 0.4b(x_1|1,1) + 1.2b(x_2|-1,-2) - 0.2b(x_1|1,1.2)b(x_2|-1,0.5)$ has a degree of interaction of 2.

- *prune*: true if the backward phase should be done, false if not. The backward phase is necessary when large ARES models are built in order to prevent overfitting.

- *threshold*: one of the stopping criteria for the forward phase. The forward phase is stopped when adding a new basis function changes R^2 by less than the threshold value or when the threshold is greater than $1 - R^2$. This parameter determines the accuracy of the ARES model.

The parameter settings of the ARES model is essential for the performance of ARES. One might for example choose a very low value for the *threshold* parameter and a high value for the *maxFuncs* parameter, so that a very accurate regression function is found, but this will take a very long time. On the other hand, when a small model is built by choosing a high *threshold* value and a low *maxFuncs* value, errors will be larger.

3.5 Log transformation

One of the major findings in Peerenboom *et al.* was the use of a log transformation prior to PCA [11]. In their work, they implemented a PCA global model and tested it on a state-to-state kinetics model of CO_2. The model includes vibrationally excited states of several species. In equilibrium, the population of these excited states follow a Boltzmann

distribution, where the fraction of the population of two excited states is proportional to the exponent of the energy difference of the two levels:

$$\frac{n_2}{n_1} \propto \exp\left(\frac{-\Delta E}{k_{\mathrm{B}} T}\right), \tag{3.30}$$

with n_1 and n_2 the densities of the excited states and ΔE the energy difference of the two excited states. The consequence of this exponential behaviour is that the population of the excited states may vary by several orders of magnitude, also outside equilibrium. These large variations of the species densities are challenges that need to be dealt with. Therefore, the dynamic range of the densities was reduced by applying a log transformation on the training set. The log transformation improves the reconstruction of the species densities from the PC scores drastically. The major advantage of the log transformation for the reconstruction is that it guarantees that the reconstructed densities will be positive. This is not true in general for PCA without log transformation, because the PC scores are allowed to have positive and negative values. When the densities are recovered using the matrix multiplication of equation (3.20), a significant error is made that may result in negative densities. When lookup tables are used, the recovered densities will always be positive, whether a log transformation is used or not, because the densities are interpolated from positive densities. This is not necessarily true when extrapolation is used, but extrapolation is not considered in this work. For this reason, one might argue that a log transformation is not necessary, when lookup tables are used to recover the species densities and PC sources. Peerenboom *et al.* [11], however, found that the log transformation improved the manifold shape. After the log transformation, the manifold was much smoother and well separated. Hence, a log transformation should also be used when the back transformation is done with lookup tables.

The consequence of the log transformation for the Global Model is that equation (3.5) needs to be rewritten in terms of $\log(\boldsymbol{n})$:

$$\frac{\partial \log(\boldsymbol{n})}{\partial t} = \boldsymbol{S} \oslash \boldsymbol{n} \tag{3.31}$$

Here, the $\oslash$ sign represents a Hadamard division. For the Hadamard division $\boldsymbol{C} = \boldsymbol{A} \oslash \boldsymbol{B}$, the elements of $\boldsymbol{C}$ are calculated as $C_{ij} = A_{ij}/B_{ij}$. The source term (3.15) for the PC continuity equations changes in the same manner:

$$\boldsymbol{S_Z} = (\boldsymbol{S} \oslash \boldsymbol{n})\, \boldsymbol{D}^{-1} \boldsymbol{A}. \tag{3.32}$$

The use of a log transformation before applying PCA is not completely new. Some research areas deal with variables that are believed to follow a log-normal distribution, for example the chemical composition of archaeological artefacts in archaeometric applications [62]. In such cases, the normality of the data is desirable and achieved by a log transformation. It is also argued in [62] that the log transformation tends to stabilize the variance of the variables and therefore gives them equal importance, similar to scaling.

Another field where the log transformation is used is allometry, which studies the size and shape of organisms during their growth [63]. Many of the relative growth studies have been related to the application of the allometry equation [64],

$$y = bx^{\alpha}, \tag{3.33}$$

with y a biological variable, such as the length of an organ, x the physical measure of the organism, α the scaling exponent of the power law and b a proportionality constant. It is common here to take the logarithm of x and y before PCA, because this transform equation (3.33) into a simple linear relationship. This transformation fits well with the linear nature of PCA.

In some cases, a log transformation is applied on compositional data, which consists of variables that normalized such that the sum of the variables equals one. Examples include molar and mass fractions. It appears that these constraints cause difficulties, because it gives a bias towards negative values among the correlations of the variables, so that an independent set of compositional variables will not have zero correlations between them [22]. Therefore, Aitchison [65] suggests to apply a log transformation of the variables divided by their geometric mean.

3.6 An alternative reduction method: ILDM

The Intrinsic Low-Dimensional Manifold (ILDM) method is an alternative reduction method to PCA. ILDM is based on the fact that a chemical system of reactions has widely varying time scales [66]. It is able to accurately model a chemical system for a certain time range, which can be specified by the user. For example, if the user is only interested in the long time scales of the system and is not interested in the initial disturbances of the system, then the system can be described by only the slow processes and the fast processes are assumed to be infinitely fast. It is also possible to neglect the slow processes and retain the fast processes, but this will not be considered in this work. One advantage of ILDM is that the time scales are directly calculated from the mathematical description of the chemical system through the sources, as in equation (3.9). The time scales are calculated by a diagonalization of the Jacobian matrix of the source term $\boldsymbol{S}$:

$$\boldsymbol{J} = \boldsymbol{A}\boldsymbol{L}\boldsymbol{A}^{-1}, \tag{3.34}$$

where $\boldsymbol{A}$ contains the eigenvectors of the Jacobian $\boldsymbol{J}$ and $\boldsymbol{L}$ is a diagonal matrix containing the eigenvalues of the Jacobian. The timescales τ_i are calculated from the eigenvalues λ_i:

$$\tau_i = \frac{1}{|\mathrm{Re}(\lambda_i)|} \tag{3.35}$$

Based on these time scales, the user can decide which time scaled should be included and which time scales should not be included. One can for example set a cutoff time scale

Table 3.1: Differences between PCA and ILDM.

	PCA	ILDM
Manifold	Correlations	Time scales
Input	Training set	Math. description of chemistry
Reduction	Variance	Time scales
Usability	Manifold shape	Stiffness

below which the time scales are considered to be fast and above which the time scales are slow.

One of the most important aspects of ILDM is the manifold. The manifold is the low-dimensional space that, in our case, describes all slow processes present in the system. The manifold is calculated by imposing that the fast processes are perpendicular to the manifold and thus the slow processes are parallel to the manifold. The manifold is described by a limited number of parameters, which is equal to the amount of time scales incorporated in the manifold. The amount of parameters depends on the cut off time scale. A reduced simulation is run by only solving the continuity equations for the parameters:

$$\frac{\partial \boldsymbol{n}_p}{\partial t} = \boldsymbol{S}_p(\boldsymbol{n}(\boldsymbol{n}_p)), \tag{3.36}$$

with $\boldsymbol{n}_p$ the species densities of the parameters and $\boldsymbol{n}$ the densities of all species. The source terms of the parameters $\boldsymbol{S}_p$ do not just depend on the parameters $\boldsymbol{n}_p$, but also on the densities of other species. Therefore, it is necessary to recover the densities of all species $\boldsymbol{n}$ using a lookup table, parametrized by the parameters $\boldsymbol{n}_p$.

In order to determine the manifold, it is necessary to do an initial guess of a point that lies on the manifold. The equilibrium point is guaranteed to lie on the manifold, because all possible initial conditions will move towards the equilibrium point. The equilibrium point is the combination of species densities for which the source terms are zero, and thus the species densities are constant. The equilibrium point is found by simply running the full global model until the equilibrium point is reached.

The most important fundamental differences between PCA and ILDM are summarized in table 3.1. The first one is related to the manifold: for PCA the manifold is determined by looking at the correlations of the species densities, whereas for ILDM the manifold is based on the slow (or fast) time scales. Both methods require some input: PCA demands a training set for the calculation of the PCs, whereas ILDM only needs the mathematical description of the chemical system, given by the set of reactions present in the plasma. Hence ILDM does not depend on the availability of a training set. The success of PCA mainly depends on the manifold. If the manifold has a nice shape, then PCA is likely to give a good result. ILDM is very useful when the system is stiff, i.e. the ratio of the slowest time scale to the fastest time scale is high. It tends to fail when many parameters are needed to model the slow timescales, since this results in high-dimensional lookup table.

Chapter 4

Verification

In this chapter, we will verify the developed PCA global model by comparing our results with the results of the model of [11]. First, a short description of the generation of the training data is given. After that, the reconstruction of the species densities from the PC scores is studied. Next, the shape of the manifold and results of the PCA global model are compared. This chapter also serves to demonstrate some of the important aspects of the PCA global model, which are the manifold, the reconstruction of the species densities and the influence of the log transformation as described in chapter 3.

4.1 Training data generation

The training data for the PCA global model was generated by a CO_2 plasma model [11] that includes state-to-state vibrational kinetics of CO_2 and CO. ZDPlasKin [67] was used for the description of the plasma chemistry, coupled with the Boltzmann solver BOLSIG+ [68] and a solver for the set of differential equations. The chemistry of the model contains 56 species, summarized in table 4.1. The vibrational levels of CO_2 consist of 21 asymmetric modes and the lowest 4 grouped effective symmetric modes. Three types of reactions are included in the model: electron impact reactions, heavy particle reactions that change the vibrational energy and heavy particle reactions that form or destruct species. These reactions are discussed by Peerenboom *et al.* in [11] and in even more detail in [9]. Most of the 1683 reactions describe the state-to-state interaction of the vibrational levels. For the model, constant gas pressure, gas temperature, electron density and reduced electric field intensity E/N was assumed, where E is the electric field and $N = \sum n_i$ the total gas density. This model represents a gas with a constant temperature that flows through a homogeneous plasma and mimics as a first order approximation a microwave surfaguide discharge [69]. The plasma parameters were chosen in the typical range of experiments of microwave discharges [5]: The gas pressure was 100 Torr, the reduced electric field 50 Td with a frequency of 2.45 GHz and the gas temperature was 300 K. Seven simulations were performed with different fixed ionization degrees with values $1 \cdot 10^{-7}$, $2 \cdot 10^{-7}$, $5 \cdot 10^{-7}$, $1 \cdot 10^{-6}$, $2 \cdot 10^{-6}$, $5 \cdot 10^{-6}$ and $1 \cdot 10^{-5}$. The electric field was switched off as soon as the

Table 4.1: List of species that are included in the CO_2 model. The numbers between brackets are the number of excited states included. The vibrational modes of CO_2 consist of 21 asymmetric modes and the lowest 4 grouped effective symmetric modes.

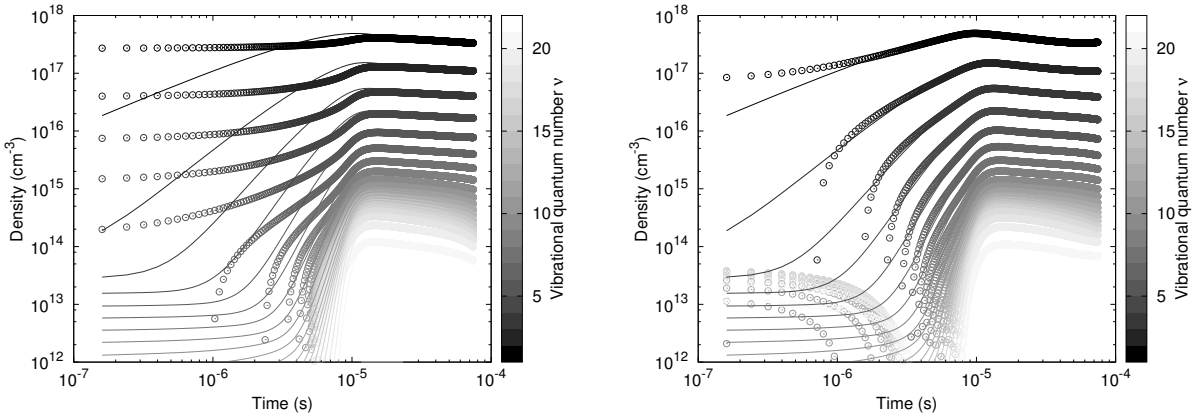
Neutral ground states	CO_2, CO, O_2, O_3, O, C_2O, C, C_2
Vibrational levels	$CO_2(25)$, $CO(10)$, $O_2(4)$
Electronic states	$CO_2(2)$, $CO(4)$, $O_2(2)$
Charged species	CO_2^+, e

specific energy input was 1 eV per molecule, which was found to be the optimal value for achieving high energy efficiency for the CO_2 conversion [11]. In the PCA calculations, only the parts of the simulations with high ionization degrees were included in the training set. The species densities of the seven runs with different ionization degrees were included in the PCA training set. Please note that we did not run the model ourselves, but the model output for the seven ionization degrees was given to us by Peerenboom. The species C_2 was removed from the training set, because its density was constant for each of the seven simulations. Including a constant variable in a training set limits the choice of scaling to 'level' scaling. All of the other scaling methods include the standard deviation or range, which is zero for a constant variable. This is problematic for the scaling using other methods.

4.2 A priori reduction

In the first part of the verification, we reconstruct the species densities from q PC scores and compare the results. We start with the most basic *a priori* reconstruction, without the log transformation and using relation (3.20) for the reconstruction. Results for 5 and 10 PCs using range scaling are shown in figure 4.1. Here, the densities of the asymmetric vibrational levels of CO_2 (solid lines) are compared with the reconstructed densities (circles). The lowest asymmetric mode of CO_2 is given in black and higher modes are given by lighter colours. The reconstruction for $t > 10^{-5}$ s is accurate, whereas the reconstruction for $t < 10^{-5}$ s is poor. Even unphysical negative densities are generated, which is seen by the circles running off the figure. The reconstruction becomes more accurate by increasing the amount of PCs, since this increases the fraction of the total variance that is accounted for, see equation (2.40), but the errors remain significant. Ideally, the mixture is described by a low amount of PCs, thus increasing the amount of PCs is not a viable option.

The reconstruction is improved by applying a log transformation prior to PCA, which decreases the dynamic range of the species densities. Another benefit of the log transformation is that negative densities are prevented. Figure 4.2 shows the reconstruction of the asymmetric stretch mode levels of CO_2 with a log transformation. The densities remain physical for the entire time range and the errors are greatly reduced, but not completely. Even these small errors are still problematic because of the strong nonlinearity of the



(a) 5 PCs. (b) 10 PCs.

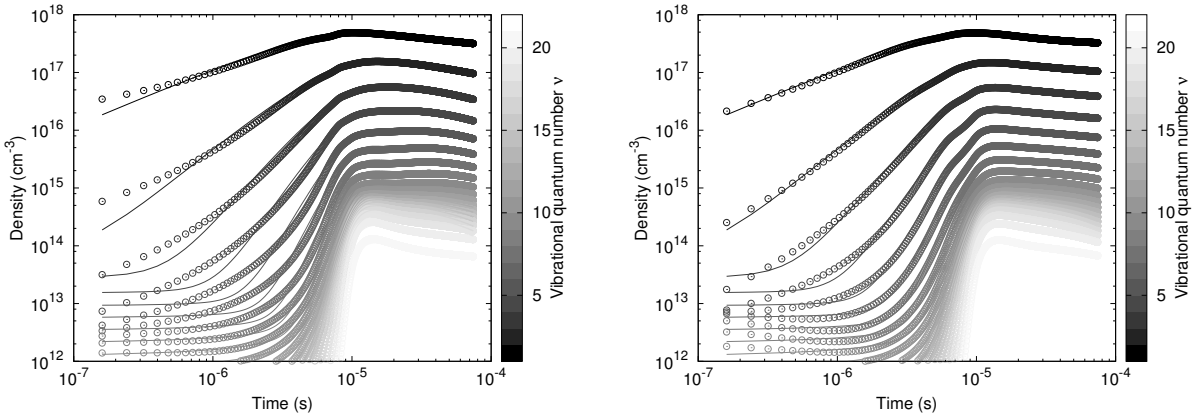
Figure 4.1: Reconstruction of the asymmetric modes (circles) without a log transformation. The original densities are given by the solid lines. Darker colours correspond to low vibrational quantum numbers. Range scaling was used.

species source terms $\boldsymbol{S_n}$, which need to be evaluated in the PC continuity equations. Due to the log transformation, it is necessary to take the exponent of the species densities to calculate $\boldsymbol{S_n}$, which significantly decreases the accuracy of the PC sources. Therefore, we will use linear interpolation from lookup tables for the reconstruction of both the species densities and the PC sources. An alternative for the reconstruction are nonlinear regression techniques, which are applied for this model in chapter 6.

In the lookup tables, the species densities are stored as a function of the first two PC scores. Details about the lookup tables and the linear interpolation are found in section 3.4.2. In this chapter we use the MATLAB scatteredInterpolant class that uses the Delaunay triangulation for the lookup tables, because the results that we found using this class closely match the results found by Peerenboom *et al.* in [11] and other classes were not suited. In figure 4.3, the reconstruction of the asymmetric vibrational level using relation (3.20) is compared with the reconstruction using lookup tables. The reconstruction using the lookup tables is much more accurate than the reconstruction using equation (3.20). The accuracy of both reconstruction methods used in figure 4.3 are quantified in [11] by R^2 statistics:

$$R^2 = 1 - \frac{\sum_i (y_i - y_{\text{new},i})^2}{\sum_i (y_i - \bar{y})^2}, \tag{4.1}$$

with y_i the known values, $\bar{y}$ the mean of the values and $y_{\text{new},i}$ the predicted values. The values of R^2 that were found in the present work are compared with the values reported by [11] in table 4.2, showing good agreement. Figure 4.4 shows the variance of the PCs, given by the eigenvalues of the covariance matrix. Two PCs account for more than 90 % of the variance of the training set.

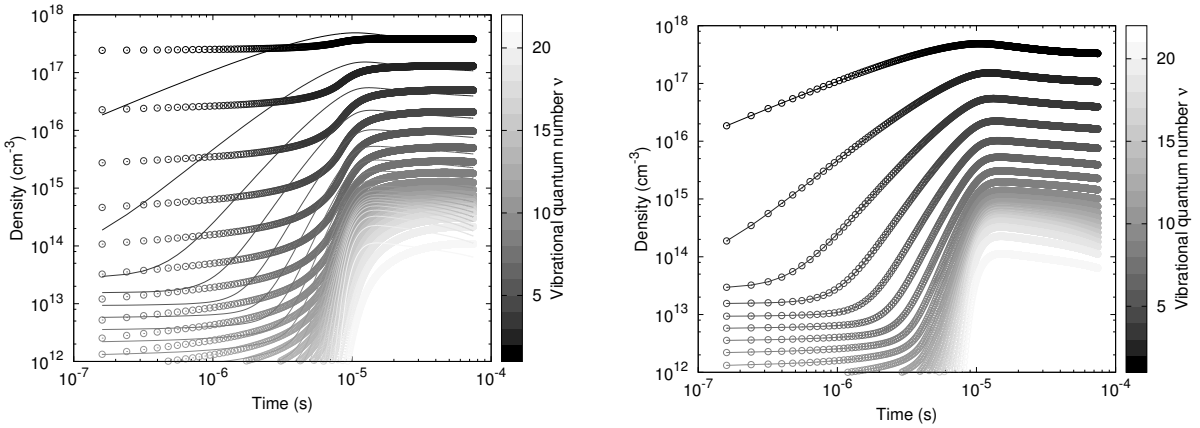


(a) 5 PCs. (b) 10 PCs.

Figure 4.2: Reconstruction of the asymmetric modes (solid) with a log transformation. The original densities are given by the solid lines. Darker colours correspond to low vibrational quantum numbers. Range scaling was used.

Table 4.2: Comparison of the reconstruction using (3.20) and linear interpolation from a lookup table using the R^2 statistics for the log of some species. A log transformation and 'range' scaling were used.

Species	Reconstruction present work	Reconstruction [11]	Lookup table present work	Lookup table [11]
CO_2	0.9256	0.9255	1	1
CO	0.9889	0.9889	1	1
O_2	0.9984	0.9984	1	1
O_3	0.9596	0.9596	1	1



(a) Relation (3.20)

(b) Lookup

Figure 4.3: Recovery of the time evolution of the asymmetric vibrational modes of CO_2 using (a) relation (3.20) and (b) and interpolation from a lookup table. Log transformation and 'range' scaling method are applied.

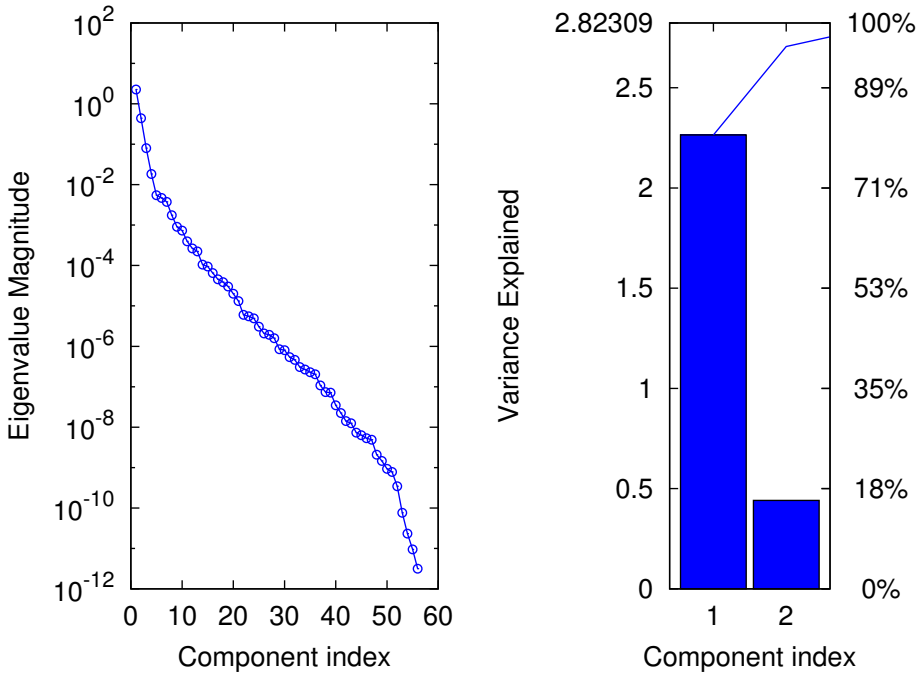
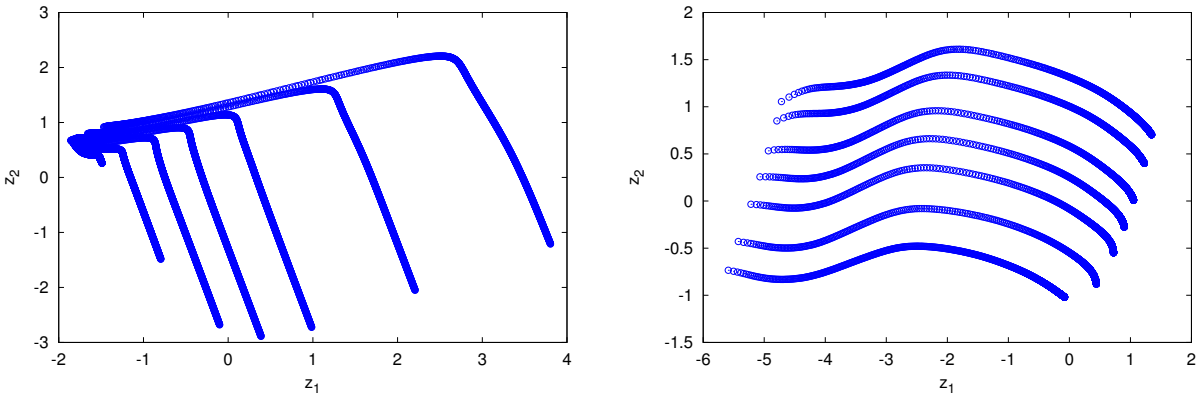


Figure 4.4: Eigenvalues of the covariance matrix and the percentage of explained variance. A log transformation and 'range' scaling are employed.



(a) Without log transformation

(b) With log transformation

Figure 4.5: The manifold. The PC scores of the training set are given by the blue circles. The 'range' scaling method is applied. In (b) a log transformation is applied, which results in smoother and well-separated than in (a), where no log transformation was used.

4.3 Manifold

In this section, we will have a closer look at the manifold. As was explained in section 3.3, the manifold represents the lower dimensional space that shows the time dependent trajectories of the plasma models of seven different ionization degrees that were included in the training set. The manifold is shown by plotting the first two PC scores of the training set. First, the influence of the log transformation on the manifold is demonstrated. Figure 4.5 shows both the manifold without the log transformation in 4.5a and the manifold with log transformation in 4.5b. The blue circles represent the PC scores of the first PC on the horizontal axis and the PC scores of the second PC on the vertical axis. There are seven trajectories that correspond to the seven different ionization degrees that are present in the training set. The manifold without log transformation is problematic in several aspects: the trajectories have sharp bends and are not well separated. The manifold with a log transformation does not have these problems, since the trajectories are well separated, non-overlapping, and does not contain large gradients, which is essential for the performance of PCA as was explained in chapter 3.

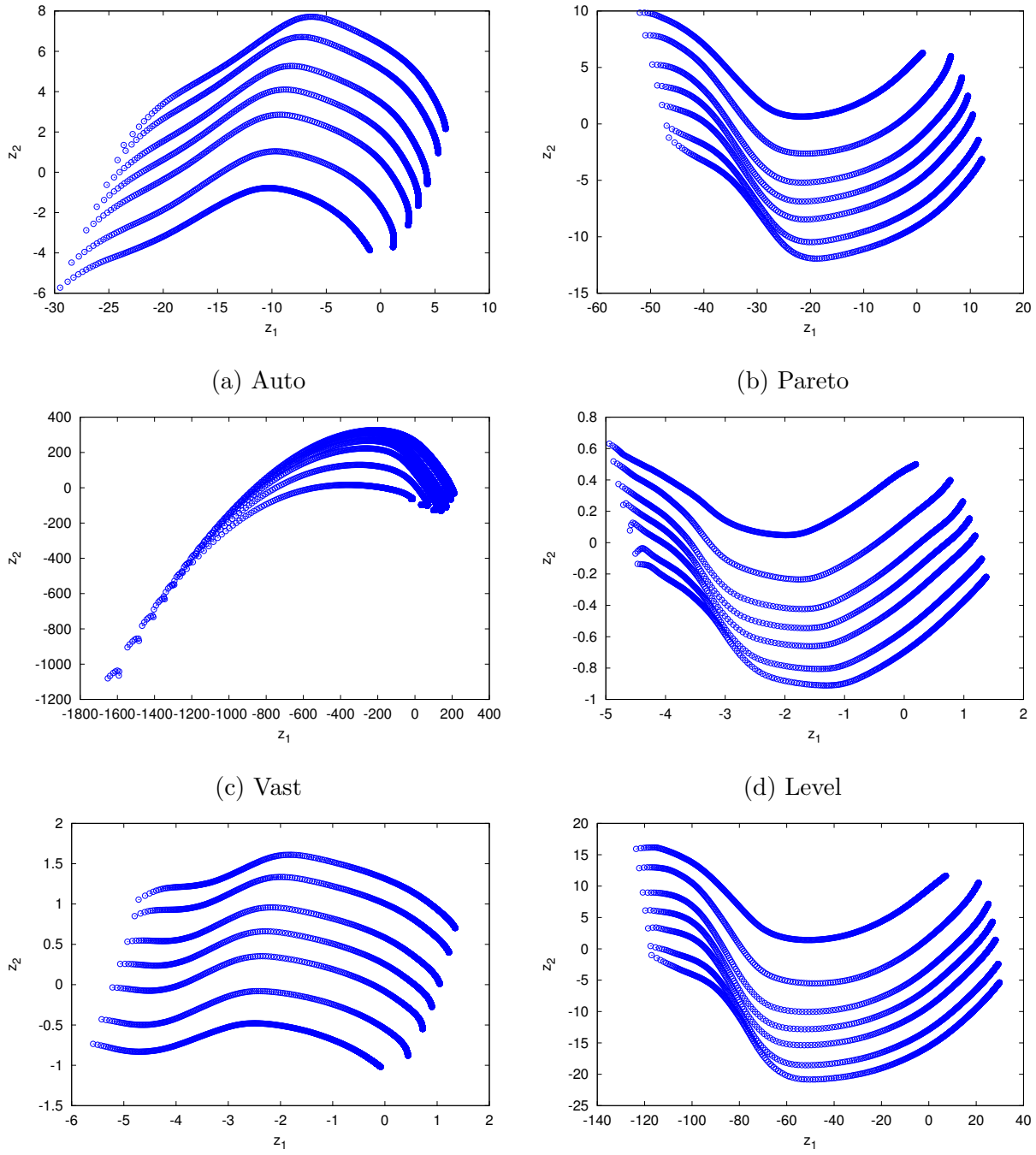
Besides the log transformation, the shape of the manifold strongly depends on the scaling of the input data. Figure 4.6 shows the manifolds for the different scaling methods, listed in table 2.3. Some of the scaling methods give similar trajectories in this example, such as for 'pareto', 'level' and without scaling. All the scaling methods give a unique manifold, except for 'vast' scaling. The uniqueness of the manifold is very important for the lookup tables, because it is necessary that the species densities can be associated with single values of the PC scores. Therefore, 'vast' scaling cannot be used in this case for the PCA global model. The manifold plots of the present work shown in figure 4.6, closely resemble the ones given in [11], but there is one difference. For some scaling methods, the

plots are flipped. This difference is caused in the calculation of the PC coefficient matrix $\boldsymbol{A}$, that is calculated as the matrix that contains the eigenvectors of the covariance matrix $\boldsymbol{C}$. If $\boldsymbol{a}_i$ is an eigenvector of $\boldsymbol{C}$, then also $-\boldsymbol{a}_i$ is an eigenvector, so sign differences might exist for $\boldsymbol{a}_i$. This sign difference is also visible in the manifold, since the PC scores are calculated by projecting the data on the first q eigenvectors $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_q$. A similar sign ambiguity exists when the PCs are calculated using SVD, which is described in [70]. The manifolds of both works are identical when taking into account this sign ambiguity. The 'range' scaling methods gives the best manifold, because of small gradients, well-separated trajectories and uniqueness and is thus expected to give the best results in the next section, where we will solve the PC continuity equations.

4.4 A posteriori reduction

So far, we have only reconstructed the data that was already in the training set, but we also want to generate new data by solving the PC continuity equations. As a first step, we try to reproduce the data for an ionization degree of $1 \cdot 10^{-6}$ by solving the PC continuity equations with the initial condition for this ionization degree. The log transformation, 'range' scaling and lookup tables were used in the calculations. In every time step, the PC source terms were recovered from two lookup tables using the first two PCs. The species densities were recovered after the simulation was finished. Figure 4.7b shows the results of the PC global model for an ionization degree of 10^{-6}. The densities of the asymmetric vibrational modes of CO_2 are plotted as a function of time. The results calculated from the PCA model are denoted by circles, whereas the results from the full model are given by the solid lines. The vibrational levels of CO_2 calculated from the PC continuity equations are in close agreement with the densities that were calculated from the full model. The manifold is shown in figure 4.7a. The blue circles are the PC scores of the training set and the red solid line represents the solution of the PC continuity equations. We see that the calculated PC scores correctly follow the middle reaction trajectory. This is expected thanks to the smooth, well-separated and unique manifold for 'range' scaling.

The same calculations were done for the 'level' and 'vast scaling methods, which have a less smooth manifold and that do not have a unique manifold. The solutions of the reduced model for 'level' scaling is given in figure 4.8. The plot of the manifold in 4.8a shows that the manifold is not as smooth for low values of $\boldsymbol{z}_1$. Also, the manifold is not as well-separated as 'range' scaling. Consequently, the PC solution does not completely follow the training set. These errors are also visible for the asymmetric vibrational modes of CO_2, which are shown in figure 4.8b. and small errors are visible for the asymmetric vibrational stretch modes. The result of the reduced model is less accurate for 'level' scaling than for 'range' scaling. The manifold for 'vast' scaling, together with the PC solution, is given in figure 4.9a. The solution in red does not follow the training with an ionization degree of $1 \cdot 10^{-6}$, but follows the training set with an ionization degree of FILL IN. This is due to the fact that the manifold on the bottom left is not unique. The errors of the calculated vibrational levels are large, see figure 4.9b. This shows that the choice of scaling



(a) Auto

(b) Pareto

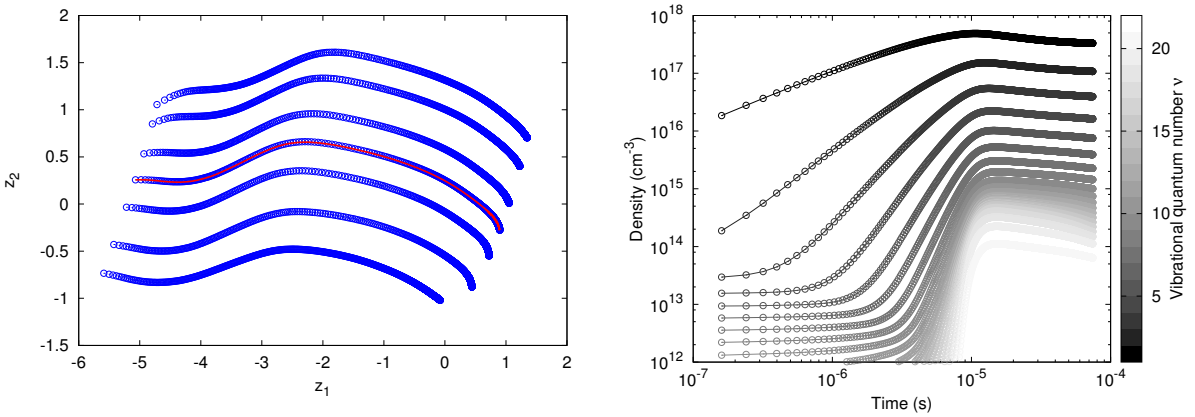
(c) Vast

(d) Level

(e) Range

(f) No scaling

Figure 4.6: The influence of the scaling method on the shape of the manifold. The blue circles are the PC scores of the training set. The trajectories are different due to the use of different scaling methods.



(a) Manifold

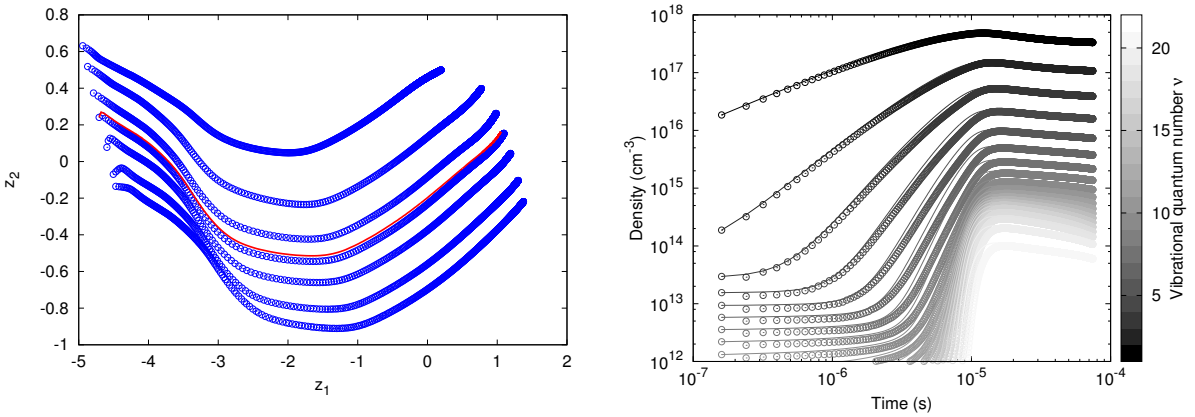
(b) CO_2 vibrational levels

Figure 4.7: Comparison of the full and reduced calculation for the 'range' scaling method. In (a) the PC scores of the training set is given by the blue circles and the reduced solution by the red line. In (b) the reconstructed densities of the asymmetric modes (circles) and the original densities (solid lines) are plotted.

is important in order to find an appropriate manifold and accurate results.

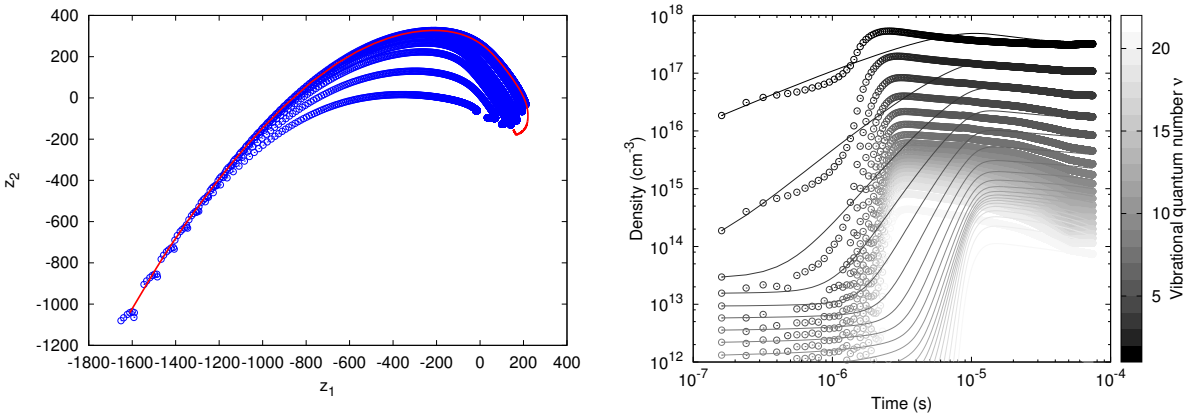
It is not surprising that the solution of the PC continuity equations is accurate when choosing an initial condition that was already in the training set. In this case, the calculated PC scores are very close to the manifold, such as in figure 4.7a, so apparently the error that is made by the linear interpolation is small. The linear interpolation in the lookup tables is more challenging when an initial condition is chosen that is not present in the training set. The new initial conditions are chosen by picking nine equally spaced PC scores between the first point of the 3rd and 4th, and 4th and 5th manifold trajectory. These initial conditions represent new ionization degrees. The PC model was solved until a time of $7.38 \cdot 10^{-6}$ s. This was the time at which the electric field was switched off for an ionization degree of 10^{-5} that is part of the training set. Results are shown in figure 4.10, where the blue circles are the PC scores of the training set and the red solid lines represent the reduced calculations. The reduced model trajectories follow the manifold shape well. Figure 4.10b shows the CO density after $7.38 \cdot 10^{-6}$ s as a function of the ionization degree. The blue squares are the CO ground densities of the training set after $7.38 \cdot 10^{-6}$ s and the red circles are the CO densities that were found for the reduced calculations. Clearly, there is a transition of the CO_2 conversion at an ionization degree of about 10^{-6}, which is explained by Peerenboom *et al.* [9, 11]. The reduced calculations give a smooth prediction of the CO density for ionization degrees that were not in the training set. Thus, the reduced calculations are accurate for ionization degrees that are in the range of the ionization degrees that were included in the training set.

Figure 4.10 also contains results of reduced simulations for ionization degrees that were not in the training set. These are the red trajectories that lie above the seven trajectories in the manifold. In this case, the scatteredInterpolant class performs linear extrapolation



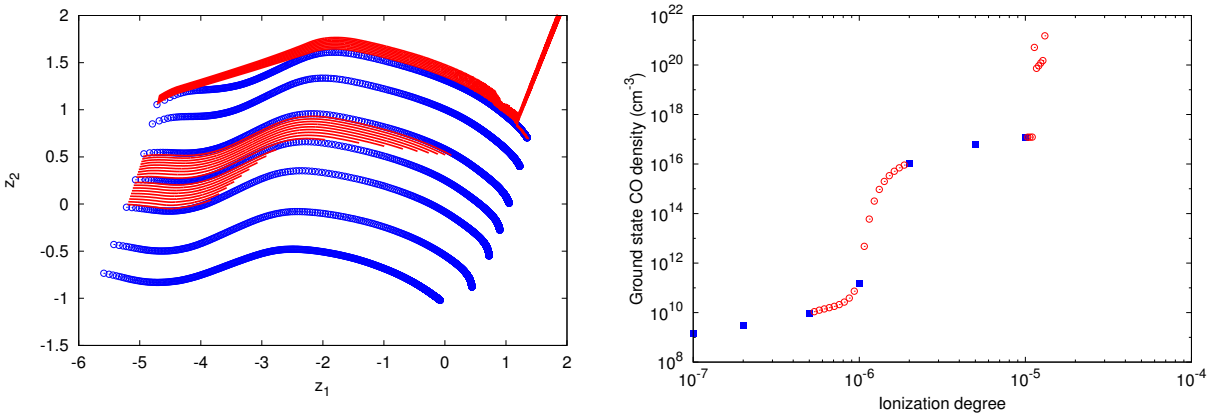
(a) Manifold (b) CO_2 vibrational levels

Figure 4.8: Comparison of the full and reduced calculation for the 'level' scaling method.



(a) Manifold (b) CO_2 vibrational levels

Figure 4.9: Comparison of the full and reduced calculation for the 'vast' scaling method.



(a) New initial conditions

(b) CO_2 conversion

Figure 4.10: Comparison of the full and reduced calculations for different ionization degrees. In (a) the manifold is plotted with the training (blue circles) and the solutions (red lines). Figure (b) gives the predicted CO_2 conversion using the reduced model in reduced and predictions from the full model are given in blue.

using the boundary points of the lookup table. The trajectories do not really follow the manifold shape at the beginning of the simulation and at the end some instabilities occur before the solution shoots off the manifold. The exact same behaviour is shown in the figure that Peerenboom *et al.* present in [11]. The consequence of this behaviour is that the predicted values for the CO_2 conversion are inaccurate. In order to explain the result, we plot the triangulation of the top part of the manifold, as is given in figure 4.11. We see that some triangles connect only points from the upper manifold trajectory. The boundary of this part of the triangulation consists of only one single triangle, thereby skipping all the points in between. This explains why the solution of figure 4.10a does not follow the manifold shape, but instead goes in a straight line.

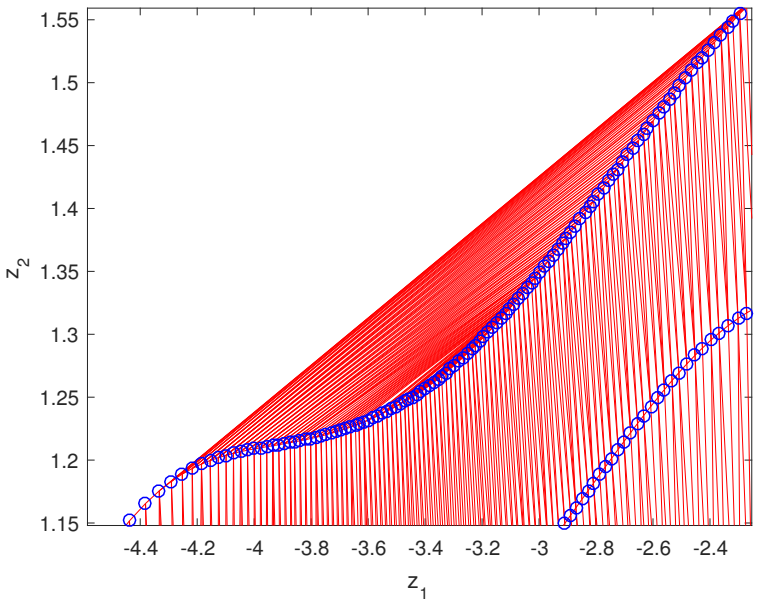


Figure 4.11: Delaunay triangulation of the top of the manifold.

Chapter 5

A comparison with ILDM

This chapter contains a comparison of the accuracy of PCA with the accuracy of another reduction method: Intrinsic Low Dimensional Manifold (ILDM). The comparison is done for three models: a molecular argon model with seven species, an argon model with 78 levels and a CO_2 microwave model.

5.1 Molecular argon model

5.1.1 Chemistry

The first test case for the comparison is done for a global model of molecular argon, developed in [71], containing seven species. Three grouped excited states of argon are considered in the model: the metastable levels $1s_3$ and $1s_5$ are grouped in a single species Ar^m, the resonant levels $1s_2$ and $1s_4$ are combined into species Ar^r, and finally the $3p^54p$ levels are used through species Ar(4p). The ionic species are given by Ar^+ and a molecular ionic species Ar_2^+. The electrons are calculated from charge neutrality, thereby assuming that the plasma is quasi-neutral. The complete set of reactions is given in table 5.1, including expressions for the rate coefficients and their references as was given by Rahimi [71]. The rate coefficient for reaction (M21) was reported incorrectly by Rahimi, since the unit conversion from cm^{-3} to m^{-3} was not done. The electron temperature is chosen to have a constant value of 32 400 K, because the ILDM model was only implemented for constant electron temperatures. The heavy particle temperature has a fixed value of 600 K. The model is solved by using the stiff ODE solver LSODA, developed by Hindmarsh and Petzold [72].

Table 5.1: Reaction set of the molecular argon model. Temperatures are in eV unless stated otherwise.

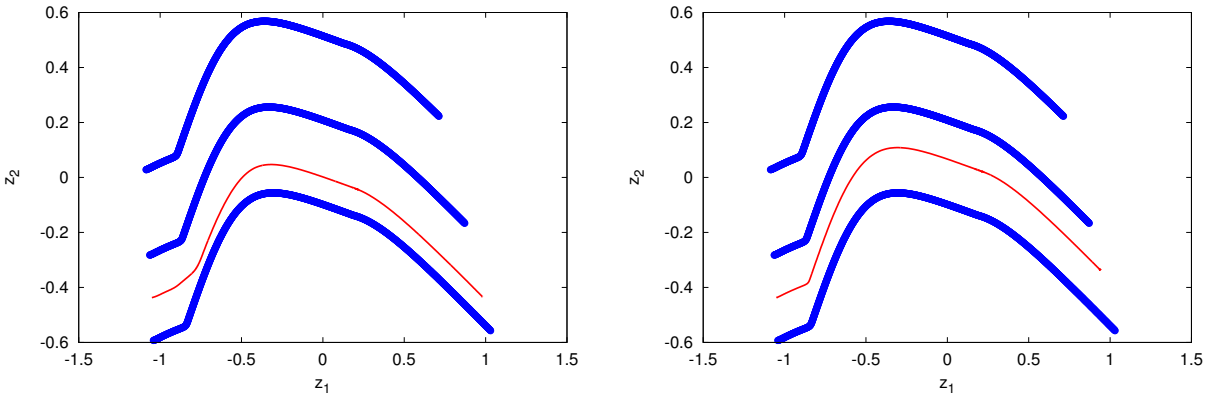
No.	Process	Rate coefficient	Reference
	Excitation from ground state		
(M1)	$Ar + e \rightarrow Ar^m + e$	$2.50 \cdot 10^{-15} T_e^{0.74} \exp(-11.56/T_e)$	[73]
(M2)	$Ar + e \rightarrow Ar^r + e$	$2.50 \cdot 10^{-15} T_e^{0.74} \exp(-11.56/T_e)$	[73]
(M3)	$Ar + e \rightarrow Ar(4p) + e$	$1.40 \cdot 10^{-14} T_e^{0.71} \exp(-13.20/T_e)$	[74]
	Deexitation		
(M4)	$Ar^r + e \rightarrow Ar + e$	$4.30 \cdot 10^{-16} T_e^{0.74}$	[73]
(M5)	$Ar^m + e \rightarrow Ar + e$	$4.30 \cdot 10^{-16} T_e^{0.74}$	[73]
(M6)	$Ar(4p) + e \rightarrow Ar + e$	$3.90 \cdot 10^{-16} T_e^{0.71}$	[74]
	Stepwise Excitation and Deexcitation		
(M7)	$Ar^r + e \rightarrow Ar(4p) + e$	$8.90 \cdot 10^{-13} T_e^{0.51} \exp(-1.59/T_e)$	[73]
(M8)	$Ar^m + e \rightarrow Ar(4p) + e$	$8.90 \cdot 10^{-13} T_e^{0.51} \exp(-1.59/T_e)$	[73]
(M9)	$Ar^m + e \rightarrow Ar^r + e$	$2 \cdot 10^{-13}$	[73]
(M10)	$Ar^r + e \rightarrow Ar^m + e$	$3 \cdot 10^{-13}$	[73]
(M11)	$Ar(4p) + e \rightarrow Ar^r + e$	$1.50 \cdot 10^{-13} T_e^{0.51}$	[73]
(M12)	$Ar(4p) + e \rightarrow Ar^m + e$	$1.50 \cdot 10^{-13} T_e^{0.51}$	[73]
	Direct ionization		
(M13)	$Ar + e \rightarrow Ar^+ + e + e$	$2.30 \cdot 10^{-14} T_e^{0.68} \exp(-15.76/T_e)$	[74]
	Stepwise ionization		
(M14)	$Ar^m + e \rightarrow Ar^+ + e + e$	$6.80 \cdot 10^{-15} T_e^{0.67} \exp(-4.20/T_e)$	[73]
(M15)	$Ar^r + e \rightarrow Ar^+ + e + e$	$6.80 \cdot 10^{-15} T_e^{0.67} \exp(-4.20/T_e)$	[73]
(M16)	$Ar(4p) + e \rightarrow Ar^+ + e + e$	$1.8 \cdot 10^{-13} T_e^{0.61} \exp(-2.61/T_e)$	[74]
	Dissociative Recombination		
(M17)	$Ar_2^+ + e \rightarrow Ar^r + Ar$	$0.6 \cdot 10^{-12} (T_e(\mathrm{K})/300)^{0.66}$	[75]
(M18)	$Ar_2^+ + e \rightarrow Ar^m + Ar$	$0.6 \cdot 10^{-12} (T_e(\mathrm{K})/300)^{0.66}$	[75]
	Three-body recombination		
(M19)	$Ar^+ + e + e \rightarrow Ar + e$	$8.75 \cdot 10^{-39} T_e^{-4.5}$	[76]
	Ion conversion		
(M20)	$Ar^+ + Ar + Ar \rightarrow Ar_2^+ + Ar$	$2.25 \cdot 10^{-43} (T_h(\mathrm{K})/300)^{-0.40}$	[77]
	Electron-impact dissociation		

No.	Process	Rate coefficient	Reference
(M21)	$Ar_2^+ + e \rightarrow Ar^+ + Ar + e$	$1.11 \cdot 10^{-12} \exp\left(-\frac{2.94-3(T_h-0.026)}{T_e}\right)$	[77]
	Atom-impact dissociation		
(M22)	$Ar_2^+ + Ar \rightarrow Ar^+ + Ar + Ar$	$5.22 \cdot 10^{-16} T_h^{-1.00} \exp(-1.304/T_h)$	[77]
	Radiation		
(M23)	$Ar^r \rightarrow Ar + h\nu$	$1 \cdot 10^5$	[73]
(M24)	$Ar(4p) \rightarrow Ar^r + h\nu$	$3.2 \cdot 10^7$	[73]
(M25)	$Ar(4p) \rightarrow Ar^m + h\nu$	$3 \cdot 10^7$	[73]

A training set was generated for the PCA model by running the full (global) model for three different initial pressures, namely 158 Pa, 316 Pa and 631 Pa. The pressure is not constant due to the presence of molecules and is thus expected to change. For the reduced calculations, an initial condition was chosen so that the initial pressure equals 223 Pa, which is not part of the training set. In chapter 4 it was shown that for an initial condition that is part of the training set, the calculated densities are very accurate, because errors due to the linear interpolation in the lookup tables are expected to be small. It is, however, more challenging to choose initial conditions in between the initial conditions of two models that are present in the training set, since in that case the errors due to the linear interpolation are expected to be largest. The PC continuity equations are solved using the MATLAB ode45 solver [78], which is based on the Dormand-Prince method, an explicit Runge-Kutta (4,5) formula [79]. A log transformation and the 'range' scaling method were used, because it gives a unique, well-separated and smooth manifold. In figure 5.1, a comparison is given between the solution of the PC continuity equations when the PC sources are recovered using lookup tables with the Delaunay triangulation and the custom triangulation. The blue circles represent the PC scores of the training set and the red solid line gives the solution of the PC continuity equations starting from the left. For the Delaunay triangulation the red line does not follow the manifold shape, but moves towards the bottom trajectory in blue. This is not the case for the custom triangulation, where the red line stays in between the two trajectories. The result for the custom triangulation clearly outperforms the result for the Delaunay triangulation.

5.1.2 Comparison

A comparison of the densities of Ar^r is given in figure 5.2 for the full model, the PCA model and the ILDM model. The result for the full model is given in blue, the PCA model in red and the ILDM model in green. Three regions are identified based on the result for the full model. The first region from $t = 1 \cdot 10^{-9}$ s to $t = 2 \cdot 10^{-8}$ s shows a decrease of the Ar^r density. The second region shows another decrease of the Ar^r density from $t = 2 \cdot 10^{-8}$ s



(a) Delaunay.

(b) Custom.

Figure 5.1: Manifold for the molecular argon model using 'range' scaling and a log transformation.

Table 5.2: Values of R^2 for the molecular argon model. The values for PCA using the custom triangulation are higher than for the Delaunay triangulation.

Method	Ar	Ar^r	Ar^m	Ar(4p)	Ar^+	Ar_2^+	e
PCA Delaunay	0.9452	0.9834	0.9821	0.9790	0.8984	0.9984	0.8982
PCA custom	0.9770	0.9992	0.9991	0.9989	0.9934	0.9966	0.9934
ILDM	-4.2543	-0.5498	-0.5639	-0.5866	-0.0431	-0.0399	0.2835

to $t = 1 \cdot 10^{-6}$ s. Equilibrium has been reached in the third region. The PCA model was able to reproduce the behaviour of the three regions, but the decrease of the density of Ar^r happens earlier than for the full model. The boundary of the training set is reached at $t = 4.86 \cdot 10^{-7}$ s. The result from the ILDM model only matches the full simulation after $t = 5 \cdot 10^{-7}$ s. ILDM is problematic for this chemistry, because the time scales all have similar values. The inclusion of only one time scale in the reduced simulation is therefore expected to give inaccurate results. This is also visible in figure 5.2. The error of all densities from the reduced simulations using PCA and ILDM with respect to the full model is quantified in table 5.2 using the coefficient of determination R^2. The error for PCA using the Delaunay triangulation are also included. The PCA calculations using the custom triangulation shows very good agreement with the full model, whereas the result using the Delaunay triangulation is not as good. The values of R^2 for ILDM are relatively low, but this was expected.

5.1.3 Including the electron energy

In the comparison for the molecular argon model in the previous section, only results for constant electron temperatures were considered, because the ILDM method was only

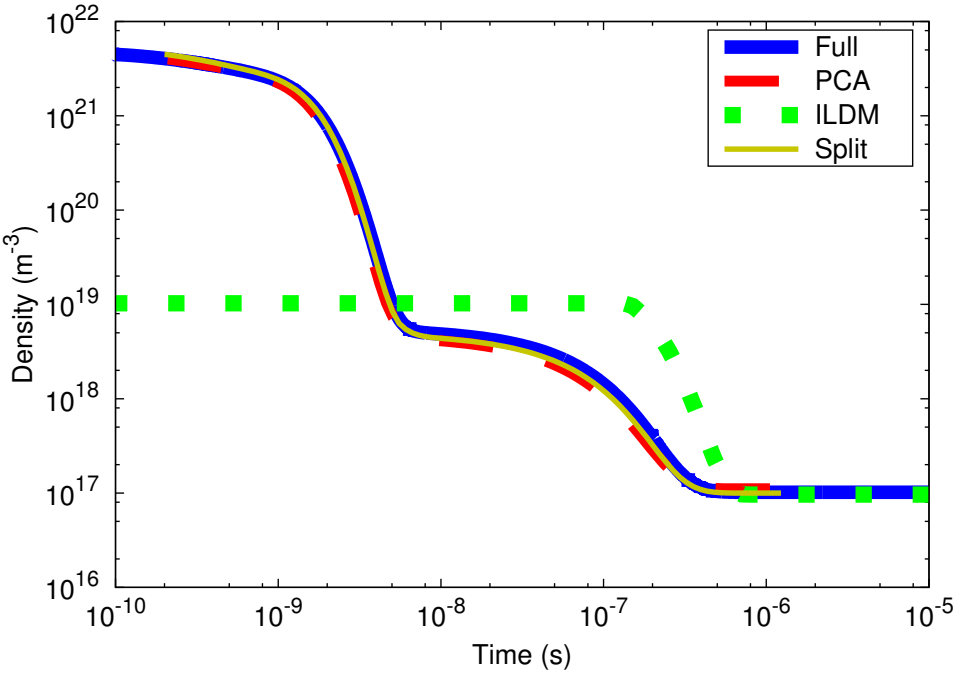


Figure 5.2: Results for the density of Ar^r as a function of time for the full model in blue, the PC model in red and the ILDM model in green. For the PC model, 'range' scaling, a log transformation and lookup tables with the custom triangulation were used.

implemented for constant electron temperatures. It is, however, still possible to do the comparison for varying electron temperature for the PCA model and the full model, which is discussed here. The parameters of the molecular argon model were kept the same. The initial value of the electron temperature was set at 32 400 K. First, the manifold, which is shown in figure 5.3a, is studied. The manifold is not unique and is therefore split into three separate parts, as described in section 3.4.2. The solution of the PC continuity equations are shown in figure 5.3, where the blue circles are the PC scores of the training set and the red line is the calculated solution. The solution follows the manifold trajectories very well. This shows that including the electron energy as a variable in PCA possibly gives a less smooth and not unique manifold, but this can be solved by dividing the manifold.

Figure 5.4 shows the density of Ar^r and the electron temperature as a function of time. Both figures show that there the PCA model agrees well with the full model. This shows that, despite keeping the electron temperature constant for the comparison with ILDM, PCA is capable of also calculating the electron temperature accurately.

5.2 Argon model with 78 levels

5.2.1 Chemistry

The second model that is used for the comparison is a global argon model with 78 levels, developed by Graef [53]. The list of species included in the model (apart from the ground states of Ar and Ar^+) is given in table 5.3. Two groups of species are formed based on the total angular momentum quantum number of the core of the Ar atom: $j_c = 3/2$ (un-primed) and $j_c = 1/2$ (primed). Most of the levels with high energies are described by combined levels, also called level blocks [53]. The blocks consist of all levels with the same principal and orbital quantum number and core configuration. Only the 4s, 5s, 4p, 5p and 3d levels (both primed and un-primed) with the same principal and orbital quantum numbers and core configuration are modelled as individual levels. The energies of the level blocks are calculated as the weighted average of the energies:

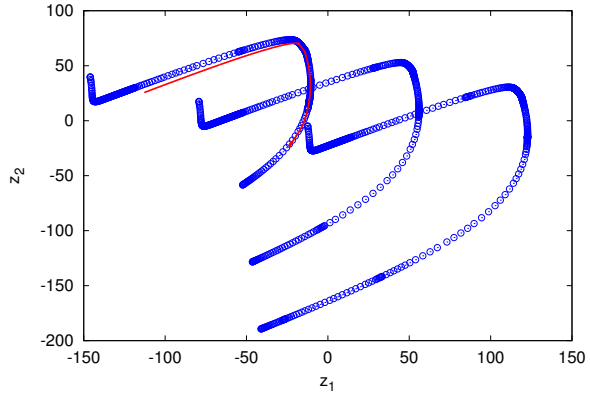
$$E_{\text{block}} = \frac{\sum_l g_l E_l}{\sum_l g_l}, \tag{5.1}$$

where E_l is the energy and g_l the statistical weight of level l, used in the block.

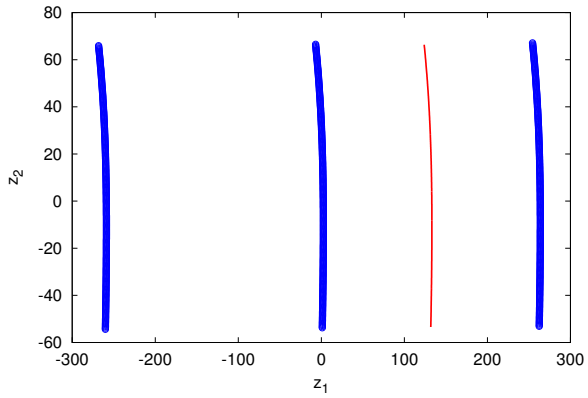
Two types of reactions are included in the model: radiative transitions and electron excitation. First, the radiative transitions will be treated. The Einstein coefficients for spontaneous emission were taken from the online NIST database [80]. Due to the use of blocks, special attention needs to be paid for transitions that originate from a level block (b_u):

$$A(b_u, l) = \frac{\sum_j g_j A(u_j, l)}{\sum_i g_i}. \tag{5.2}$$

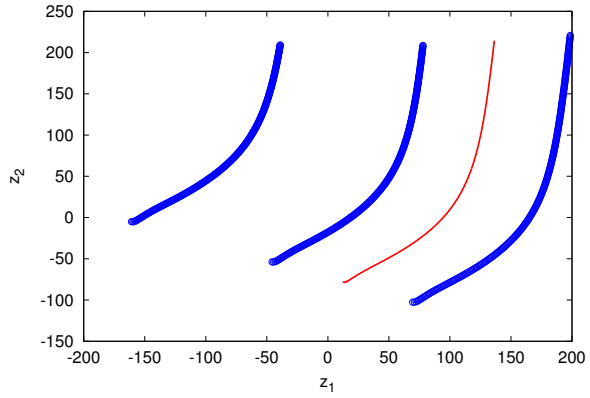
where in the numerator the summation is done over all levels u_j in block b_u with a transition to level l. If both levels of a transition are part of a block, then another summation is



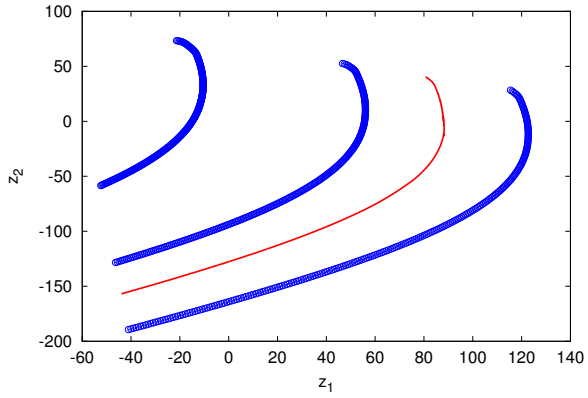
(a) Full manifold



(b) Part 1

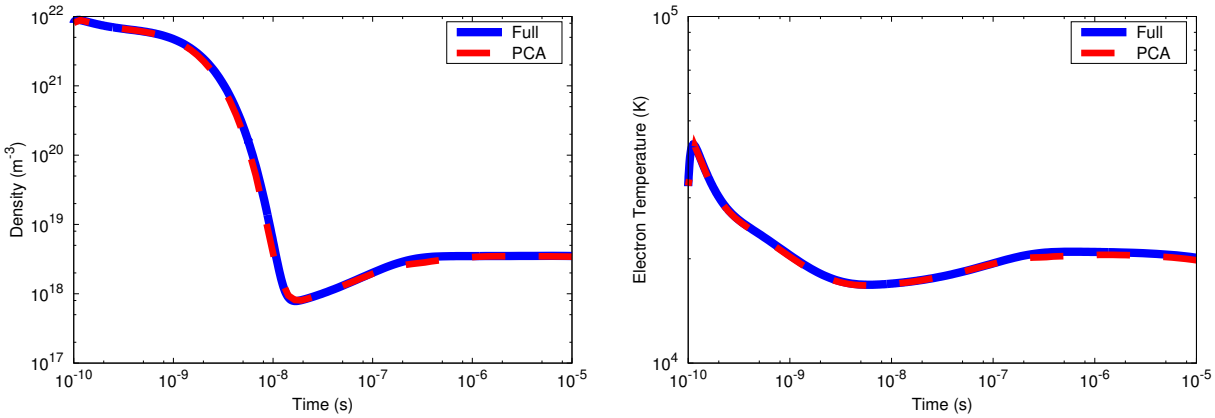


(c) Part 2



(d) Part 3

Figure 5.3: The solutions of the PC continuity equations for the molecular argon model, where the electron energy was included as a variable in the training set. The blue circles are the PC scores of the training set and the red lines are the solutions of the PC continuity equations. Figure (a) shows the full manifold, which is not unique and thus gives a bad solution. The training set can be described uniquely if it is split into three parts. Figures (b), (c) and (d) give the three parts into which the manifold is split together with the solution.



(a) Ar^r density (b) Electron temperature

Figure 5.4: Comparison of the density of Ar^r and the electron temperature as a function of time for the full model and the PCA model. Results for the full model are given in blue and results for the PCA model in red.

required over k:

$$A(b_u, b_l) = \sum_k \frac{\sum_j g_j A(u_j, l_k)}{\sum_i g_i}. \tag{5.3}$$

A complete list of the radiative transitions can be found in appendix B.

The electron excitation reactions are described by cross sections. The cross sections for excitation from the ground state are given by Yanguas-Gil *et al.* [81] and used as lookup tables. The cross sections were determined experimentally for excitation to the 4s, 4p, 3d, 5s and 5p levels. For some species the cross sections are given for blocks, whereas Graef models these levels individually. The cross section of a block σ_B can be split into cross section for individual levels σ_i using:

$$\sigma_j = \frac{g_j}{\sum_n g_n} \sigma_B \tag{5.4}$$

where the summation is done over all the levels used in the block.

The electron excitation processes to higher levels are reported by Vlček [82] as semi-empirical cross sections, through fit parameters for Drawin cross sections. Vlček gives fit parameters for transitions to levels 4d, 5d, 6d (only unprimed), 6s, 7s and 8s (only un-primed). Details about the Drawin cross sections are given by Graef.

Zatsarinny *et al.* [83] report cross sections for excitation from 4s levels, which were used as lookup tables. The transitions from the four individual 4s levels to the 4s, 3d and 5s levels are given.

The cross sections for the remaining transitions were taken from Kimura *et al.* [84], who list fit parameters for Drawin cross sections. Some cross sections needed to be split into separate cross sections according to equation (5.4).

Table 5.3: Species that are present in the argon model with 78 levels. The model also contains Ar, Ar^+ and e.

Individual levels (number of levels)	Grouped levels Primed and un-primed
4s (2), 4s' (2), 5s (2), 5s' (2)	6s, 7s, 8s, 9s, 10s
4p (6), 4p' (4), 5p (6), 5p' (4)	6p, 7p, 8p, 9p
3d (8), 3d' (4)	4d, 5d, 6d, 7d, 8d
	4f, 5f, 6f, 7f

Finally, some cross sections for ionization were included. Ionization from the ground state is described by a cross section reported by Yanguas-Gil *et al.*, which was taken from experiments, performed by Rapp *et al.* [85]. Cross sections for ionization from the other levels are given by Vlček as fit parameters for the Drawin cross sections.

For the model, charge neutrality was assumed. The electron temperature is chosen to have a constant value of 0.9 eV. The heavy particle temperature has a fixed value of 600 K. The model is solved using the LSODA solver [72], with the relative and absolute tolerances set to 10^{-10}. This low value was necessary to get accurate tabulation of the source terms because slow processes are present, hence the net sources are much lower than the sources and sinks.

A training set was generated for the PCA model by running the full (global) model for two different pressures, namely 0.44 Pa and 1.0 Pa. The reduced model was run for a pressure of 0.67 Pa. The initial conditions for the densities of the excited states of argon were set to zero. The initial values of the densities of the ions and neutrals were chosen in such a way that the initial ionization degree was 1 %. The PC continuity equations are solved using the MATLAB ode45 solver. The species densities and PC sources are recovered using lookup tables with the custom triangulation. A log transformation and the 'vast' scaling method were used, because it gives a unique and relatively smooth manifold.

5.2.2 Comparison

Figure 5.5 shows the density of the 4s[3/2]1 state as a function of time for the full model in blue and the reduced models of PCA in red and ILDM in green. Three regimes are identified based on the full model result. In the first regime the density increases from $0\,\mathrm{m}^{-3}$ to about $4 \cdot 10^9\,\mathrm{m}^{-3}$ in the first 10^{-4} s. In the second the density remains constant until $t = 1$ s. At that time a slow process occurs, the third regime, which increases the density again to a value of $2 \cdot 10^{10}\,\mathrm{m}^{-3}$. This behaviour indicates that the chemistry is stiff, hence ILDM is very promising. The stiffness also becomes evident from the timescales that are calculated from ILDM. The largest timescale is $\tau = 1.3$ s, whereas all other timescales are smaller than $1.1 \cdot 10^{-6}$ s. The result for ILDM does not capture the first regime, but the second and third regimes are captured very well. The PC model managed to reproduce all regimes, but the calculated densities for the second transition were less accurate than

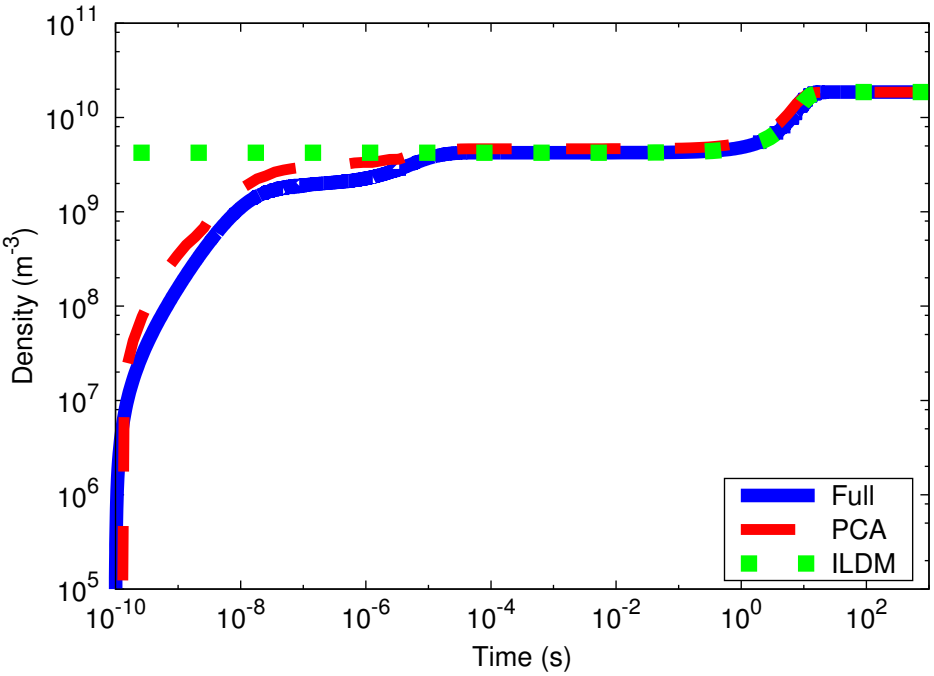


Figure 5.5: Time evolution of the density of the 4s[3/2]1 state of argon for the full, PC and ILDM model. The result for the full model is given in blue, the PCA model in red and the ILDM model in green.

the densities calculated by ILDM.

Table 5.4 shows the values of R^2 for some densities of the reduced simulations compared to the full simulation. The results for the PCA calculations show good agreement with the densities from the full simulation. The values of R^2 for ILDM are lower than PCA for the excited states of argon. The values for ground state Ar, Ar^+ and e are very high, because the densities of these species are constant during the fast time scales, thus no error is made for these species when neglecting the fast time scales. One could, however, argue that the error of ILDM should only be quantified for slow processes, when the user is only interested in these slow processes and neglects fast processes. Therefore, table 5.4 also includes values of R^2 for $t > 10^{-4}$ s, which are all approximately equal to one. For this reason, ILDM suits this model very well if the user is not interested in the initial disturbances of the system, but only in the dynamics of the system for longer times. Errors are very low for this case. PCA is more useful if the user is interested in the entire time range and gives acceptable results.

Table 5.4: Values of R^2 for the argon model with 78 levels.

Method	Ar	Ar^+	4s[3/2]1	4s[3/2]2
PCA	0.9417	0.9768	0.9751	0.9581
ILDM (full range)	1	1	0.9208	-1.5854
ILDM ($t > 10^{-4}$ s)	1	1	1	1

Table 5.5: Species that are present in the CO_2 model. The number of excited states that are included in the model is indicated by the number between brackets.

Neutral ground states	CO_2, CO, O_2, C_2O, O, O_3
Vibrational levels	$CO_2(25)$, CO(10), $O_2(3)$
Electronic states	$CO_2(2)$, CO(4), $O_2(2)$
Charged species	CO_2^+, e

5.3 CO_2 microwave model

5.3.1 Chemistry

The third model that is used for the comparison of PCA and ILDM is a CO_2 microwave model. The chemistry of this model is based on the chemistry that was developed by Koelman *et al.* [6]. They present an extensive CO_2 chemistry, including several vibrational modes of CO_2, CO and O_2 as well as several electronically excited states of the same species. The chemistry was modified by removing all ionic species, except for the CO^+ species as well as the C and C_2 species. This was necessary, because the full model model would not reach the equilibrium point, which is needed for the ILDM calculations. Instead, the densities show spikes after a long time and the solver returns an error. It was not clear what caused this solver error, but other solvers gave similar problems. A list of species is given in table 5.5 and the complete set of reactions after removing the species is given in appendices C, D, E and F. The reactions include electron impact ionization and excitation reactions, electron attachment and electron-ion recombination reactions, neutral-neutral and ion-neutral interactions, VV energy transfer reactions (exchange of vibrational energy between species) and VT energy transfer reactions (conversion of vibrational energy into heat). More details about these reactions are found in [6].

Similar to the other models, the electron density was calculated from charge neutrality. The electron temperature was set to a constant value of 3 eV and the gas temperature was 300 K. The stiff ODE solver LSODA was used to solve the complete set of differential equations.

The training set for PCA was made by running the full global model for two different initial pressures of 71.9 kPa and 162 kPa. The reduced calculations were done for a pressure of 107 kPa. As usual, the PC continuity equations are solved using the MATLAB ode45 solver. The species densities and PC sources are recovered using lookup tables with

Table 5.6: Values of R^2 for the CO_2 microwave model with an electron temperature of 3 eV.

Method	CO_2	CO	CO_2^+	CO_2[v01]	CO[v01]
PCA	0.9818	0.8556	0.9130	0.9107	0.8897
ILDM	-0.8738	-1.6697	-0.7725	-0.3430	-1.5694

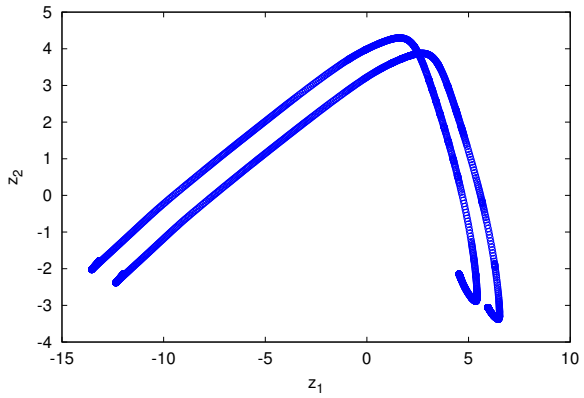
the custom triangulation. A log transformation and the 'auto' scaling method were used. Unfortunately, none of the scaling methods give a unique manifold and therefore the manifold is cut into four separate pieces. Figure 5.6 shows the solution of the PC continuity equations for the four separate parts of the manifold.

5.3.2 Comparison for a high electron temperature

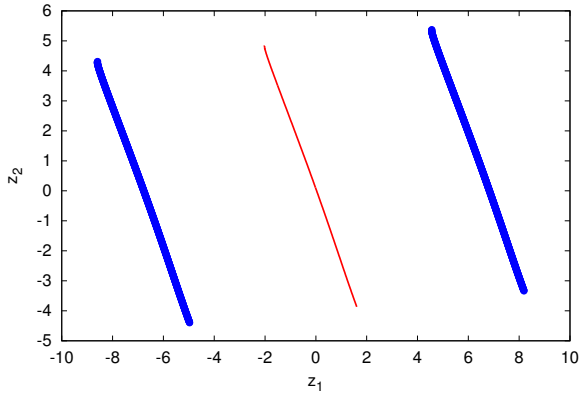
Figure 5.7 shows the density of CO_2^+ as a function of time for the full model in blue, PCA in red and ILDM in green. The density shows a smooth transition from $10^{10}\,\mathrm{m}^{-3}$ to $10^{21}\,\mathrm{m}^{-3}$ in about $10^{-7}\,\mathrm{s}$, before reaching the equilibrium value of $10^{22}\,\mathrm{m}^{-3}$. The PCA solution increases slightly faster towards the equilibrium point than the solution of the full model. This behaviour was also observed for the molecular argon model and the argon model with 78 levels. The solution for the ILDM model shows a nearly constant density of CO_2^+ and fails to capture the major changes of the densities. When we look at the ILDM timescales at the equilibrium point, we see that there are eight high values with very close values. It is, however, very impractical to take this many parameters, since this requires an eight dimensional lookup table, which would be incredibly large. In our case, only one parameter was used. For this reason ILDM is not a useful reduction method for this model when only one parameter is used. This is confirmed by the calculated R^2 values in table 5.6, which shows good agreement for the PCA result but not for ILDM. ILDM can, however, still give useful information about the system.

With ILDM it is possible to get insight in the behaviour of the chemical system using the eigenvalues of the Jacobian of the sources. These eigenvalues can be converted into time scales using equation (3.35). Linear systems with sources in the form of $S_i = \sum_j k_j n_j$ have a constant Jacobian, hence the eigenvalues and time scales are constant. The CO_2 chemistry is not linear and the Jacobian depends on the species densities, so the eigenvalues and time scales will change over time. In this case the nomenclature "time scales" is not correct, but a more appropriate phrase would be "local time scales", which specifies that only in the vicinity of a certain point in time the system has a time scale that is equal to the local time scale. For simplicity we will refer to the local time scales as time scales from now on.

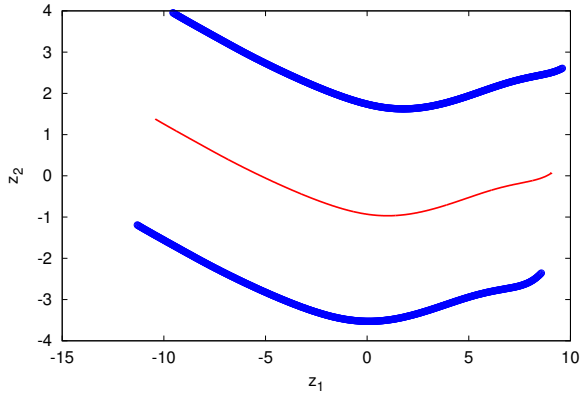
Figure 5.8 presents the local time scales for this chemical system, with the colors distinguishing between the 54 different eigenvalues that are used to calculate the time scales. This figure shows that the time scales are all showing a similar behavior, indicating that the species densities are strongly coupled via scaling relations indeed. When the system



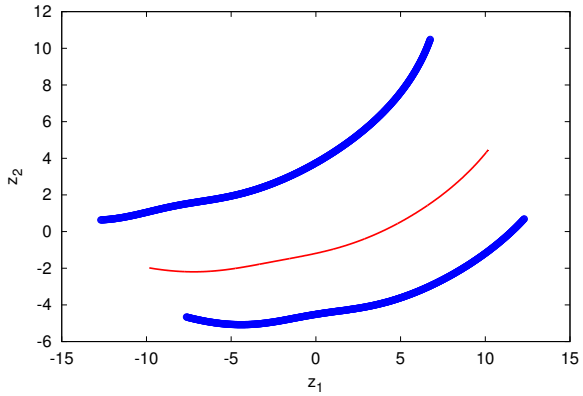
(a) Full manifold



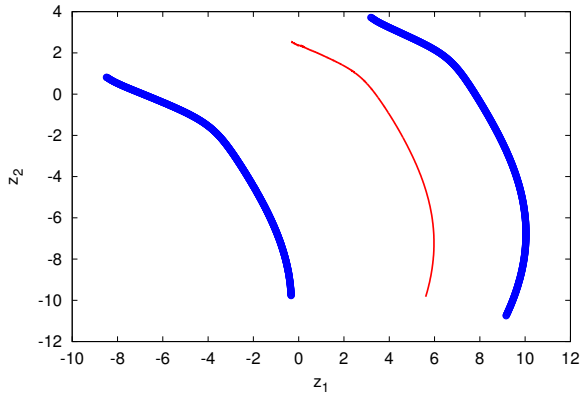
(b) Part 1



(c) Part 2



(d) Part 3



(e) Part 4

Figure 5.6: The solutions of the PC continuity equations for the CO_2 model for a high electron temperature. The manifold was split into four separate parts.

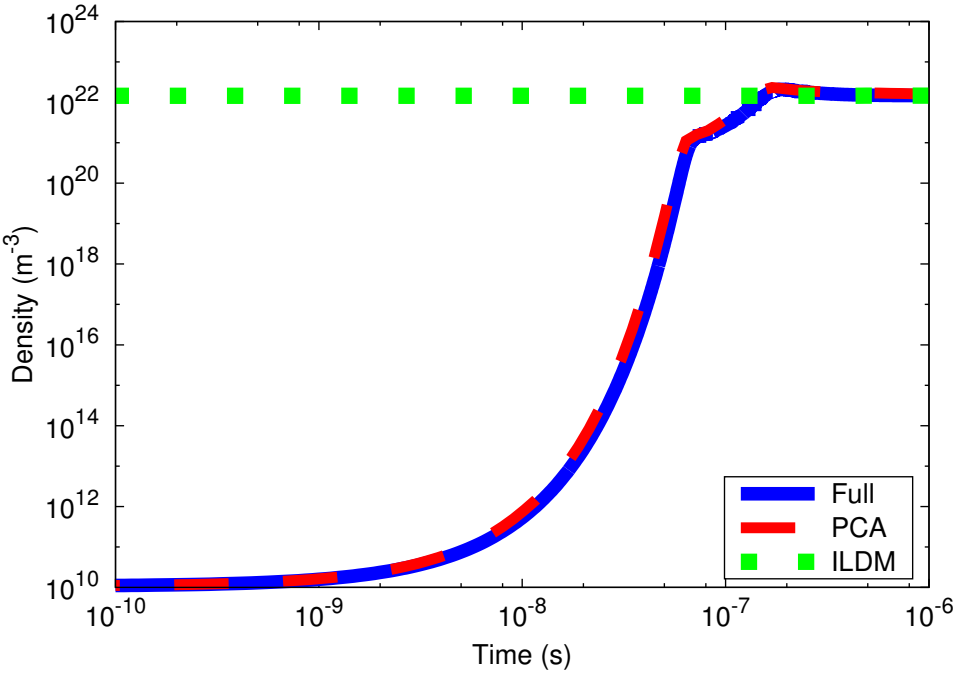


Figure 5.7: Comparison of the density of CO_2^+ as a function of time for the full model in blue, the PC model and the ILDM model. For the PC model, 'auto' scaling, a log transformation and lookup tables with the custom triangulation were used.

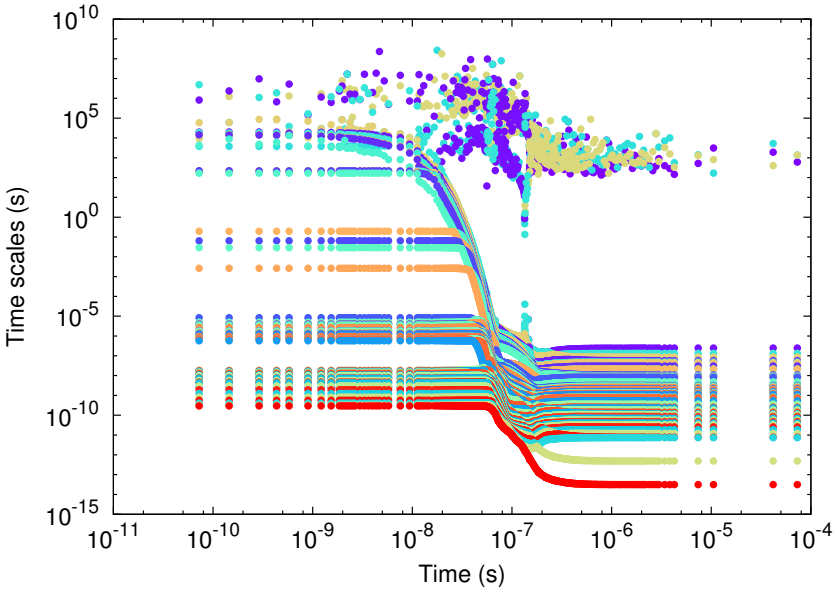
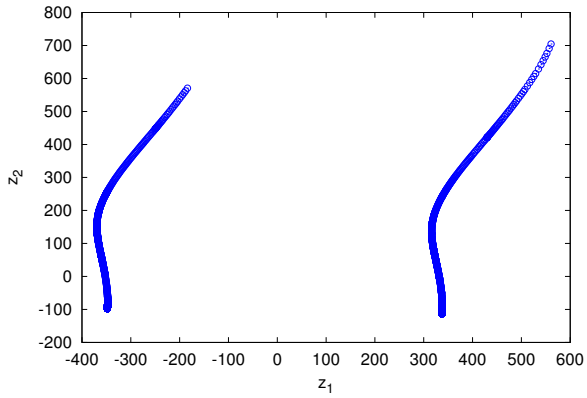


Figure 5.8: Time scales for the CO_2 model as a function of time for an electron temperature of 3 eV. The different colors represent the different eigenvalues that are used to calculate the time scales of the system. The spike like feature in the time scales around 0.2 µs is a result of the sign swap in one of the eigenvalues, from positive to negative.

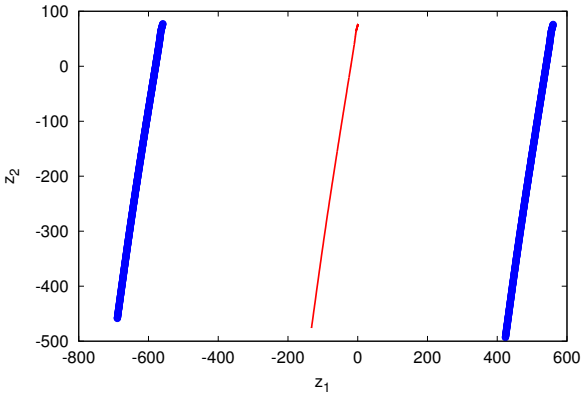
is in equilibrium, which it is at 10 µs, one can see that the time scales seem to form a continuum of time scales, rather than one well separated time scale for the parameter as the previous chemical systems showed. When solving the system for only this parameter that shows the highest time scale, and disregarding all other parameters, the cutoff time is equal to the time scale of that very same parameter. This is exactly what happened in the result of figure 5.7 that is produced using ILDM. Expanding the number of parameters in the ILDM calculations would indeed lower the cutoff time, but this would result in an increase in the dimensionality of the lookup tables. Practically, expanding the number of parameters to more than two or three would result in an unfeasable method due to the high dimensionality of the lookup tables (and the resulting memory needed for that). For that reason ILDM will only be able to determine the equilibrium densities of the system, and no temporal information. However, it does give valuable information about the equilibrium of the system. From the smallest time scale in combination with the requirement that all eigenvalues need to be negative in equilibrium, we can determine with certainty that this system is at equilibrium around 1 µs.

5.3.3 Comparison for a low electron temperature

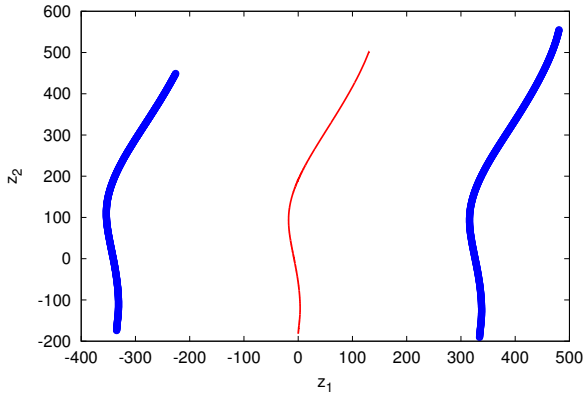
The same analysis was done for the CO_2 chemistry with an electron energy of 1 eV. For ILDM such a low electron temperature is problematic for this chemistry, since the manifold only consists of a few points. This makes it impractical to do a reduced simulation using



(a) Full manifold



(b) Part 1

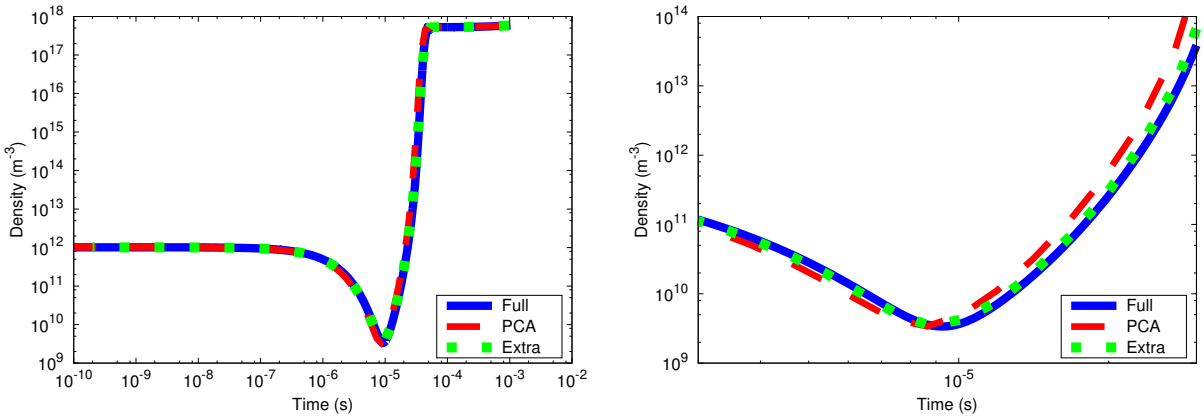


(c) Part 2

Figure 5.9: The solutions of the PC continuity equations for the CO_2 model for a high electron temperature. The manifold was split into two separate parts.

ILDM. Therefore PCA will be compared with only the full model. For PCA, 'vast' scaling was employed, together with the usual log transformation, lookup tables and custom triangulation. For this model it was necessary to split the manifold into two parts. The solution of the PC continuity equations for these two parts of the manifold is shown in figure 5.9.

The time evolution of the density of C for the full model in blue and the PCA model in red is shown in figure 5.10. The blue line represents the result for the full model and the red line gives the result for the PCA model. Similar to the previous results in this chapter, the density calculated from the PCA model shows gradients that occur slightly earlier than the gradients for the full model. This behaviour was not present in cases where an initial condition was chosen that was part of the training set, such as in section 4.4. This is not the case for the results in this chapter. A possible explanation for this observation is that the reduced model moves too quickly through the manifold. The speed of the trajectory through the manifold depends on the PC sources, so this indicates that the PC



(a) Full view (b) Zoomed in

Figure 5.10: Comparison of the density of C as a function of time for the full model, the PC model and the ILDM model. For the PC model, 'vast' scaling, a log transformation and lookup tables with the custom triangulation were used.

Table 5.7: Values of R^2 for the CO_2 microwave model with an electron temperature of 1 eV. The solution becomes more accurate with the extra trajectory in the manifold.

Method	CO_2	CO	CO_2^+	CO_2[v01]	CO[v01]
PCA	0.9810	0.9855	0.9636	0.9722	0.9601
Extra	0.9991	0.9993	0.9979	0.9987	0.9981

sources are overestimated due to errors of the linear interpolation in the lookup tables. This hypothesis is tested by including a run of the full global model for an initial pressure of 119 kPa in the training set. This initial pressure is much closer to the pressure for which the reduced global model is run, hence errors due to linear interpolation are expected to be smaller and the solution is expected to be closer to the full model. Figure 5.10 also shows the result of the PCA model using this extra pressure in the training set, given by the green line. The solution for the extra pressure is much closer to the full model than the solution without the extra pressure in red. The accuracy of the reduced calculations is quantified by the values of R^2, which are given in table 5.7. The values for the reduced model for the extra pressure are higher, which confirms the hypothesis. The error of the linear interpolation can be limited by ensuring that the variation of the parameter, in this case the pressure, is not too large. Another possibility to reduce the error of interpolation is to use nonlinear interpolation. Amidror [56] gives a description of a cubic triangular interpolation method, which is based on the Clough-Tocher method. The advantage of cubic interpolation schemes is that they also have continuous derivatives.

The time scales of the CO_2 model are also analysed for the low electron temperature case. Figure 5.11 shows the time scales as a function of time for 1 eV. Three very large time scales are visible, which correspond to conservation laws of the elements C, O and

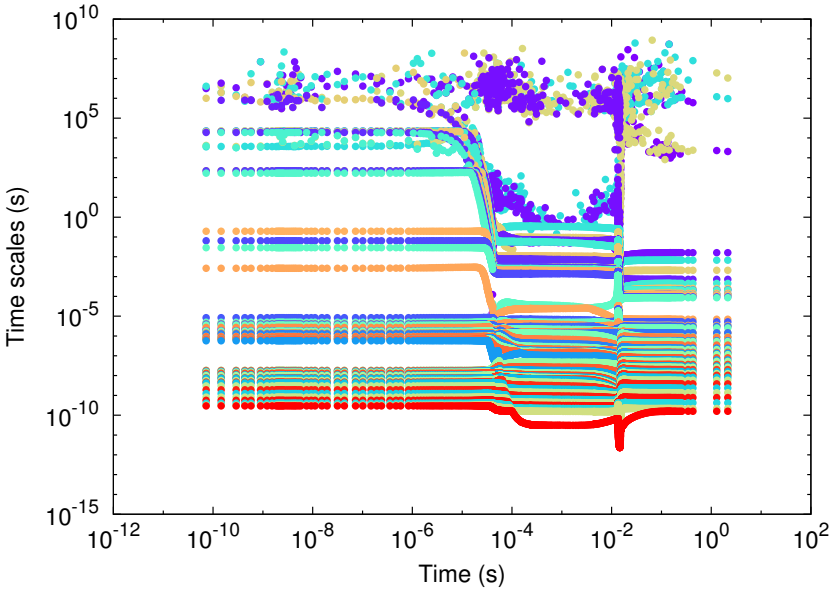


Figure 5.11: Time scales for the CO_2 model as a function of time for an electron temperature of 3 eV. The different colors represent the different eigenvalues that are used to calculate the time scales of the system. The spike like feature in the time scales around 0.2 µs is a result of the sign swap in one of the eigenvalues, from positive to negative.

electrons. After 10^{-1} s the slowest time scale is constant and approximately 1 s, hence the system reaches equilibrium at about 1 s. Apparently, the results shown in figure 5.10 did not reach equilibrium!

When the models are run upto equilibrium we see some strange behaviour, which is shown in figure 5.12. The density of CO is plotted as a function of time for the full model in blue and the PCA model in red. After 10^{-2} s the PCA result shows some steep gradients and oscillations. These features are not caused by PCA, but by the training set. The result of the full model, given by COLOR shows the same behaviour. Thus it should always be verified that the PCA training set contains correct results, since results from the PCA global model strongly depend on this training set. If the training set contains correct results, PCA will also give good results, but if the training set contains bad results, PCA will produce the same bad results.

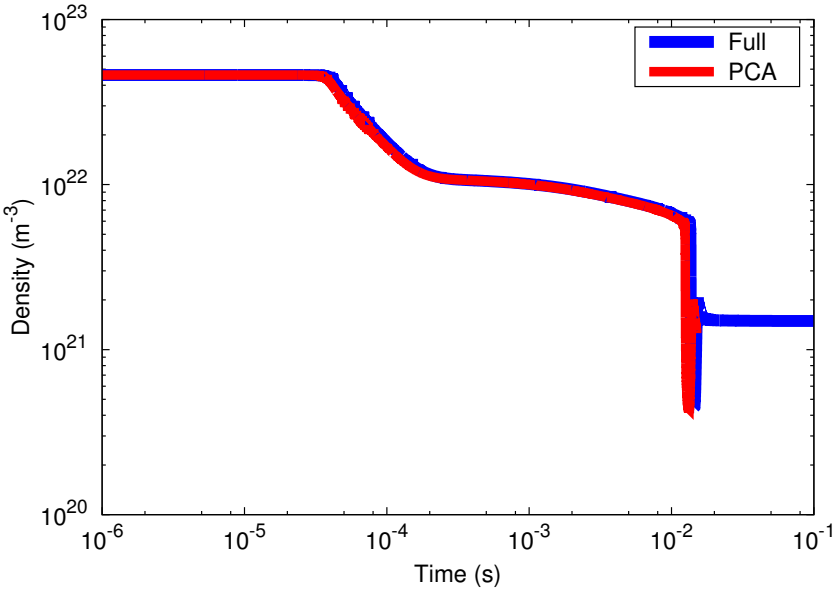


Figure 5.12: Results when the CO_2 model is run upto equilibrium for the full model in red and the PCA model in blue. Oscillations occur after 10^{-2} s.

Chapter 6

Nonlinear regression

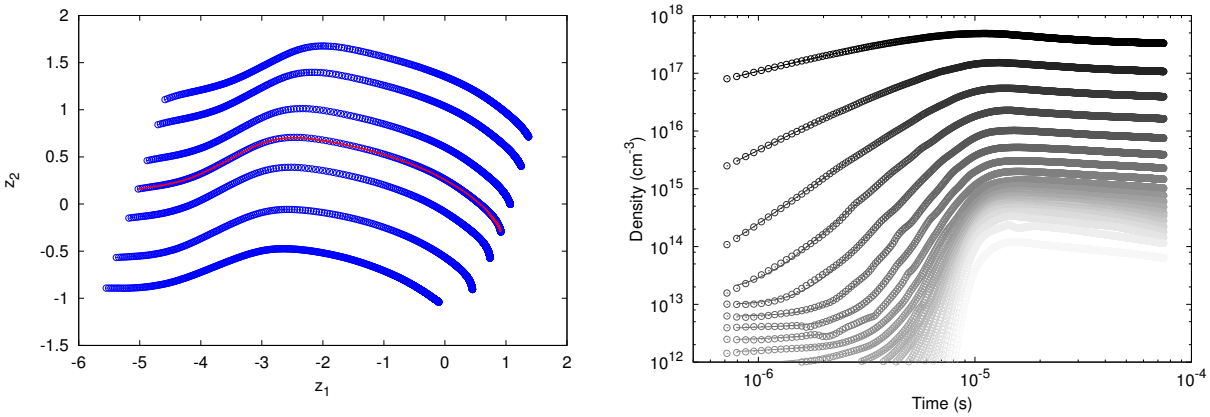
In the previous chapters, we used lookup tables for the reconstruction of the species densities and the PC sources. The main disadvantage of the lookup tables is that the simulations become much more expensive when the amount of PCs is increased. Therefore, we will use an alternative method for the reconstruction: MARS, which is called ARES since it is a registered trademark. PCA and ARES are applied on the model by Peerenboom *et al* and on the molecular argon model.

6.1 CO_2 model

First, we test the reconstruction using ARES on the training data for the CO_2 model by Peerenboom *et al.*, which was introduced in section 4.1. For PCA, the 'range' scaling and log transformation were used. The parameters for ARES are listed in table 6.1. The parameters are chosen such that the ARES model for the PC sources is more accurate than the ARES model for the species densities, since it is very important to calculate the PC sources accurately. Therefore, a high value for *maxFuncs*, which is the maximum amount of basis functions, was chosen. In all cases, piecewise-cubic basis functions were chosen because of their smoothness. The backward phase was enabled by setting *prune* to true, which is necessary to prevent overfitting, since a large number of basis functions is allowed. The maximum amount of interactions between the variables was set to 2, since

Table 6.1: ARES parameters for the CO_2 model of Peerenboom *et al.*.

Parameter	PC sources	Densities
cubic	true	true
maxFuncs	200	100
maxInteractions	2	2
prune	true	true
threshold	10^{-6}	10^{-5}



(a) Manifold (b) Densities

Figure 6.1: Reduced calculation using ARES for the model by Peerenboom *et al.*

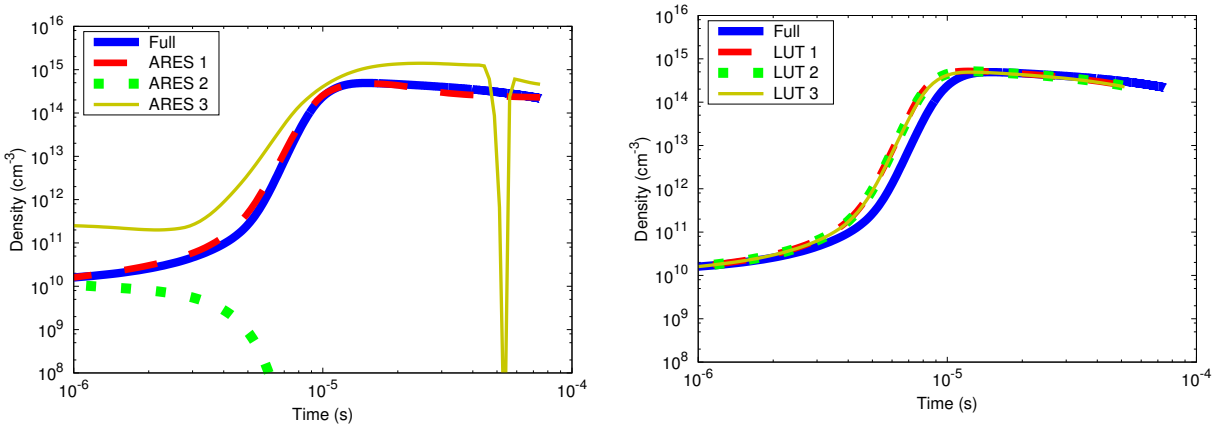
Table 6.2: Values of R^2 for the PCA model using lookup tables and ARES.

Method	CO_2	CO	O_2	O_3
Lookup table	1.0000	1.0000	1.0000	1.0000
ARES	0.9998	0.9743	0.9998	0.9594

higher values increase the time needed to find the ARES model significantly for the same *threshold* values.

Two reduced simulations were run for the CO_2 model using ARES. First, the initial condition with an ionization degree of 10^{-6} chosen, which was already in the training set. The calculated PC scores are shown in figure 6.1a, along with the reaction trajectories. The solution follows the manifold shape very well. Figure 6.1b shows the densities of the vibrationally excited states as a function of time. The calculated densities, denoted by the circles, show some slight deviations from the densities obtained from the full simulation given by the solid lines. These errors were not visible for the calculation using lookup tables. Table 6.2 lists the R^2 values for both the lookup tables and ARES. The values show that calculations using ARES result in some errors, whereas the calculations using the lookup tables are very accurate.

One of the major differences between the lookup tables and ARES is the amount of points that is used for the recovery of the species densities and the PC sources. As explained in section 3.4.2 the interpolation in the lookup tables is done using a triangulation. When we want to lookup a density or PC source for given PC scores z_1 and z_2, the triangle is found in which this point lies. After that the linear interpolation is done in this single triangle, hence only three points are used for the interpolation in the lookup table. ARES, however, constructs a regression function based on all the PC scores that are in the training set. Hence the amount data sets included in the PCA training set potentially influences the regression function. In order to study this influence several reduced calculations for an



(a) ARES

(b) Lookup tables

Figure 6.2: Reduced calculation using ARES and lookup tables for the model by Peerenboom *et al.* The result for the full model is given in blue for the first asymmetric vibrational mode of CO_2. For the reduced calculations results for a training set with ionization degrees of $1 \cdot 10^{-7}$, $2 \cdot 10^{-7}$, $5 \cdot 10^{-7}$, $2 \cdot 10^{-6}$, $5 \cdot 10^{-6}$ and $1 \cdot 10^{-5}$ are given in red, ionization degrees of $2 \cdot 10^{-7}$, $5 \cdot 10^{-7}$, $2 \cdot 10^{-6}$ and $5 \cdot 10^{-6}$ in blue and ionization degrees of $5 \cdot 10^{-7}$ and $2 \cdot 10^{-6}$ in yellow.

ionization degree of $1 \cdot 10^{-6}$ were done for different training sets. Results for the density of the first asymmetric vibrational mode of CO_2 are shown in figure 6.2 for both ARES and the lookup tables. The result for the full model is given in blue. For the reduced calculations results for a training set with ionization degrees of $1 \cdot 10^{-7}$, $2 \cdot 10^{-7}$, $5 \cdot 10^{-7}$, $2 \cdot 10^{-6}$, $5 \cdot 10^{-6}$ and $1 \cdot 10^{-5}$ are given in red, ionization degrees of $2 \cdot 10^{-7}$, $5 \cdot 10^{-7}$, $2 \cdot 10^{-6}$ and $5 \cdot 10^{-6}$ in blue and ionization degrees of $5 \cdot 10^{-7}$ and $2 \cdot 10^{-6}$ in yellow. The reduced calculations using ARES with a training set of six ionization degrees agrees with the full simulation, but calculations where less ionization degrees were included show unacceptable errors. The results using ARES are clearly different for different training sets. This is not the case for the lookup tables, for which results are shown in figure 6.2b. The reduced calculation for the different training sets almost perfectly overlap. The changes of the densities occur earlier for the reduced calculation than for the full model, which was also observed in chapter 5. The observations are confirmed by the values of R^2 listed in table 6.3. The result for ARES using six ionization degrees in the training sets shows very good agreement with the full model. Errors are large for the other two cases, where only four and two ionization degrees were included. For the lookup tables the result is slightly less accurate when more ionization degrees were included in the training set.

Table 6.3: Values of R^2 for the log of the densities for the PCA model using lookup tables and ARES. ARES 1 and LUT 1 corresponds to the training set that included ionization degrees of $1 \cdot 10^{-7}$, $2 \cdot 10^{-7}$, $5 \cdot 10^{-7}$, $2 \cdot 10^{-6}$, $5 \cdot 10^{-6}$ and $1 \cdot 10^{-5}$, ARES 2 and LUT 2 to $2 \cdot 10^{-7}$, $5 \cdot 10^{-7}$, $2 \cdot 10^{-6}$ and $5 \cdot 10^{-6}$, and ARES 3 and LUT 3 to $5 \cdot 10^{-7}$ and $2 \cdot 10^{-6}$.

Name	CO_2	CO	O_2	O_3	CO_2[v01]
ARES 1	0.9981	0.9976	0.9975	0.9976	0.9909
ARES 2	-146.06	-80.966	-91.520	-0.0201	-417.72
ARES 3	-4.4901	-0.7891	0.1760	-2.6663	0.2911
LUT 1	0.9261	0.9512	0.9584	0.9353	0.9322
LUT 2	0.9327	0.9554	0.9618	0.9405	0.9343
LUT 3	0.9441	0.9620	0.9672	0.9487	0.9360

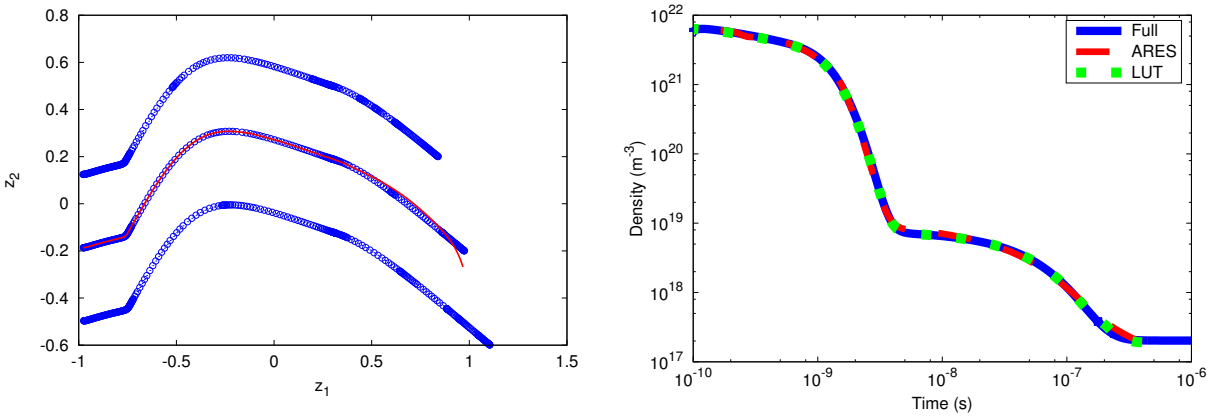
Table 6.4: ARES parameters for the molecular argon model.

Parameter	PC sources	Densities
cubic	true	true
maxFuncs	200	100
maxInteractions	2	2
prune	true	true
threshold	10^{-8}	10^{-5}

6.2 Molecular argon model

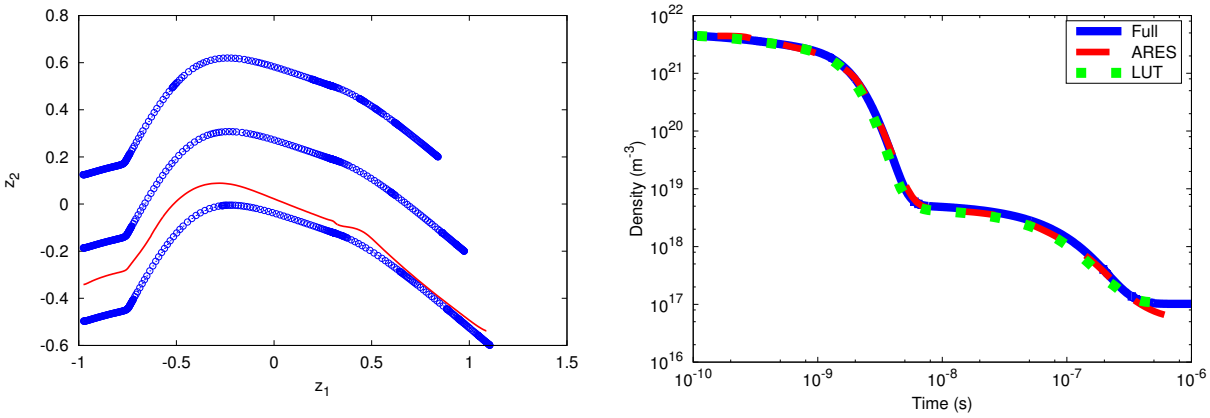
The second model for which ARES is used for the reconstruction is the molecular argon model. Details about this model are given in section 5.1.1. For the PCA model the 'range' scaling method and a log transformation were used. The ARES parameters for the molecular argon model are listed in table 6.4. The value of the *threshold* parameter for the PC sources was set to a lower value than the value used for the CO_2 model. First, an initial condition was chosen that was already in the training set. The density of Ar^r as a function of time and the solution in the manifold are given in figure 6.3. The calculated PC scores in the manifold plot closely follow the blue trajectories in the first part, but deviations are visible close to the end. These deviations are also visible for the densities given as a function of time in figure 6.3b,

The errors are larger when an initial condition is chosen that is not part of the training set. Results for such a case are shown in figure 6.4, where also the density of Ar^r as a function time and the manifold are plotted. In the initial phase the reduced solution follows the manifold well, but after some time the solution becomes very inaccurate and moves towards the bottom manifold trajectory. This is also visible for the density of Ar^r, which is shown in figure 6.3b, where the final value of the density is significantly lower than the equilibrium value.



(a) Manifold (b) Ar^r density

Figure 6.3: Reduced calculation of the molecular argon model using ARES.



(a) Manifold (b) Densities

Figure 6.4: Reduced calculation of the molecular argon model using ARES. Initial condition between the manifold.

Peerenboom *et al.* [11] showed that linear extrapolation of the manifold quickly becomes very inaccurate when the PC solution leaves the manifold. This is not necessarily true for nonlinear regression, since nonlinear regression is able to take into account nonlinearities. Furthermore, a nonlinear regression function is based on all points instead of just a few for lookup tables. However, since ARES already gives significant errors when the PC solution is still within the range of the training set, ARES might not be suited for calculations outside the range of the training set. Other forms of nonlinear regression, such as Artificial Neural Networks (ANN), might have a better performance.

Chapter 7

Conclusion

The aim of this work was to evaluate the accuracy of the PCA global model compared to a full global model and an ILDM-based global model Results from a PCA global were compared with the full model and an ILDM global model. Three global models were used for the comparison: a molecular argon model, an argon model with 78 levels and a microwave CO_2 model. For all of the models it was found that the results obtained PCA global model are in close agreement with results found from the full model. The ILDM-based global model was very accurate in some specific cases, when only a limit time range was considered.

The results for the PCA global model show a similar trend for all three models. It was observed that the changes in the species densities occur slightly earlier for the PCA results than for the full model results. A possible explanation was that the reduced model moves too quickly through the manifold due to overestimation of the PC sources. The PC sources were determined from linear interpolation in the lookup tables. It was shown that the errors are reduced when an extra model run is added that reduces the errors due to linear interpolation, which confirms the hypothesis. Thus it must be ensured that the separation distance of the PC scores of the training set is not too large. Nonlinear interpolation methods could potentially reduce errors in the calculation of the PC sources even more.

The accuracy of the reduction method ILDM strongly depends on the time scales that are included in the model. ILDM calculates time scales automatically from the description of the chemical reaction system. The user can decide which time scales are included by defining a cut-off time. We only included the longest time scale. For times longer than this time scale ILDM results are in exact agreement with the full model, whereas for PCA some errors are still visible. ILDM is less accurate than PCA for times below the cut off time, but this is expected since ILDM only incorporates time scales larger than the cut off time. For the CO_2 model ILDM was only able to resolve the equilibrium values of the species densities. The result can be improved by including more time scales in the model, but this was not a viable option for this model since it would require extremely large lookup tables.

The PCA global model was improved by two additions. The first improvement was made in the linear interpolation, which is based on triangular methods as is described by

Amidror [56]. It was found that the triangulation of the PC scores has a large impact on the accuracy of the PCA global model. The Delaunay triangulation was not suited for this study, so a custom triangulation was implemented that is much more suited for the PCA global model. Results using the custom triangulation show much smaller errors than result using the Delaunay triangulation. Secondly, a method was developed that makes it possible to solve the PCA global model also for manifolds for which the PC scores of the training set cannot be described uniquely by two PCs. This problem could be solved by increasing the amount of PCs to three, but increasing the amount of PCs also increases the computational load of the model due to the lookup tables. As an alternative, the training set is divided into separate unique parts. The PC continuity equations are solved for these unique parts separately. The manifold is split at positions where the curvature of the training set trajectories attains a maximum, since it was found that uniqueness was often violated when the curvature was high. It was shown that the splitting of the manifold does not introduce any significant errors but even gave small improvements.

Peerenboom *et al.* [11] proposed the use of nonlinear regression for the recovery of the PC sources and species densities. The advantage of nonlinear regression is that it is able to take into account nonlinearities unlike linear interpolation. Furthermore, nonlinear regression takes into account all points in the training set, hence it has more potential for extrapolation. In cooperation with Luca Vialetto and Dr Emile Carbone the nonlinear regression method MARS/ARES was studied as an alternative to linear interpolation of a lookup table for the reconstruction of the PC sources and species densities. It was shown that results using MARS/ARES are much less accurate than the results obtained using the lookup tables. Other available forms of nonlinear regression, such as Artificial Neural Networks (ANN), might be a better alternative, but this is beyond the scope of this work.

Chapter 8

Acknowledgements

After spending a full year at EPG for my graduation project, I would like to thank everyone who contributed to this work. In particular, I would like to thank my daily supervisor Peter Koelman for all the valuable time and effort that you spent on my project. This includes giving useful feedback on my results and spending many hours to give comments on my thesis. Your guidance has certainly helped me to bring this project to a good end. I also really appreciate the help that Jan van Dijk gave me during the year. Although you had very little time, you were always willing to give useful comments about my work and give useful ideas.

During the project I also worked together with Tafizur Rehman, who performed a different form of reduction using ILDM. I want to thank you for helping me with running the ILDM codes that you wrote and comparing the results from our reduction methods. I hope that you will succeed to defend your PhD thesis and have a succesful career in India.

I am grateful that Luca Vialetto and Emile Carbone have been largely involved in my work on PCA. I really enjoyed all the constructive and fun Skype conversations that we had during the year. I apologize for all the times Peter and I were late for the meetings :)

I want to thank Wouter for helping me whenever I experienced problems related to PLASIMO. You spent much time dealing with issues related to the Global Model. Without your fixes I would not have been able to produce the results in this thesis. You have also been very helpful whenever there were problems with my computer, for example when the old computer was overheating.

I would like to thank the PLASIMO team for the year I was able to spend with you. Jesper Janssen helped me a lot when Peter, Jan and Wouter were not available. I shared many fun experiences with Sebastiaan Selvi, who was doing his graduation project in the PLASIMO section of EPG during the same time as me. And also the other members have always given useful ideas for my work and also fun conversations.

I also got to know the other member of EPG, especially during the lunch breaks, which often contained humorous and interesting conversation. All of you are also thanked. I also really enjoyed the ping-pong session that I had with you, although there were only a few.

Last but not least I would like to thank my parents, my brothers and friends for supporting and motivating me when this was needed. Your support has certainly helped

to keep me motivated and bring this project to a good end.

Bibliography

[1] J Chang, Dennis YC Leung, CZ Wu, and ZH Yuan. A review on the energy production, consumption, and prospect of renewable energy in china. *Renewable and Sustainable Energy Reviews*, 7(5):453–468, 2003.

[2] X Tu and JC Whitehead. Plasma-catalytic dry reforming of methane in an atmospheric dielectric barrier discharge: Understanding the synergistic effect at low temperature. *Applied Catalysis B: Environmental*, 125:439–448, 2012.

[3] Danhua Mei, Xinbo Zhu, Ya-Ling He, Joseph D Yan, and Xin Tu. Plasma-assisted conversion of co2 in a dielectric barrier discharge reactor: understanding the effect of packing materials. *Plasma Sources Sci. Technol*, 24(1):015011, 2015.

[4] VD Rusanov, AA Fridman, and GV Sholin. The physics of a chemically active plasma with nonequilibrium vibrational excitation of molecules. *Physics-Uspekhi*, 24(6):447–474, 1981.

[5] Alexander Fridman. *Plasma chemistry*. Cambridge university press, 2008.

[6] Peter Koelman, Stijn Heijkers, Samaneh Tadayon Mousavi, Wouter Graef, Diana Mihailova, Tomas Kozak, Annemie Bogaerts, and Jan van Dijk. A comprehensive chemical model for the splitting of co2 in non-equilibrium plasmas. *Plasma Processes and Polymers*, 14(4-5), 2017.

[7] Francesco Taccogna and Giorgio Dilecce. Non-equilibrium in low-temperature plasmas. *The European Physical Journal D*, 70(11):251, 2016.

[8] Alice M Harling, David J Glover, J Christopher Whitehead, and Kui Zhang. Novel method for enhancing the destruction of environmental pollutants by the combination of multiple plasma discharges. *Environmental science & technology*, 42(12):4546–4550, 2008.

[9] Tomáš Kozák and Annemie Bogaerts. Splitting of co2 by vibrational excitation in non-equilibrium plasmas: a reaction kinetics model. *Plasma Sources Science and Technology*, 23(4):045004, 2014.

[10] Robby Aerts, Tom Martens, and Annemie Bogaerts. Influence of vibrational states on co2 splitting by dielectric barrier discharges. *The Journal of Physical Chemistry C*, 116(44):23257–23273, 2012.

[11] K.S.C. Peerenboom, A. Parente, T. Kozak, A. Bogaerts, and G. Degrez. Dimension reduction of non-equilibrium plasma kinetic models using principal component analysis. *Plasma Sources Science and Technology*, 24, 2015.

[12] Hai P Le, Ann R Karagozian, and Jean-Luc Cambier. Complexity reduction of collisional-radiative kinetics for atomic plasma. *Physics of Plasmas*, 20(12):123304, 2013.

[13] Thierry E Magin, Marco Panesi, Anne Bourdon, Richard L Jaffe, and David W Schwenke. Coarse-grain model for internal energy excitation and dissociation of molecular nitrogen. *Chemical Physics*, 398:90–95, 2012.

[14] Dr R Bates, AE Kingston, and RW Pt McWHIRTER. Recombination between electrons and atomic ions. i. optically thin plasmas. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, pages 297–312, 1962.

[15] DR Bates, AE Kingston, and RWP McWhirter. Recombination between electrons and atomic ions. ii. optically thick plasmas. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, pages 155–167, 1962.

[16] JAM Van der Mullen. Excitation equilibria in plasmas; a classification. *Physics Reports*, 191(2-3):109–220, 1990.

[17] Ulrich Maas and Stephen B Pope. Simplifying chemical kinetics: intrinsic low-dimensional manifolds in composition space. *Combustion and flame*, 88(3):239–264, 1992.

[18] Ulrich Maas. Efficient calculation of intrinsic low-dimensional manifolds for the simplification of chemical kinetics. *Computing and Visualization in Science*, 1(2):69–81, 1998.

[19] Ralph Lehmann. An algorithm for the determination of all significant pathways in chemical reaction systems. *Journal of atmospheric chemistry*, 47(1):45–78, 2004.

[20] Aram H Markosyan, Alejandro Luque, Francisco J Gordillo-Vázquez, and Ute Ebert. Pumpkin: A tool to find principal pathways in plasma chemical models. *Computer Physics Communications*, 185(10):2697–2702, 2014.

[21] JA Van Oijen and LPH De Goey. Modelling of premixed counterflow flames using the flamelet-generated manifold method. *Combustion Theory and Modelling*, 6(3):463–478, 2002.

[22] I.T. Jolliffe. *Principal Component Analysis, Second Edition*. Springer, 2002.

[23] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[24] Catriona J MacLeod and Henrik Moller. Intensification and diversification of new zealand agriculture since 1960: An evaluation of current indicators of land use change. *Agriculture, ecosystems & environment*, 115(1):201–218, 2006.

[25] James C Liao, Riccardo Boscolo, Young-Lyeol Yang, Linh My Tran, Chiara Sabatti, and Vwani P Roychowdhury. Network component analysis: Reconstruction of regulatory signals in biological systems. *Proceedings of the National Academy of Sciences of the United States of America*, pages 15522–15527, 2003.

[26] Leo H Chiang, Evan L Russell, and Richard D Braatz. Fault diagnosis in chemical processes using fisher discriminant analysis, discriminant partial least squares, and principal component analysis. *Chemometrics and intelligent laboratory systems*, 50(2):243–252, 2000.

[27] James E Overland and RW Preisendorfer. A significance test for principal components applied to a cyclone climatology. *Monthly Weather Review*, 110(1):1–4, 1982.

[28] Seema Vyas and Lilani Kumaranayake. Constructing socio-economic status indices: how to use principal components analysis. *Health policy and planning*, 21(6):459–468, 2006.

[29] Donald A Jackson and Yong Chen. Robust principal component analysis and outlier detection with ecological data. *Environmetrics*, 15(2):129–139, 2004.

[30] Yong He, Xiaoli Li, and Xunfei Deng. Discrimination of varieties of tea using near infrared spectroscopy by principal component analysis and bp model. *Journal of Food Engineering*, 79(4):1238–1242, 2007.

[31] John Novembre and Matthew Stephens. Interpreting principal component analyses of spatial population genetic variation. *Nature*, 200:8.

[32] PS Kwarteng and AY Chavez. Extracting spectral contrast in landsat thematic mapper image data using selective principal component analysis. *Photogramm. Eng. Remote Sens*, 55:339–348, 1989.

[33] John D Horel. A rotated principal component analysis of the interannual variability of the northern hemisphere 500 mb height field. *Monthly Weather Review*, 109(10):2080–2092, 1981.

[34] Torunn Berg, Oddvar Røyset, Eiliv Steinnes, and Marit Vadset. Atmospheric trace element deposition: principal component analysis of icp-ms data from moss samples. *Environmental Pollution*, 88(1):67–77, 1995.

[35] Robert Raskin and Howard Terry. A principal-components analysis of the narcissistic personality inventory and further evidence of its construct validity. *Journal of Personality and Social Psychology*, 54(5,890-902), 1988.

[36] J Edward Jackson and Govind S Mudholkar. Control procedures for residuals associated with principal component analysis. *TECHNOMETRICS*, 21(3), 1979.

[37] Stephanie A. Navarro Silvera, Susan T. Mayne, Harvey A. Risch, Marilie D. Gammon, Thomas Vaughan, Wong-Ho Chow, Joel A. Dubin, Robert Dubrow, Janet Schoenberg, Janet L. Stanford, A. Brian West, Heidrun Rotterdam, and William J. Blot. Principal component analysis of dietary and lifestyle patterns in relation to risk of subtypes of esophageal and gastric cancer. *Annals of Epidemiology*, 21:543–550, 2011.

[38] Christopher Chatfied and Alexander J Collins. *Introduction to multivariate analysis*. Springer, 2013.

[39] J. Edward Jackson. *A User's Guide To Principal Components*. John Wiley & Sons, Inc., 1991.

[40] Hans A Hegt, Ron J De La Haye, and Nadeem A Khan. A high performance license plate recognition system. In *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, volume 5, pages 4357–4362. IEEE, 1998.

[41] Michael Unser. On the approximation of the discrete karhunen-loeve transform for stationary processes. *Signal Processing*, 7(3):231–249, 1984.

[42] Anindya Chatterjee. An introduction to the proper orthogonal decomposition. *Current science*, pages 808–817, 2000.

[43] A Hannachi, IT Jolliffe, and DB Stephenson. Empirical orthogonal functions and related techniques in atmospheric science: A review. *International journal of climatology*, 27(9):1119–1152, 2007.

[44] Richard Everson, Peter Cornillon, Lawrence Sirovich, and Andrew Webber. An empirical eigenfunction analysis of sea surface temperatures in the western north atlantic. *Journal of Physical Oceanography*, 27(3):468–479, 1997.

[45] R. Bro and A.K. Smilde. Principal component analysis. *Analytical Methods*, 6:2812–2831, 2014.

[46] Herve Abdi and Lynne J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2, June 2010.

[47] George H Dunteman. *Principal components analysis*. Number 69. Sage, 1989.

[48] Alessandro Parente and James C. Sutherland. Principal component analysis of turbulent combustion data: Data pre-processing and manifold sensitivity. *Combust. Flame*, 160, 2013.

[49] Howard Anton and Chris Rorres. *Elementary linear algebra with supplementary applications*. John Wiley & Sons (Asia) Pte Ltd, 10th edition, 2011.

[50] Peter Läuchli. Jordan-elimination und ausgleichung nach kleinsten quadraten. *Numerische Mathematik*, 3(1):226–240, 1961.

[51] Ronald Aylmer Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 222:309–368, 1922.

[52] José A Bittencourt. *Fundamentals of plasma physics*. Springer Science & Business Media, 2013.

[53] WAAD Graef. *Zero-dimensional models for plasma chemistry*. PhD thesis, Eindhoven University of Technology, 2012.

[54] James C Sutherland and Alessandro Parente. Combustion modeling using principal component analysis. *Proceedings of the Combustion Institute*, 32(1):1563–1570, 2009.

[55] Derek Dalle. Comparison of numerical techniques for euclidean curvature. *Rose-Hulman Undergraduate Mathematics Journal*, 7(1):12, 2006.

[56] Isaac Amidror. Scattered data interpolation methods for electronic imaging systems: a survey. *Journal of electronic imaging*, 11(2):157–176, 2002.

[57] Boris Delaunay. Sur la sphere vide. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, 7(793-800):1–2, 1934.

[58] Benjamin J Isaac, Jeremy N Thornock, James Sutherland, Philip J Smith, and Alessandro Parente. Advanced regression methods for combustion modelling using principal components. *Combustion and flame*, 162(6):2592–2601, 2015.

[59] Amir Biglari and James C Sutherland. A filter-independent model identification technique for turbulent combustion modeling. *Combustion and Flame*, 159(5):1960–1970, 2012.

[60] Jerome H Friedman. Multivariate adaptive regression splines. *The annals of statistics*, pages 1–67, 1991.

[61] G. Jekabsons. Areslab: Adaptive regression splines toolbox for matlab/octave, 2016. available at http://www.cs.rtu.lv/jekabsons/.

[62] MJ Baxter. Standardization and transformation in principal component analysis, with applications to archaeometry. *Applied Statistics*, pages 513–527, 1995.

[63] M Hills. Allometry. *Encyclopedia of Statistical Sciences*.

[64] Pierre Jolicoeur. 193. note: the multivariate generalization of the allometry equation. *Biometrics*, 19(3):497–499, 1963.

[65] John Aitchison. Principal component analysis of compositional data. *Biometrika*, 70(1):57–65, 1983.

[66] Eric L Haseltine and James B Rawlings. Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *The Journal of chemical physics*, 117(15):6959–6969, 2002.

[67] S Pancheshnyi, B Eismann, GJM Hagelaar, and LC Pitchford. Computer code zdplaskin. In *11th Int. Symp. on High Pressure, Low Temperature Plasma Chemistry*, 2008.

[68] GJM Hagelaar and LC Pitchford. Solving the boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models. *Plasma Sources Science and Technology*, 14(4):722, 2005.

[69] Tiago Silva, Nikolay Britun, Thomas Godfroid, and Rony Snyders. Optical characterization of a microwave pulsed discharge used for dissociation of co2. *Plasma Sources Science and Technology*, 23(2):025009, 2014.

[70] Rasmus Bro, Evrim Acar, and Tamara G Kolda. Resolving the sign ambiguity in the singular value decomposition. *Journal of Chemometrics*, 22(2):135–140, 2008.

[71] S Rahimi, M Jimenez-Diaz, S Hübner, EH Kemaneci, JJAM Van der Mullen, and J Van Dijk. A two-dimensional modelling study of a coaxial plasma waveguide. *Journal of Physics D: Applied Physics*, 47(12):125204, 2014.

[72] Linda Petzold. Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations. *SIAM journal on scientific and statistical computing*, 4(1):136–148, 1983.

[73] Min-Hyong Lee, Sung-Ho Jang, and Chin-Wook Chung. On the multistep ionizations in an argon inductively coupled plasma. *Physics of plasmas*, 13(5):053502, 2006.

[74] Sumio Ashida, C Lee, and MA Lieberman. Spatially averaged (global) model of time modulated high density argon plasmas. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, 13(5):2498–2507, 1995.

[75] Fumihiko Kannari, Minoru Obara, and Tomoo Fujioka. An advanced kinetic model of electron-beam-excited krf lasers including the vibrational relaxation in krf*(b) and collisional mixing of krf*(b, c). *Journal of applied physics*, 57(9):4309–4322, 1985.

[76] Y P Raizer. *Gas discharge physics*. Springer Berlin, 1991.

[77] Y Kabouzi, DB Graves, E Castaños-Martínez, and M Moisan. Modeling of atmospheric-pressure plasma columns sustained by surface waves. *Physical Review E*, 75(1):016402, 2007.

[78] Lawrence F Shampine and Mark W Reichelt. The matlab ode suite. *SIAM journal on scientific computing*, 18(1):1–22, 1997.

[79] John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

[80] J. A. Manion, R. E. Huie, R. D. Levin, D. R. Burgess Jr., V. L. Orkin, W. Tsang, W. S. McGivern, J. W. Hudgens, V. D. Knyazev, D. B. Atkinson, E. Chai, A. M. Tereza, C.-Y. Lin, T. C. Allison, W. G. Mallard, F. Westley, J. T. Herron, R. F. Hampson, and D. H. Frizzell, editors. *NIST Standard Reference Database 17, Version 7.0 (Web Version)*. National Institute of Standards and Technology, Gaithersburg MD, 20899, 2015.

[81] Ángel Yanguas-Gil, José Cotrino, and Luís L Alves. An update of argon inelastic cross sections for plasma discharges. *Journal of Physics D: Applied Physics*, 38(10):1588, 2005.

[82] J Vlček. A collisional-radiative model applicable to argon discharges over a wide range of conditions. i. formulation and basic data. *Journal of Physics D: Applied Physics*, 22(5):623, 1989.

[83] O Zatsarinny and K Bartschat. B-spline breit–pauli r-matrix calculations for electron collisions with argon atoms. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 37(23):4693, 2004.

[84] A Kimura, H Kobayashi, M Nishida, and P Valentin. Calculation of collisional and radiative transition probabilities between excited argon levels. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 34(2):189–215, 1985.

[85] Donald Rapp and Paula Englander-Golden. Total cross sections for ionization and attachment in gases by electron impact. i. positive ionization. *The Journal of Chemical Physics*, 43(5):1464–1479, 1965.

[86] Mark A Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243, 1991.

[87] Dong Dong and Thomas J McAvoy. Nonlinear principal component analysis based on principal curves and neural networks. *Computers & Chemical Engineering*, 20(1):65–78, 1996.

[88] Jong-Min Lee, ChangKyoo Yoo, Sang Wook Choi, Peter A Vanrolleghem, and In-Beum Lee. Nonlinear process monitoring using kernel principal component analysis. *Chemical engineering science*, 59(1):223–234, 2004.

[89] Thomas M Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE transactions on electronic computers*, (3):326–334, 1965.

[90] Sebastian Mika, Bernhard Schölkopf, Alex J Smola, Klaus-Robert Müller, Matthias Scholz, and Gunnar Rätsch. Kernel pca and de-noising in feature spaces. In *Advances in neural information processing systems*, pages 536–542, 1999.

[91] J. J. Lowke, A. V. Phelps, and B. W. Irwin. Predicted electron transport coefficients and operating characteristics of CO_2N_2he laser mixtures. *Journal of Applied Physics*, 44(10):4664–4671, 1973.

[92] Y. Itikawa and A. Ichimura. Cross sections for collisions of electrons and photons with atomic oxygen. *Journal of Physical and Chemical Reference Data*, 19(3):637–651, 1990.

[93] James E. Land. Electron scattering cross sections for momentum transfer and inelastic excitation in carbon monoxide. *Journal of Applied Physics*, 49(12):5716–5721, 1978.

[94] A. V. Phelps. Tabulations of collision cross sections and calculated transport and reaction coefficients for electron collisions with O_2. (28):1–12, 09/01/1985 1985. JILA Pub. 3215.

[95] L.L. Alves. The ist-lisbon database on lxcat. *Journal of Physics: Conference Series*, 565(1):012007, 2014.

[96] Lindsay B G Mangan M A and Stebbings R F. *J. Phys.B At. Mol. Opt. Phys.*, 33:3225, 2000.

[97] Hirokazu Hokazono and Haruo Fujimoto. Theoretical analysis of the CO_2 molecule decomposition and contaminants yield in transversely excited atmospheric CO_2 laser discharge. *Journal of Applied Physics*, 62(5):1585–1594, 1987.

[98] R. E. Beverly. Ion aging effects on the dissociative-attachment instability in CO_2 lasers. *Optical and Quantum Electronics*, 14(6):501–513, 1982.

[99] T. Kozák and A. Bogaerts. Evaluation of the energy efficiency of CO_2 conversion in microwave discharges using a reaction kinetics model. *Plasma Sources Science Technology*, 24(1):015024, February 2015.

[100] A Eremin, ZIBOROV V. S.and SHUMOVA V. V., VOIKI D., and ROTH P. Formation of o(1d) atoms in thermal decomposition of CO_2. *Kinetics and catalysis*, 38:1, 1997.

[101] Thomas G. Beuthe and Jen-Shih Chang. Chemical kinetic modelling of non-equilibrium ar-CO_2 thermal plasmas. *Japanese Journal of Applied Physics*, 36(7S):4997, 1997.

[102] W. Tsang and R. F. Hampson. Chemical kinetic data base for combustion chemistry. part i. methane and related compounds. *Journal of Physical and Chemical Reference Data*, 15(3):1087–1279, 1986.

[103] A. Cenian, A. Chernukho, V. Borodin, and G. liwiski. Modeling of plasma-chemical reactions in gas mixture of CO_2 lasers i. gas decomposition in pure CO_2 glow discharge. *Contributions to Plasma Physics*, 34(1):25–37, 1994.

[104] W.L. Shackleford, F.N. Mastrup, and W.C. Kreye. Excitation and quenching of CO fourth positive chemiluminescence due to reactions involving C_2O. *J. Chem. Phys.*, 57, 1972.

[105] R. Atkinson, D.L. Baulch, R.A. Cox, Jr. Hampson, R.F., J.A. Kerr, M.J. Rossi, and J. Troe. Evaluated kinetic and photochemical data for atmospheric chemistry: supplement vi. iupac subcommittee on gas kinetic data evaluation for atmospheric chemistry. *J. Phys. Chem. Ref. Data*, 26:1329 – 1499, 1997.

[106] H. Hippler, R. Rahn, and J. Troe. Temperature and pressure dependence of ozone formation rates in the range 1 - 1000 bar and 90 - 370 K. *The Journal of Chemical Physics*, 93(9):6560–6569, 1990.

[107] Jay A. Blauer and Gary R. Nickerson. *A Survey of Vibrational Relaxation Rate Data for Processes Important to* CO_2-N_2-H_2O *Infrared Plume Radiation*. 1973.

[108] M. Capitelli, C.M. Ferreira, B.F. Gordiets, and A.I. Osipov. *Plasma Kinetics in Atmospheric Gases*. Springer Series on Atomic, Optical, and Plasma Physics. Springer Berlin Heidelberg, 2000.

[109] Thomas G. Kreutz, James A. O'Neill, and George W. Flynn. Diode laser absorption probe of vibration-vibration energy transfer in carbon dioxide. *The Journal of Physical Chemistry*, 91(22):5540–5543, 1987.

Appendix A

Nonlinear PCA

The assumption of linearity in PCA is a very powerful but limiting assumption. In some cases, PCA might perform poorly due to the linearity assumption. This can happen when the input data contains nonlinearities, such as in figure 2.4. Several nonlinear PCA methods have been developed that are able to handle nonlinear data. Examples are a nonlinear PCA method based on auto-associative neural network developed by Kramer [86], another nonlinear PCA approach based on pincipal curves and neural networks developed by Dong and McAvoy [87], but the most popular method is kernel PCA (KPCA) [88]. The basic idea of KPCA is to first map the input dat into a feature space F through nonlinear mapping and then to perform PCA in that feature space. According to Cover's theorem, the nonlinear structure of data is more likely to be linear after nonlinear mapping to a high-dimensional space [89]. Therefore, the feature space is usually chosen to have a much higher dimensionality than the input data.

The feature space F is related to the input space through a nonlinear map Φ. KPCA algorithms are usually written in such a way that it is necessary to calculate dot products of the variables $\boldsymbol{x}_i$ mapped into F, which have the form $\langle \Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j) \rangle$. The dot product is defined as as $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \boldsymbol{x}^{\mathrm{T}} \boldsymbol{y}$. The calculation of these dot products for all possible combinations of $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is very expensive and might not be computationally viable, because the dimensionality of F is much higher than the dimensionality of the original space. Therefore, KPCA uses kernels, which have the advantage that the mapping Φ does not need to be carried out explicitly. A kernel function has the following form:

$$k(\boldsymbol{x}, \boldsymbol{y}) = \langle \Phi(\boldsymbol{x}), \Phi(\boldsymbol{y}) \rangle \,. \tag{A.1}$$

Examples of kernel functions are a polynomial kernel:

$$k(\boldsymbol{x}, \boldsymbol{y}) = \langle \boldsymbol{x}, \boldsymbol{y} \rangle^d \,, \tag{A.2}$$

with d the power of the polynomial kernel, the sigmoid kernel

$$k(\boldsymbol{x}, \boldsymbol{y}) = \tanh\left(\beta_0 \langle \boldsymbol{x}, \boldsymbol{y} \rangle + \beta_1\right) \tag{A.3}$$

where β_0 and β_1 are constants and the radial basis kernel

$$k(\boldsymbol{x}, \boldsymbol{y}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{y}\|}{2\sigma^2}\right) \tag{A.4}$$

with constant σ.

A KPCA algorithm consists of three main steps that are very similar to (linear) PCA. Firstly, the kernel matrix $\boldsymbol{K}$ is computed, which contains elements $K_{ij} = \langle \Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j) \rangle$. The kernel matrix is the equivalent of the covariance matrix in PCA. In the second step, the eigenvectors of $\boldsymbol{K}$ are calculated. The eigenvalue problem in KPCA is slightly different than in PCA:

$$N\lambda_i \boldsymbol{a}_i = \boldsymbol{K}\boldsymbol{a}_i, \tag{A.5}$$

with N the dimensionality of the feature space F, λ_i the ith eigenvalue of $\boldsymbol{K}$ and $\boldsymbol{a}_i$ the ith eigenvector of $\boldsymbol{K}$. The eigenvectors are normalized in the following way:

$$\boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{a}_i = \lambda_i. \tag{A.6}$$

Finally, the PC scores are calculated by projecting $\Phi(\boldsymbol{x})$ onto the PCs in F:

$$\boldsymbol{z}_i = \sum_{j=1}^{N} a_{ji} \langle \Phi(\boldsymbol{x}_j), \Phi(\boldsymbol{x}) \rangle . \tag{A.7}$$

The main disadvantage of KPCA is that although it provides PCs that are nonlinearly related to the original variables, the reconstruction of the input data from the PC scores is difficult and troublesome [88, 90].

Appendix B

Data for the argon model with 78 levels

The data for the argon model with 78 levels, used in section 5.2. The data was taken from Graef [53].

Table B.1: A list of all the levels that were included in the argon model with 78 levels, used in section 5.2, together with their energy and statistical weight.

name	energy	weight	name	energy	weight
ground	0.000000	1			
$4s[3/2]_2$	11.548354	5	$4s[3/2]_1$	11.623592	3
$4s'[1/2]_0$	11.723160	1	$4s'[1/2]_1$	11.828071	3
$4p[1/2]_1$	12.907015	3	$4p[5/2]_3$	13.075715	7
$4p[5/2]_2$	13.094872	5	$4p[3/2]_1$	13.153143	3
$4p[3/2]_2$	13.171777	5	$4p[1/2]_0$	13.273038	1
$4p'[3/2]_1$	13.282638	3	$4p'[3/2]_2$	13.302227	5
$4p'[1/2]_1$	13.327856	3	$4p'[1/2]_0$	13.479886	1
$3d[1/2]_0$	13.845038	1	$3d[1/2]_1$	13.863668	3
$3d[3/2]_2$	13.903454	5	$3d[7/2]_4$	13.979237	9
$3d[7/2]_3$	14.012738	7	$3d[5/2]_2$	14.063027	5
$3d[5/2]_3$	14.099055	7	$3d[3/2]_1$	14.152514	3
$3d'[5/2]_2$	14.213671	5	$3d'[3/2]_2$	14.234022	5
$3d'[5/2]_3$	14.236105	7	$3d'[3/2]_1$	14.303668	3
$5s[3/2]_2$	14.068297	5	$5s[3/2]_1$	14.089968	3
$5s'[1/2]_0$	14.241027	1	$5s'[1/2]_1$	14.255085	3

name	energy	weight	name	energy	weight
$5p[1/2]_1$	14.463995	3	$5p[5/2]_3$	14.499053	7
$5p[5/2]_2$	14.506067	5	$5p[3/2]_1$	14.524913	3
$5p[3/2]_2$	14.528913	5	$5p[1/2]_0$	14.575948	1
$5p'[3/2]_1$	14.680650	3	$5p'[1/2]_1$	14.687118	3
$5p'[3/2]_2$	14.688290	5	$5p'[1/2]_0$	14.738115	1
6s	14.842395	8	6s'	15.020082	4
7s	15.182461	8	7s'	15.359150	4
8s	15.363559	8	8s'	15.541244	4
9s	15.470217	8	9s'	15.647963	4
10s	15.539020	8	10s'	15.716050	4
6p	15.027492	24	6p'	15.204578	12
7p	15.285607	24	7p'	15.441823	12
8p	15.418657	24	8p'	15.595913	12
9p	15.505837	24	9p'	15.687000	12
4d	14.780255	40	4d'	14.967431	20
5d	15.146943	40	5d'	15.316640	20
6d	15.343672	40	6d'	15.515428	20
7d	15.454865	40	7d'	15.632548	20
8d	15.530544	40	8d'	15.713471	20
4f	14.905574	56	4f'	15.083141	28
5f	15.213270	56	5f'	15.390949	28
6f	15.380265	56	6f'	15.557897	28
7f	15.481085	56	7f'	15.658571	28
ion	15.75961	6			

Table B.2: A list of the radiative transitions that were used for the argon model with 78 levels in section 5.2. For each transition the wavelength and transition probability is given.

upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$	upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$
4s[3/2]1	ground	106.7	119	4s'[1/2]1	ground	104.8	510

upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$	upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$
4p[1/2]1	4s'[1/2]0	1047.3	0.98	4p[1/2]1	4s'[1/2]1	1149.1	0.19
4p[1/2]1	4s[3/2]1	966.0	5.43	4p[1/2]1	4s[3/2]2	912.5	18.9
4p[5/2]3	4s[3/2]2	811.8	33.1	4p[5/2]2	4s'[1/2]1	978.7	1.47
4p[5/2]2	4s[3/2]1	842.7	21.5	4p[5/2]2	4s[3/2]2	801.7	9.28
4p[3/2]2	4s'[1/2]1	922.4	5.03	4p[3/2]2	4s[3/2]1	800.6	4.9
4p[3/2]2	4s[3/2]2	763.5	24.5	4p[3/2]1	4s'[1/2]0	867.0	2.43
4p[3/2]1	4s'[1/2]1	935.4	1.06	4p[3/2]1	4s[3/2]1	810.4	25.0
4p[3/2]1	4s[3/2]2	772.4	5.18	4p[1/2]0	4s[3/2]1	751.7	40.2
4p'[3/2]1	4s'[1/2]0	795.0	18.6	4p'[3/2]1	4s'[1/2]1	852.4	13.9
4p'[3/2]1	4s[3/2]1	747.3	0.022	4p'[3/2]1	4s[3/2]2	714.9	0.625
4p'[3/2]2	4s'[1/2]1	841.1	22.3	4p'[3/2]2	4s[3/2]1	738.6	8.47
4p'[3/2]2	4s[3/2]2	706.9	3.8	4p'[1/2]1	4s'[1/2]0	772.6	11.7
4p'[1/2]1	4s'[1/2]1	826.7	15.3	4p'[1/2]1	4s[3/2]1	727.5	1.83
4p'[1/2]1	4s[3/2]2	696.7	6.39	4p'[1/2]0	4s'[1/2]1	750.6	44.5
4p'[1/2]0	4s[3/2]1	667.9	0.236				
3d[1/2]0	4p[1/2]1	1321.8	8.1	3d[1/2]0	4p'[1/2]1	2397.3	0.36
3d[1/2]0	4p'[3/2]1	2204.6	0.12	3d[1/2]1	4p[1/2]1	1296.0	7.4
3d[1/2]1	4p'[1/2]1	2314.0	0.17	3d[1/2]1	4p'[3/2]1	2133.9	0.032
3d[1/2]1	4p'[3/2]2	2208.3	0.14	3d[1/2]1	4p[5/2]2	1612.7	0.039
3d[3/2]2	4p[1/2]1	1244.3	4.9	3d[3/2]2	4p'[1/2]1	2154.0	0.11
3d[3/2]2	4p[3/2]1	1694.5	0.26	3d[3/2]2	4p[3/2]2	1652.4	2.5
3d[3/2]2	4p'[3/2]2	2062.2	0.39	3d[3/2]2	4p[5/2]2	1533.4	0.12
3d[7/2]3	4p[3/2]2	1442.4	0.088	3d[7/2]3	4p[5/2]2	1350.8	11
3d[5/2]2	4p[3/2]1	1391.1	7.3	3d[5/2]2	4p[5/2]2	1280.6	5.7
3d[5/2]2	4p[5/2]3	1255.8	0.12	3d[5/2]3	4p'[3/2]2	1556.0	0.0098
3d[5/2]3	4p[5/2]2	1234.7	2	3d[5/2]3	4p[5/2]3	1211.6	3.1
3d[3/2]1	4p[1/2]0	1409.7	4.3	3d[3/2]1	4p[3/2]1	1264.2	11
3d[3/2]1	4p[5/2]2	1172.3	0.952	3d[3/2]1	ground	87.6	270
3d'[5/2]2	4p'[3/2]1	1331.7	13	3d'[5/2]2	4p'[3/2]2	1360.3	2.2
3d'[5/2]2	4p[5/2]2	1108.2	0.83	3d'[3/2]2	4p'[1/2]1	1368.2	6.2
3d'[3/2]2	4p[3/2]1	1167.2	0.369	3d'[3/2]2	4p[3/2]2	1147.1	3.76
3d'[5/2]3	4p'[3/2]2	1327.6	15	3d'[3/2]1	4p[1/2]0	1203.0	0.42
3d'[3/2]1	4p'[1/2]0	1505.1	5.2	3d'[3/2]1	4p'[1/2]1	1270.6	7.1
3d'[3/2]1	4p'[3/2]1	1214.3	4.5	3d'[3/2]1	4p[3/2]2	1077.6	0.396
3d'[3/2]1	ground	86.7	313				
5s[3/2]2	4p[1/2]1	1067.6	4.9	5s[3/2]2	4p'[1/2]1	1674.5	0.31

upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$	upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$
5s[3/2]2	4p[3/2]1	1382.9	0.46	5s[3/2]2	4p'[3/2]1	1578.1	0.059
5s[3/2]2	4p[3/2]2	1354.8	3.3	5s[3/2]2	4p'[3/2]2	1618.4	0.12
5s[3/2]2	4p[5/2]2	1273.7	1.1	5s[3/2]2	4p[5/2]3	1249.1	11
5s[3/2]1	4p[1/2]0	1517.7	1.3	5s[3/2]1	4p'[1/2]0	2032.3	0.16
5s[3/2]1	4p[1/2]1	1048.1	2.44	5s[3/2]1	4p'[1/2]1	1626.9	0.03
5s[3/2]1	4p[3/2]1	1350.3	4.6	5s[3/2]1	4p'[3/2]1	1535.7	0.45
5s[3/2]1	4p[3/2]2	1323.5	2.7	5s[3/2]1	4p'[3/2]2	1573.9	0.029
5s[3/2]1	4p[5/2]2	1246.0	8.9	5s[3/2]1	ground	88.0	77
5s'[1/2]0	4p'[1/2]1	1357.7	5.1	5s'[1/2]0	4p[1/2]1	929.4	3.26
5s'[1/2]0	4p[3/2]1	1159.5	2.22	5s'[1/2]0	4p'[3/2]1	1293.7	10
5s'[1/2]1	4p[1/2]0	1262.5	0.38	5s'[1/2]1	4p'[1/2]0	1599.4	1.9
5s'[1/2]1	4p'[1/2]1	1337.1	3.4	5s'[1/2]1	4p[1/2]1	919.7	1.76
5s'[1/2]1	4p[3/2]1	1144.5	0.28	5s'[1/2]1	4p'[3/2]1	1275.0	2.0
5s'[1/2]1	4p[3/2]2	1125.1	1.39	5s'[1/2]1	4p'[3/2]2	1301.2	8.9
5s'[1/2]1	4p[5/2]2	1068.6	0.21	5s'[1/2]1	ground	87.0	35
5p[1/2]1	4s'[1/2]0	452.4	0.0898	5p[1/2]1	4s'[1/2]1	470.4	0.109
5p[1/2]1	4s[3/2]1	436.5	0.012	5p[1/2]1	4s[3/2]2	425.2	0.111
5p[5/2]3	3d[3/2]2	2081.7	0.076	5p[5/2]3	3d[7/2]4	2385.2	1.1
5p[5/2]3	4s[3/2]2	420.2	0.967	5p[5/2]2	4s'[1/2]1	463.0	0.0383
5p[5/2]2	4s[3/2]1	430.1	0.377	5p[5/2]2	4s[3/2]2	419.2	0.28
5p[3/2]1	4s'[1/2]0	442.5	0.0073	5p[3/2]1	4s'[1/2]1	459.7	0.0947
5p[3/2]1	4s[3/2]1	427.3	0.797	5p[3/2]1	4s[3/2]2	416.5	0.288
5p[3/2]2	4s'[1/2]1	459.1	0.0062	5p[3/2]2	4s[3/2]1	426.7	0.312
5p[3/2]2	4s[3/2]2	416.0	1.4	5p[1/2]0	4s'[1/2]1	451.2	1.18
5p[1/2]0	4s[3/2]1	419.9	2.57	5p'[3/2]1	4s'[1/2]0	419.2	0.539
5p'[3/2]1	4s'[1/2]1	434.6	0.297	5p'[3/2]1	4s[3/2]1	405.6	0.027
5p'[1/2]1	3d[3/2]2	1582.1	0.087	5p'[1/2]1	4s'[1/2]0	418.3	0.561
5p'[1/2]1	4s'[1/2]1	433.7	0.387	5p'[1/2]1	4s[3/2]1	404.7	0.041
5p'[1/2]1	4s[3/2]2	395.0	0.455	5p'[3/2]2	4s'[1/2]1	433.5	0.568
5p'[3/2]2	4s[3/2]1	404.6	0.333	5p'[3/2]2	4s[3/2]2	394.9	0.056
5p'[1/2]0	4s'[1/2]1	426.1	3.98				
4d	4p[1/2]0	822.6	0.069	4d	4p[1/2]1	661.9	0.53
4d	4p'[1/2]1	853.7	0.16	4d	4p[3/2]1	770.8	0.094
4d	4p'[3/2]1	827.9	0.018	4d	4p[3/2]2	762.0	0.19
4d	4p'[3/2]2	838.8	0.21	4d	4p[5/2]2	735.6	0.20
4d	4p[5/2]3	727.4	0.48				

upper level	lower level	λ [nm]	A [$10^6\,s^{-1}$]	upper level	lower level	λ [nm]	A [$10^6\,s^{-1}$]
6s	4p[1/2]0	790.0	0.13	6s	4p'[1/2]0	910.0	0.036
6s	4p[1/2]1	640.6	0.88	6s	4p'[1/2]1	818.6	0.11
6s	4p[3/2]1	742.1	0.77	6s	4p'[3/2]1	794.9	0.066
6s	4p[3/2]2	734.0	0.83	6s	4p'[3/2]2	805.0	0.089
6s	4p[5/2]2	709.5	1.03	6s	4p[5/2]3	701.8	1.67
6s'	4p[1/2]0	709.7	0.11	6s'	4p'[1/2]0	805.0	0.27
6s'	4p[1/2]1	586.7	0.52	6s'	4p'[1/2]1	732.7	1.02
6s'	4p[3/2]1	670.8	0.23	6s'	4p'[3/2]1	713.6	0.98
6s'	4p[3/2]2	664.1	0.12	6s'	4p'[3/2]2	721.7	1.86
6s'	4p[5/2]2	644.0	0.038				
7s	4p[1/2]0	649.3	0.035	7s	4p[1/2]1	544.9	0.37
7s	4p[3/2]1	616.6	0.22	7s	4p'[3/2]1	652.6	0.020
7s	4p[3/2]2	611.0	0.50	7s	4p'[3/2]2	659.4	0.023
7s	4p[5/2]2	593.9	0.53	7s	4p[5/2]3	588.5	0.81
7s'	4p[1/2]0	594.3	0.09	7s'	4p[1/2]1	505.6	0.48
7s'	4p'[1/2]1	610.4	0.33	7s'	4p[3/2]1	566.8	0.25
7s'	4p'[3/2]1	597.1	0.41	7s'	4p'[3/2]2	602.8	0.68
7s'	4p[5/2]2	547.6	0.15				
8s	4p[1/2]1	504.7	0.29	8s	4p'[1/2]1	609.0	0.028
8s	4p'[3/2]1	595.8	0.056	8s	4p[3/2]2	560.9	0.21
8s	4p'[3/2]2	601.5	0.052	8s	4p[5/2]2	546.5	0.18
8s	4p[5/2]3	541.9	0.38	8s'	4p[3/2]1	523.3	0.20
8s'	4p'[3/2]1	548.9	0.14	8s'	4p'[3/2]2	553.7	0.20
9s	4p[1/2]0	564.3	0.079	9s	4p[1/2]1	483.7	0.064
9s	4p[3/2]2	535.1	0.060	9s	4p[5/2]2	522.0	0.049
9s	4p[5/2]3	517.8	0.15				
10s	4p[5/2]2	507.3	0.098	10s	4p[5/2]3	503.3	0.051
6p	4s'[1/2]0	375.2	0.0088	6p	4s'[1/2]1	387.5	0.038
6p	4s[3/2]1	364.2	0.072	6p	4s[3/2]2	356.4	0.088
6p'	4s'[1/2]0	356.1	0.030	6p'	4s'[1/2]1	367.2	0.092
6p'	4s[3/2]1	346.2	0.028				
7p	4s'[1/2]1	358.6	0.021	7p'	4s'[1/2]1	343.1	0.033

upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$	upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$
4d'	4p[1/2]0	731.7	0.0087	4d'	4p[1/2]1	601.7	0.31
4d'	4p'[1/2]1	756.2	0.15	4d'	4p[3/2]1	690.5	0.11
4d'	4p'[3/2]1	735.9	0.11	4d'	4p[3/2]2	683.4	0.22
4d'	4p'[3/2]2	744.6	0.24	4d'	4p[5/2]2	662.1	0.047
4d'	4p[5/2]3	655.4	0.11				
5d	4p[1/2]0	661.6	0.029	5d	4p[1/2]1	553.5	0.42
5d	4p'[1/2]1	681.6	0.038	5d	4p[3/2]1	627.7	0.11
5d	4p'[3/2]1	665.0	0.026	5d	4p[3/2]2	621.8	0.18
5d	4p'[3/2]2	672.1	0.069	5d	4p[5/2]2	604.2	0.29
5d	4p[5/2]3	598.6	0.60	5d'	4p[1/2]0	606.7	0.012
5d'	4p[1/2]1	514.5	0.37	5d'	4p'[1/2]1	623.4	0.23
5d'	4p[3/2]1	578.1	0.24	5d'	4p'[3/2]1	609.6	0.30
5d'	4p[3/2]2	573.1	0.21	5d'	4p'[3/2]2	615.5	0.43
5d'	4p[5/2]2	558.0	0.27	5d'	4p[5/2]3	553.3	0.084
6d	4p[1/2]0	598.8	0.023	6d	4p'[1/2]0	665.2	0.0091
6d	4p[1/2]1	508.8	0.21	6d	4p'[1/2]1	615.1	0.016
6d	4p[3/2]1	570.9	0.023	6d	4p'[3/2]1	601.6	0.015
6d	4p[3/2]2	566.0	0.19	6d	4p'[3/2]2	607.3	0.015
6d	4p[5/2]2	551.3	0.074	6d	4p[5/2]3	546.7	0.40
6d'	4p[1/2]1	475.3	0.22	6d'	4p[3/2]1	529.0	0.090
6d'	4p'[3/2]1	555.3	0.067	6d'	4p[3/2]2	524.8	0.064
6d'	4p'[3/2]2	560.2	0.18	6d'	4p[5/2]2	512.2	0.10
6d'	4p[5/2]3	508.2	0.016				
7d	4p[1/2]0	568.3	0.013	7d	4p[1/2]1	486.6	0.24
7d	4p'[1/2]1	582.9	0.0041	7d	4p[3/2]1	543.1	0.034
7d	4p[3/2]2	538.7	0.035	7d	4p'[3/2]2	576.0	0.016
7d	4p[5/2]2	525.4	0.13	7d	4p[5/2]3	521.1	0.22
7d'	4p[1/2]1	454.9	0.0095	7d'	4p'[3/2]2	532.0	0.091
8d	4p[1/2]0	549.2	0.0090	8d	4p[1/2]1	472.6	0.043
8d	4p'[3/2]1	551.6	0.0059	8d	4p[3/2]2	521.5	0.038
8d	4p[5/2]2	509.0	0.028	8d	4p[5/2]3	505.1	0.083
4f	3d[1/2]0	1169.1	0.036	4f	3d[1/2]1	1190.0	0.66
4f	3d[3/2]1	1646.4	0.76	4f	3d[3/2]2	1237.2	0.75
4f	3d'[5/2]2	1791.9	0.11	4f	3d[5/2]3	1537.3	1.77

upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$	upper level	lower level	λ [nm]	A $[10^6\,s^{-1}]$
4f	3d[7/2]4	1338.4	2.3	4f	5s[3/2]2	1480.8	0.053
4f'	3d'[3/2]1	1590.6	1.38	4f'	3d[3/2]2	1051.0	0.68
4f'	3d'[3/2]2	1460.2	2.3	4f'	3d'[5/2]3	1463.7	5.1
4f'	3d[7/2]3	1158.3	0.20				

Appendix C

Electron impact ionization and excitation reactions

Electron impact ionization and excitation reactions for the CO_2 microwave model. This table was taken from the work by Koelman et al. [6]. All ionic species except CO_2^+ were removed as well as the C and C_2 species. The reaction numbers were not changed.

Table C.1: The electron impact ionization and excitation reactions in this model, with the corresponding ID and reference from which the data originates. For the reaction ID is unchanged with respect to [9]. For an added reaction the ID ends with an additional a. Most, but not all, of the reactions are described by a cross section.

No.	Reaction	Ref.
X1	$e^- + CO_2 \rightarrow e^- + CO_2$	[91] [a]
X2	$e^- + CO_2 \rightarrow e^- + e^- + CO_2^+$	[91] [a]
X4	$e^- + CO_2 \rightarrow e^- + e^- + C^+ + O_2$	[92] [b]
X8	$e^- + CO_2 \rightarrow e^- + CO + O$	[92] [b]
X9	$e^- + CO_2 \rightarrow e^- + CO_2[e_1]$	[91] [a]
X10	$e^- + CO_2 \rightarrow e^- + CO_2[e_2]$	[91] [a]
X11	$e^- + CO_2 \rightarrow e^- + CO_2[v_a]$	[91]
X12	$e^- + CO_2 \rightarrow e^- + CO_2[v_b]$	[91]
X13	$e^- + CO_2 \rightarrow e^- + CO_2[v_c]$	[91]
X14	$e^- + CO_2 \rightarrow e^- + CO_2[v_d]$	[91]
X15	$e^- + CO_2 \rightarrow e^- + CO_2[v_1]$	[91] [c]
X16	$e^- + CO \rightarrow e^- + CO$	[93] [a]
X21	$e^- + CO \rightarrow e^- + CO[e_1]$	[93] [a]
X22	$e^- + CO \rightarrow e^- + CO[e_2]$	[93] [a]
X23	$e^- + CO \rightarrow e^- + CO[e_3]$	[93] [a]
X24	$e^- + CO \rightarrow e^- + CO[e_4]$	[93] [a]
X25	$e^- + CO \rightarrow e^- + CO[v_1]$	[93] [c]

No.	Reaction	Ref.
X31	$e^- + O_2 \rightarrow e^- + O_2$	[94] [a]
X32	$e^- + O_2 \rightarrow e^- + O + O$	[92] [b]
X36	$e^- + O_2 \rightarrow e^- + O_2[v_1]$	[95]
X37	$e^- + O_2 \rightarrow e^- + O_2[v_2]$	[95]
X38	$e^- + O_2 \rightarrow e^- + O_2[v_3]$	[95]
X39	$e^- + O_2 \rightarrow e^- + O_2[e_1]$	[95] [a]
X40	$e^- + O_2 \rightarrow e^- + O_2[e_2]$	[95] [a]
X41	$e^- + O_3 \rightarrow e^- + O_3$	[96]
X42	$e^- + O_3 \rightarrow e^- + O_2 + O$	[9]
X47	$e^- + O \rightarrow e^- + O$	[96]

[a] The same cross section is used for the vibrationally excited species.

[b] The cross section is modified according to equation (4) of [9] for vibrationally excited species. For electronically excited species the energy data from the LUT is shifted with the difference in energy between the species in the ground state and the electronically excited state. Concequently the threshold energy of the process equals the threshold energy in the (modified) LUT.

[c] The cross section is modified according to equation (4) of [9] for vibrationally excited species.

Appendix D

Electron attachment and electron-ion recombination reactions.

Electron attachment and electron-ion recombination reactions for the CO_2 microwave model [6]. All ionic species except CO_2^+ were removed as well as the C and C_2 species. The reaction numbers were not changed.

Table D.1: Electron attachment and electron-ion recombination reactions. The reported rate coefficients have the units $m^3\,s^{-1}$ or m^6/s, with the gas temperature T_g in K and the electron temperature T_e in eV.

No.	Reaction	Rate coefficient	Ref
E1	$e^- + CO_2^+ \rightarrow CO[v_1] + O$	$2.00 \cdot 10^{-11}\, T_e^{-0.5}\, T_g^{-1}$	[97, 98]

Appendix E

Neutral-neutral interactions

Neutral-neutral interactions for the CO_2 microwave model, developed by Koelman et al. [6]. All ionic species except CO_2^+ were removed as well as the C and C_2 species. The reaction numbers were not changed.

Table E.1: The neutral-neutral interactions with the rate coefficients as they are included in the model, in units of $m^3\,s^{-1}$ and $m^3\,s^{-1}$. The coefficient α originates from [99], where the values are presented as estimates. The reactions and rate coefficients that are followed by an asterisk are discussed in more detail in [6].

No.	Reaction	rate	α	Ref
N1	$CO_2 + M \rightarrow CO + O + M$	$1.81 \cdot 10^{-16} \exp(-49000/T_g)$	0.8	[100, 80]
N2	$CO_2 + O \rightarrow CO + O_2$	$2.8 \cdot 10^{-17} \exp(-26500/T_g)$	0.5	[101, 80, 102]
N4	$O + CO + M \rightarrow CO_2 + M$	$8.2 \cdot 10^{-46} \exp(-1510/T_g) \cdot A$ [a]	0.0	[103, 80]
N5	$O_2 + CO \rightarrow CO_2 + O$	$4.2 \cdot 10^{-18} \exp(-24000/T_g)$	0.5	[101, 80]
N6	$O_3 + CO \rightarrow CO_2 + O_2$	$\leq 4.0 \cdot 10^{-31}$		[101, 80]
N10	$O + C_2O \rightarrow CO + CO$	$9.51 \cdot 10^{-17}$		[104]
N11	$O_2 + C_2O \rightarrow CO_2 + CO$	$3.3 \cdot 10^{-19}$		[103]
N12	$O + O_3 \rightarrow O_2 + O_2$	$8.0 \cdot 10^{-18} \exp(-2056/T_g)$		[80, 105]
N13	$O_3 + M \rightarrow O_2 + O + M$	$4.12 \cdot 10^{-16} \exp(-11430/T_g)$		[101]
N14	$O + O_2 + M \rightarrow O_3 + M$	$5.51 \cdot 10^{-46} (T_g/298)^{-2.6}$		[106]
N15	$O + O + M \rightarrow O_2 + M$	$5.2 \cdot 10^{-47} \exp(900/T_g)$		[102]

[a] $A = 2$, 1, 1 for M = CO_2, O_2 and CO, respectively.

Appendix F

VV and VT reactions

VV energy transfer reactions (vibrational energy is exchanged between two species) and VT reactions (vibrational energy is transferred to heat) as were given in Koelman et al. [6].

Table F.1: The VV and VT reactions of CO_2, CO and O_2, with the corresponding rate coefficient, obtained from [9]. The anharmonicity parameter x_e is required when applying the VV and VT rate coefficient scaling laws, see [6].

No.	Rate coefficient ($m^3\,s^{-1}$)	$x_e(\cdot 10^{-3})$	Ref	Note
V1	$CO_2v_a + M \rightarrow CO_2 + M$	0.0	[107]	a
	$7.14 \cdot 10^{-14} \exp(-177\,T_g^{-1/3} + 451\,T_g^{-2/3})$			
V2a	$CO_2v_1 + M \rightarrow CO_2v_a + M$	3.7	[107]	b
	$4.25 \cdot 10^{-7} \exp(-407\,T_g^{-1/3} + 824\,T_g^{-2/3})$			
V2b	$CO_2v_1 + M \rightarrow CO_2v_b + M$	1.0	[107]	b
	$8.57 \cdot 10^{-7} \exp(-404\,T_g^{-1/3} + 1096\,T_g^{-2/3})$			
V2c	$CO_2v_1 + M \rightarrow CO_2v_c + M$	-15.6	[107]	b
	$1.43 \cdot 10^{-11} \exp(-252\,T_g^{-1/3} + 685\,T_g^{-2/3})$			
V3	$COv_1 + M \rightarrow CO + M$	6.13	[108]	c
	$1.0 \cdot 10^{-18}\,T_g \exp(-150.7\,T_g^{-1/3})$			
V4	$COv_1 + O_2 \rightarrow CO + O_2$	6.13	[107]	
	$3.19 \cdot 10^{-12} \exp(-289\,T_g^{-1/3})$			
V5	$O_2v_1 + M \rightarrow O_2 + M$	0.0	[107]	d
	$1.30 \cdot 10^{-14} \exp(-158\,T_g^{-1/3})$			
V6	$O_2v_1 + O_2 \rightarrow O_2 + O_2$	0.0	[108]	
	$1.35 \cdot 10^{-18}\,T_g \exp(-137.9\,T_g^{-1/3})[1 - \exp(-2273/T_g)]^{-1}$			
V7a	$CO_2v_1 + CO_2 \rightarrow CO_2v_b + CO_2v_a$	2.8	[107]	
	$1.06 \cdot 10^{-11} \exp(-242\,T_g^{-1/3} + 633\,T_g^{-2/3})$			
V7b	$CO_2v_1 + CO_2 \rightarrow CO_2v_a + CO_2v_b$	17.6	[107]	
	$1.06 \cdot 10^{-11} \exp(-242\,T_g^{-1/3} + 633\,T_g^{-2/3})$			
V8	$CO_2v_1 + CO_2 \rightarrow CO_2 + CO_2v_1$	5.25	[109]	
	$1.32 \cdot 10^{-16}(T_g/300)^{0.5}\,250/T_g$			
V9	$COv_1 + CO \rightarrow CO + COv_1$	6.13	[108]	
	$3.4 \cdot 10^{-16}(T_g/300)^{0.5}\,(1.64 \cdot 10^{-6}\,T_g + 1.61/T_g)$			
V10	$CO_2v_1 + CO \rightarrow CO_2 + CO_2v_1$	5.25; 6.13	[107]	
	$4.8 \cdot 10^{-18} \exp(-153\,T_g^{-2/3})$			

[a] The rate coefficient is multiplied with 1.0, 0.7 and 0.7 for CO_2, CO and O_2, respectively.
[b] The rate coefficient is multiplied with 1.0, 0.3 and 0.4 for CO_2, CO and O_2, respectively.
[c] The same rate coefficient for M = CO_2 and CO.
[d] The rate coefficient is multiplied with 0.3 and 1.0 for M = CO_2 and CO, respectively.