TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

INSTITUT FÜR MECHANIK UND MECHATRONIK
Mechanics & Mechatronics

Diploma Thesis

Pantograph Modeling for Control

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Diplom-Ingenieurs
unter der Leitung von

Ao.Univ.Prof. Dr. Martin Kozek
Institut für Mechanik und Mechatronik
E325 A5

eingereicht an der Technischen Universität Wien

Fakultät für Maschinenwesen und Betriebswissenschaften

von

Clemens Carl Maier
Matr.Nr.: 0525072
Grüngasse 15/23
1050 Wien

Wien, am 21. Oktober 2015

Clemens Carl Maier

Eidesstattliche Erklärung

Ich erkläre eidesstattlich, dass ich die Arbeit selbständig angefertigt, keine anderen als die angegebenen Hilfsmittel benutzt und alle aus ungedruckten Quellen, gedruckter Literatur oder aus dem Internet im Wortlaut oder im wesentlichen Inhalt übernommenen Formulierungen und Konzepte gemäß den Richtlinien wissenschaftlicher Arbeiten zitiert, durch Fußnoten gekennzeichnet bzw. mit genauer Quellenangabe kenntlich gemacht habe.

Wien, am 21. Oktober 2015

Clemens Carl Maier

Danksagung

Danken möchte in erster Linie meinem Betreuer, Herrn Dr. Alexander Schirrer, welcher jederzeit zu konstruktiven Diskussionen bereit stand und daher großen Anteil daran trägt, dass meine Arbeit nun in dieser Form vorliegt. Durch diese Zusammenarbeit konnte ich einerseits wissenschaftliches Forschen erleben und, über die Inhalte dieser Arbeit hinaus, viel Neues erlernen. Des Weiteren möchte ich mich bei der gesamten Arbeitsgruppe Regelungstechnik und Prozessautomatisierung bedanken, für die unkomplizierte Unterstützung, sowie die freundliche Aufnahme in ihrer Gemeinschaft während meiner Zeit hier am Institut für Mechanik und Mechatronik.

Nicht zuletzt gebührt meinen Eltern und meiner Freundin Dank, ohne deren Unterstützung dieses Unterfangen nicht durchführbar gewesen wäre.

Kurzfassung

Die Diplomarbeit Pantograph Modeling for Control stellt zwei unterschiedliche echtzeitfähige Pantographenmodelle vor, welche hinsichtlich einer Anwendung in einer Pantographen-Oberleitungs-Co-Simulation einerseits, und für den Pantographen Reglerentwurf andererseits entwickelt wurden. Diese Modelle wurden durch eine Modell-auf-Modell Identifikation bestimmt, wobei ein existierendes White Box Modell des Pantographen aus [1] als Referenz herangezogen wurde. Des Weiteren sind beide Modelle als lokal-lineare Neuro-Fuzzy Netzwerke in Zustandsraumkonfiguration strukturiert. In der ersten Herangehensweise basiert das lineare Modellnetzwerk auf parametrisierten mechanischen Ersatzmodellen (Dreimassenschwinger), deren Parameter durch eine Ausgangsfehlermethode geschätzt wurden. Das erhaltene Pantographenmodell - pantograph LLMN (surrogate) - ist physikalisch interpretierbar und enthält strukturierte Matrizen des Zustandsraumes, welche mit Hilfe der Parameterverblendung interpoliert werden. Ein alternativer Lösungsansatz war durch das Bestreben motiviert, mögliche Synergien der Unterraummethoden der Identifikation und der lokalen Modellnetzwerke in Hinblick auf nichtlineare Modellbildung durch deren Zusammenführung zu nützen. Das dadurch erhaltene Pantographenmodell - pantograph LLMN (n4sid) - bietet eine hohe Genauigkeit über den gesamten Betriebsbereich und beinhaltet unstrukturierte Zustandsraum-Matrizen, wobei der Systemausgang durch die Anwendung der Ausgangsverblendung interpoliert wird. Eine umfassended qualitative Analyse sowie Simulationsergebnisse aller untersuchten Modelle heben das Potential dieser Modelle als Alternative zu globalen linearen Pantographenmodellen in echtzeitfähigen Pantographen-Oberleitungs-Co-Simulationen sowie beim Pantographen Reglerentwurf hervor.

Schlüsselwörter

Nichtlineare Modellierung, Pantograph; Echtzeitfähigkeit; lokal lineares Neuro-Fuzzy Netzwerk; Zustandsraumdarstellung; MIMO-System; Dreimassenschwinger; Unterraummethoden der Identifikation; N4SID;

Abstract

The thesis Pantograph Modeling for Control presents two different real-time capable pantograph models, intended for an application in pantograph-catenary co-simulations and pantograph control design. These models were derived from model-on-model identification referencing an existing nonlinear white-box pantograph model from [1] and are both structured as local-linear neuro-fuzzy networks in state-space configuration. In the first approach the local linear models of the model network are based on parametrized mechanical surrogate models (three-mass oscillator), whose parameters were identified by utilizing an output error method. The obtained pantograph model - pantograph LLMN (surrogate) - can be interpreted physically and contains structured state-space system matrices that are interpolated by utilizing the parameter blending method. In another approach an attempt was made to incorporate the subspace identification method N4SID into the local linear model network with the aim of exploiting the strengths of both methods in regard to nonlinear modeling. The obtained pantograph model - pantograph LLMN (n4sid) - provides high accuracy over the whole operating height and inherits unstructured state-space system matrices utilizing the output blending method to generate the system output. An extensive qualitative analysis as well as simulation results of all examined models are provided, emphasizing these models' potential to replace global linear pantograph models in real-time pantograph-catenary wire co-simulations and pantograph control design.

Keywords

Nonlinear modeling; pantograph; real-time capability; local linear neuro-fuzzy network; state-space system; MIMO system; three-mass-oscillator; subspace identification; N4SID;

Contents

1 Introduction **1**

1.1 Motivation . . . 2

1.1.1 State-of-the-art Pantograph System . . . 2

1.1.2 Limits in practical application of today's pantograph system . 3

1.1.3 Modern-Day Feedback Control . . . 3

1.2 Problem Statement . . . 4

1.2.1 Pantograph Model Specifications . . . 6

1.2.2 Literature Review . . . 7

2 Pantograph Modeling **8**

2.1 Existing Pantograph White-Box Model . . . 8

2.1.1 Static Examinations of the Nonlinear Pantograph . . . 12

2.1.2 Global Linear Model . . . 12

2.1.3 Dynamic Examinations of the Nonlinear Pantograph . . . 14

2.2 Developed Pantograph Model . . . 19

2.2.1 Global Inputs and Outputs of the Developed Pantograph Model 20

2.2.2 Local Linear Model Network (LLMN) Setup . . . 21

2.2.3 Local Linear Model (LLM) Setup . . . 22

3 Methodologies of Nonlinear Modeling and Identification **24**

3.1 Introduction . . . 24

3.2 Local Linear Model Network (LLMN) . . . 26

3.2.1 Neural Networks . . . 27

3.2.2 Fuzzy Systems . . . 33

3.2.3 Local Linear Neuro-Fuzzy Model . . . 36

3.2.4 Local Linear Model Tree (LOLIMOT) Algorithm . . . 38

3.3 Operating Regime Decomposition, Partitioning . . . 40

3.4 Surrogate Model (Parametrized Three-Mass Oscillator) . . . 45

3.4.1 Modeling Issues with Parametrized Models . . . 46

3.4.2 Equations of Motion of the Three-Mass Oscillator . . . 47

3.4.3 Equations of Motion in State-Space Form . . . 49

3.4.4 Parameter Vector . . . 51

3.4.5 Output Error (OE) Optimization . . . 52

3.5 Subspace Identification Methods . . . 53

3.5.1 Introduction to Subspace Identification Methods . . . 53

3.5.2 Subspace Identification . . . 59
3.5.3 State-Space Model with N4SID . . . 70
3.5.4 Modeling Issues with Subspace ID Methods . . . 72
3.6 LLMN Blending Methods . . . 72
3.6.1 Output Blending . . . 73
3.6.2 Parameter Blending . . . 75
3.6.3 Stability Considerations regarding LLMNs in State-Space Configuration . . . 76

4 Numeric Studies of Pantograph Models 78
4.1 Introduction . . . 78
4.2 Excitation Signals . . . 78
4.2.1 Noise excitation . . . 81
4.2.2 Input Data Sets . . . 82
4.3 LOLIMOT algorithm with NMSS systems . . . 85
4.3.1 Performance LOLIMOT Algorithm with NMSS systems . . . 85
4.4 Pantograph LLMN (surrogate) . . . 89
4.4.1 Simulation Results, Pantograph LLMN (surrogate) . . . 91
4.4.2 Frequency Analysis, Pantograph LLMN (surrogate) . . . 99
4.4.3 Stability Analysis, Pantograph LLMN (surrogate) . . . 102
4.5 Pantograph LLMN (n4sid) . . . 103
4.5.1 Simulation Results, Pantograph LLMN (n4sid) . . . 105
4.5.2 Frequency Analysis, Pantograph LLMN (n4sid) . . . 113
4.5.3 Stability Analysis, Pantograph LLMN (n4sid) . . . 115
4.6 Computational Efficiency and FIT . . . 116
4.7 Comparison of the Pole-Positions . . . 119
4.8 Comparision of the Examined Pantograph Models . . . 120

5 Observations and Discussion 124
5.1 Observations . . . 124
5.1.1 Pantograph LLMN (surrogate), Parameter Influence and Fisher Information . . . 124
5.1.2 Stability Considerations . . . 128
5.1.3 Discussion on Model Simplifications . . . 128
5.1.4 Subspace Identification using N4SID . . . 129
5.1.5 Potential Improvements, Further Research . . . 130
5.2 Discussion . . . 131
5.3 Main Statements, Conclusion . . . 132
5.3.1 Conclusion from a methodical point of view (scientist) . . . 133
5.3.2 Conclusion from a practical point of view (engineer) . . . 133

Bibliography 136

List of Figures

1.1 State-of-the-art pantograph. 2
1.2 Superordinated pantograph modeling problem. 4
1.3 Pantograph test bed. 7

2.1 Pantograph white-box model components and terms. 10
2.2 Pantograph white-box model DOF and parameters. 11
2.3 Pantograph white-box model I/O configuration. 12
2.4 Pantograph white-box model mapping of the static quantities. 13
2.5 Pantograph white-box model DFT of the simulation result for the pan-head velocity $\dot{\xi}$ 15
2.6 Pantograph white-box model DFT of the simulation result for the crossbar velocity $\dot{\zeta}$ 16
2.7 Pantograph white-box model DFT of the simulation result for the contact force F_p 16
2.8 Linearized pantograph white-box model pole positioning ($\mathcal{Z}$-plane). . 18
2.9 Pantograph LLMN in state-space configuration illustration. 19
2.10 Pantograph LLMN I/O configuration. 20
2.11 Dimension of movement of pantograph models (WBM versus LLMN). 22

3.1 Overview applied methods. 25
3.2 MISO neural network with one hidden layer scheme. 28
3.3 Comparison of neural network architectures. 31
3.4 Comparison of the i-th neuron of a RBF and a NF network. 32
3.5 Fuzzy system inference. 33
3.6 Illustration of a pantograph MIMO TAKAGI-SUGENO fuzzy system. . 34
3.7 Illustration of a fuzzy logic linguistic terms definition. 35
3.8 LOLIMOT based pantograph model I/O configuration. 39
3.9 LOLIMOT based pantograph model scheme. 40
3.10 Weighting and membership functions of a LLMN over partition space. 43
3.11 Membership functions of a LLMN over time. 44
3.12 Illustration of the applied partition space for pantograph LLMN. . . . 45
3.13 Pantograph LLMN (surrogate), scheme of the three-mass oscillator. . 47
3.14 Free body diagram of the three-mass oscillator. 48
3.15 Illustration of the linear least-squares problem. 55
3.16 Illustration of the partitioning of the SVD of a matrix. 56

3.17 Illustration of the dimension reduction in subspace identification. . . 63
3.18 LLMN output blending method illustration. 74
3.19 LLMN parameter blending method illustration. 74

4.1 Pantograph LLMN program code. 80
4.2 Operation point (OP) data set. 82
4.3 Operation point (OP) training data set pantograph LLMN (surrogate). 83
4.4 Whole range (WR) data set. 83
4.5 Whole range (WR) training data set pantograph LLMN (surrogate). . 84
4.6 LOLIMOT based pantograph model simulation results. 86
4.7 LOLIMOT based pantograph model NMSS systems information. . . . 87
4.8 Pantograph LLMN (surrogate) identified parameters for WR data set, illustration. 92
4.9 Pantograph LLMN (surrogate) simulation result for the OP data set, contact force F_p. 93
4.10 Pantograph LLMN (surrogate) simulation result for the OP data set. 94
4.11 Pantograph LLMN (surrogate) simulation results for WR data set. . . 96
4.12 Pantograph LLMN (surrogate) simulation results for WR data set, zoomed. 97
4.13 Pantograph LLMN (surrogate) simulation results for WR data set, contact force F_p. 98
4.14 Pantograph LLMN (surrogate, extended parameter vector) simulation results for WR data set, contact force F_p. 98
4.15 Pantograph LLMN (surrogate) simulation result for WR data set, contact force error and MSF. 99
4.16 Pantograph LLMN (surrogate) DFT of the simulation result for the pan-head velocity $\dot{\xi}$. 100
4.17 Pantograph LLMN (surrogate) DFT of the simulation result for the crossbar velocity $\dot{\zeta}$. 100
4.18 Pantograph LLMN (surrogate) DFT of the simulation result for the contact force F_p. 101
4.19 Pantograph LLMN (surrogate) pole positioning ($\mathcal{Z}$-plane). 102
4.20 Pantograph LLMN (n4sid) case study regarding the N4SID system order using OP data sets. 104
4.21 Pantograph LLMN (n4sid) simulation result for the OP data set, contact force F_p. 105
4.22 Pantograph LLMN (n4sid) simulation result for the OP data set. . . . 106
4.23 Pantograph LLMN (n4sid) simulation result for the WR data set. . . 108
4.24 Pantograph LLMN (n4sid) simulation result for the WR data set, zoomed. 109
4.25 Pantograph LLMN (n4sid) simulation result for the WR data set, contact force F_p zoomed. 110
4.26 Pantograph LLMN (n4sid) simulation result for the WR data set, contact force error and MSF. 110

4.27 Pantograph LLMN (n4sid) with parameter blending simulation result for the WR data set, contact force error and MSF. 111
4.28 Pantograph LLMN (n4sid) simulation result for the WR data set, parameter blending. 112
4.29 Pantograph LLMN (n4sid) DFT of the simulation result for the panhead velocity $\dot{\xi}$. 113
4.30 Pantograph LLMN (n4sid) DFT of the simulation result for the crossbar velocity $\dot{\zeta}$. 114
4.31 Pantograph LLMN (n4sid) DFT of the simulation result for the contact force F_p. 114
4.32 Pantograph LLMN (n4sid) pole positioning ($\mathcal{Z}$-plane). 115
4.33 Pole positioning ($\mathcal{Z}$-plane) of both pantograph LLMNs in comparison with the linearized WBM. 119
4.34 Linearized white-box pantograph model simulation result for the WR data set, contact force F_p zoomed. 120
4.35 Boxplot, comparison of the FIT of the contact force F_p for all developed pantograph models. 123

List of Tables

2.1 Modes of the pantograph white-box model. 17
2.2 Properties of the pantograph models, outlook. 23

3.1 Overview nonlinear modeling approaches. 27
3.2 Overview of neural network specific terms. 37
3.3 LLMN premise parameters for equidistant partitioning. 42
3.4 The four fundamental subspaces of a matrix. 54
3.5 Relation of the four fundamental subspaces and the partitioned SVD of a matrix. 57
3.6 Relation of the four fundamental subspaces and the QR factorization of a matrix. 57
3.7 Subspace identification methods, part 1. 68
3.8 Subspace identification methods, part 2. 69
3.9 Weighting matrices for subspace ID methods. 70

4.1 Pantograph LLMN configurations. 79
4.2 Pantograph LLMN (surrogate) parameter constraints. 89
4.3 Pantograph LLMN (surrogate) identified parameters for WR data set. 90
4.4 Collected results of computational efficiency of the developed pantograph models. 116
4.5 Collected FIT of developed pantograph models. 118
4.6 Comparison of FIT for the developed pantograph models contact force F_p. 121

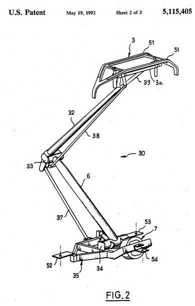
5.1 Pantograph LLMN (surrogate) free parameter allocation. 125
5.2 Pantograph LLMN (surrogate) FIM. 126
5.3 SWOT analysis. 131
5.4 Confrontation of the developed pantograph LLMNs. 132

Chapter 1

Introduction

The main task of a pantograph (mechanical framework) is to establish a continuous connection between the railway car of a electrically powered train and the overhead (or catenary) wire (compare e.g. [27]). While performing this task, the pantograph has to ensure that the resulting contact force stays within certain specified boundaries to limit attrition due to frictional contact on the one hand and to limit or minimize the inflicted stress and displacement of the overhead wire on the other hand. These challenges should be met with the utilization of modern day feedback control schemes and methods (model predictive control (MPC) and observer, see e.g. [12]), which require a real-time capable pantograph-model.

This thesis is structured as follows. The first chapter *Introduction* gives a brief introduction to the field of pantograph design and its encompassing industry. The motivation to approach this problem field is elucidated and the problem statement is formulated. The second chapter *Pantograph Modeling*, on one hand, introduces an existing white-box model of the nonlinear pantograph (which represents a simplified nonlinear model of the pantograph - reference model - and is utilized to generate reference data for the model-on-model identification of the developed models), and on the other hand lays out the specifications for the desired pantograph model, which is a nonlinear static model (realized as a local linear model network). The third chapter *Methodologies of Nonlinear Modeling and Identification* treats the applied methodologies in the chosen approaches to this nonlinear modeling problem. The superordinated methodology i.e. the chosen model structure is a neuro-fuzzy model implemented with local linear models in state-space form and therefore also referred to as local linear model network in state-space configuration. In the fourth chapter *Numeric Studies of Pantograph Models* a collection of simulation results of the identified pantograph models is presented, complemented with a preliminary discussion and analysis of the obtained findings. The fifth and final chapter *Observations and Discussion* is dedicated to an extended discussion of the developed pantograph models, also describing some observations that were made during the development phase. The main statements and conclusions are formulated there.



a b

Figure 1.1: a) State-of-the-art pantograph (Product of Siemens) and b) Scheme of a pantograph (U.S.-Patent [2])

1.1 Motivation

A state-of-the-art pantograph can be seen in Figure 1.1a (product of Siemens). Figure 1.1b shows a scheme of a pantograph framework (US Patent, [2]).

1.1.1 State-of-the-art Pantograph System

Some key facts about today's pantograph system are pointed out, describing the setting in which the problem statement will be formulated.

- Movement over a certain operating range ($0.5[m] - 2.5[m]$):
  - Different kind of dynamics are occurring. On the one hand the movement over the operating range, e.g. tunnel entrances involve a fast and large travel in position (height). One the other hand excitation is induced from the overhead line, depended on the traveling speed.
- Unstable mechanism (tipping between two equilibrium positions, both these positions lie outside the operating range):
  - Stabilized through an external force, i.e. coupling with overhead wire.
  - In the absence of the overhead wire, the pantograph framework travels into its uppermost position, after a critical torque M_{crit} is imposed via the pneumatic actuator.
  - Comparable with a pendulum which has a torsion spring mounted to its hinge joint. Initially in the lower equilibrium position (hanging down), the pendulum would travel to the upper equilibrium position after reaching a critical angle (inverse pendulum equilibrium position).
- The implemented feedback control scheme is realized via a P-controller
  - A pneumatic actuator (p_{max} = 10 [bar]) is the single control input (adjusted through traveling speed).

  - The pneumatic system of the train, which provides the drive for the actuator (torque on the pantograph framework) is fluctuating and inert, therefore precise actuator travel as well as high dynamic control are infeasible.

- Nonlinear dynamic behaviour of the referenced white-box model:
  - Consists of two main components: lower framework (nonlinear system) and pan-head (linear system) on top.
  - The nonlinearity is given by the geometry of the mechanism, all modeled springs and damping elements are linear, all bars are modeled rigid.

1.1.2 Limits in practical application of today's pantograph system

The limits in practical application of a state-of-the-art pantograph system are reached, if certain disturbances are acting on the system (compare e.g. [3], [4] or [27]). Then the optimal contact force cannot be maintained with the current feedback control scheme (simple P-controller). Such **disturbances** are:

- Variation of overhead wire position (mounting).
- Variations in the track bed and impacts from the rails.
- Tunnel entrance and exit.
- Wind/turbulence or **airstream** (especially crucial for **high speed trains**).
  - One crucial factor, that limits the maximum traveling speed of a passenger train today, is that the - with higher speed in intensity growing - airstream.
  - The airstream excites the overhead wire, which leads to oscillations and high deflections from the original position.
  - Therefore the primary function of the pantograph framework cannot be fulfilled with the current implemented feedback control scheme when reaching some critical velocity.
  - Solving this problem could be economically advantageous, making the railway more competitive in the market of passenger transportation (main rival in market: passenger plane).

1.1.3 Modern-Day Feedback Control

Implementing a modern-day feedback control scheme (e.g. model predictive control (MPC) with or without an observer) would resolve the problem of a limitation in traveling speed due to pantograph-sided control and furthermore would minimize wear of the pan-head and overhead wire by keeping the contact force inside a certain band around its optimal point (maintenance costs).

It follows a list of possible **problems in implementing** such a feedback control scheme:

- Electromagnetic induction (high currents trough overhead wire) hinders use of sensors and actuators.
- Inhospitable ambiance (outdoor use: temperature fluctuations, rain, ice, etc.).
- The state-of-the-art feedback control implementation requires very low maintenance.
- A failure safety solution has to be implemented (e.g. redundant systems) due to danger to life in case of malfunction or breakdown.
- Licensing/certification/approval is juristically challenging and time consuming and in the passenger transport market (several legislative levels: national/EU/US/...).

1.2 Problem Statement

The developed **pantograph model** is meant to be used as part of an **online simulation tool**, which shall be able to simulate the dynamics of the coupled pantograph-contact line system. A scheme thereof is depicted in Figure 1.2, which uses the following physical quantities: contact position η, collector head position ξ, contact force F_p and actuator torque M_{pa} (see Section 2.1). This Figure also illustrates how the pantograph modeling problem is located in this superordinated problem, from which the specifications for the pantograph model originate.

For the development of the pantograph model in this thesis, a fully nonlinear model (given by [1]) with six degrees of freedom (DOF) is used as **reference**. In this model, furthermore referred to as **white-box model** (WBM), all equations are derived from

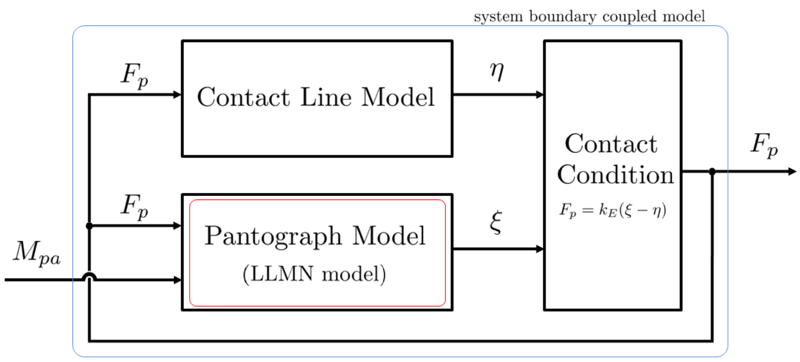


Figure 1.2: Localization of the pantograph model in the superordinated problem, which is the co-simulation of the coupled pantograph-catenary system. The contact force F_p represents the coupling condition.

first principles, i.e. this model formulates the explicit equations of motion for the two-dimensional nonlinear pantograph implementing certain simplifications (e.g. no friction). A more detailed description of this white-box model and its validity range can be found in Section 2.1.

> *Remark* 1.2.1 (**Disclaimer**). The pantograph models developed in this thesis are models identified from data generated by the white-box model (see Section 2.1 and [1]) and not measurements of a real-world pantograph. Therefore this thesis has to be understood as a **model-based study** on the model-on-model identified pantograph models, serving the purpose of comparing the - for these models developed - underlying methodologies. All examinations and results are valid only for the white-box model and are not necessarily applicable nor necessarily valid for real-world pantographs. The results have to be interpreted with a certain **caution** and need to be validated for each specific pantograph. The limitation, that friction is not considered in the white-box model is especially stressed here, because early measurements from the pantograph test-bench show hysteresis effects, indicating a decisive impact of friction on the pantograph behaviour. Nevertheless the findings of this thesis yield a comprehensive insight into the nonlinear pantograph dynamics originating form the pantograph geometry. Furthermore this thesis can be utilized as a guideline for the methodical and qualitative examination of data measured from a pantograph.

The **requirements** for the desired pantograph model are formulated in the following section.

1.2.1 Pantograph Model Specifications

The pantograph model was developed considering the following **specifications**:

- **Computational efficiency**: Real-time capable model.
- **Structure**: Nonlinear static model realized as a ***local linear model network (LLMN)*** (see Sect. 3.2, chosen as a local linear TAKAGI-SUGENO neuro-fuzzy type of network) that represents the nonlinear dynamics of the pantograph.
- **Mathematical formulation**: The ***local linear models (LLMs)*** shall contain the identified linear systems in ***state-space form*** (to be able to exploit all advantages of this form, e.g. later incorporation into an adaptive and predictive control system using model predictive control).

$$LLM_i = [\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i, \mathbf{D}_i] \tag{1.1}$$

$\mathbf{A}_i$ ... $\mathbb{R}^{n\times n}$, $\mathbf{B}_i$ ... $\mathbb{R}^{n\times m}$, $\mathbf{C}_i$ ... $\mathbb{R}^{q\times n}$, $\mathbf{D}_i$ ... $\mathbb{R}^{q\times m}$ (no direct feed-through)

with n ... number of states, m ... number of inputs and q ... number of outputs, determining the desired model as a local linear model in state-space configuration.

- **Performance**: The error compared to the fully nonlinear model (reference model) shall be held small, especially ***low frequencies*** (up to $15Hz$, which corresponds to the expected control range of the pantograph system) shall be mapped good (match in phase and amplitude of all signals is desired).

Remark 1.2.2. The requirement for the pantograph to be modeled as a **LLMN in state-space configuration** is essential to this thesis and necessary if one wants to exploit the vast methodology of control techniques available for this type of system description, which is the (in this case blended) state-space system. One key target of this thesis was to examine if there is a suitable way of developing such a model (LMN in state-space configuration) for a nonlinear problem, exemplary the pantograph.

A literature research was carried out for techniques that can be utilized to tackle this problem statement [18, 19, 33].

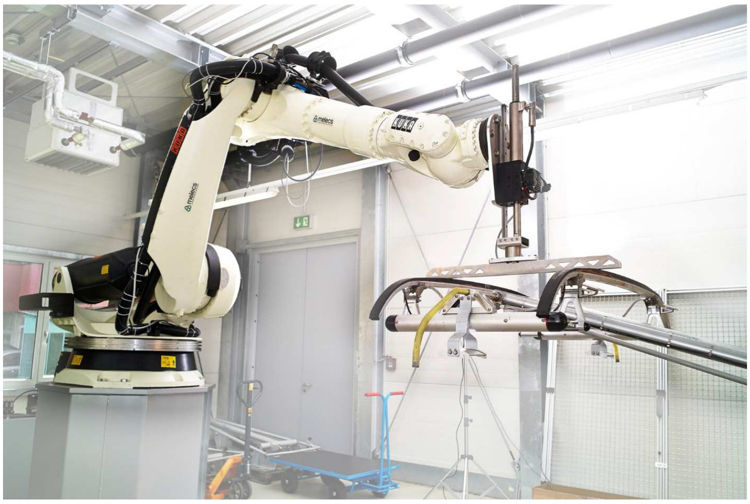


Figure 1.3: Photograph of the pantograph test bed at the Vienna University of Technology, in cooperation with Siemens.

1.2.2 Literature Review

This thesis, as described in the in this opening chapter, focuses on incorporating the nonlinear dynamics of the presented white-box pantograph model into a real-time capable local linear model network specified in Section 1.2.1. This thesis is part of a pantograph test bench project in cooperation with Siemens (see photograph in Figure 1.3) and therefore can also be seen as an evolution of the white-box pantograph model derived in [1]. Further publications on this project will follow in the near future, the pantograph test bed was introduced in [29].

In the associated literature (majority of publications) the main focus lies on solving the superordinated problem, which is given as the co-simulation of the pantograph-catenary system (also pantograph-overhead line system, compare with Figure 1.2). In those publications the nonlinear dynamic of the pantograph is not of major interest. Hence the subsystem pantograph is described as a **global linear model** and all simulations that are carried out are limited around a certain operation point, in some cases solely to test the principal functionality of the developed concepts regarding the co-simulation. These global linear models of the nonlinear pantograph are most commonly described as **two-mass oscillator surrogate models** (see e.g. [17], [3], [40], [14], [4]). In [40] an attempt is made to apply non-linear fuzzy controllers to the pantograph-catenary control problem. In [14] a comparison of different modeling approaches using pantograph surrogate models (two- to four-mass oscillators) in connection with the pantograph-catenary problem is presented.

In other approaches to the same problem the pantograph is described as a **multibody model** while the overhead line is described as a finite element model. A detailed description of these subsystem models and their coupling in simulation (co-simulation) can be found in [27].

Chapter 2

Pantograph Modeling

A considerable part of this thesis is devoted to solving a nonlinear modeling problem (with the specifications given in Section 1.2.1). As it can be read in literature (e.g. [19, 18]) a certain state of mind is necessary to succeed in this field (see Chapter 3). To set the mood for this chapter, the ***principle of incompatibility*** as stated informally in [44] is quoted here:

> As the complexity of a system increases, our ability to make precise and yet significant statements about its behaviour diminishes until a threshold is reached beyond which precision and significance (or relevance) become almost mutually exclusive characteristics. [44, page 28]

One aim of this thesis was to develop a grey-box model (see Sect. 2.2), where the existing **expert knowledge** about the real-world pantograph was incorporated into the design of the pantograph model. As a consequence of this modeling goal an **iterative design process** was put in train, where expert insight, knowledge about the purpose of the model, engineers heuristics and information in the observed data of the white-box model were mixed. An illustration of this engineering cycle can be found e.g. in [18, page 30, Figure 1.18]. The following sections treat the main findings of this design process and the resulting conclusions for the desired pantograph models.

2.1 Existing Pantograph White-Box Model

This section gives a brief introduction to an existing white-box model (WBM) of the pantograph, which was used as a reference model for this thesis. For further information on the white-box pantograph model see [1].

Terms of the white-box model used in this thesis (in compliance with [1]):

A sketch of the white-box pantograph model can be seen in Figure 2.1. The individual components of the pantograph and the associated terms that will be used throughout the thesis are as following (compare [1] Chapter 2 and Figure 1.1a):

- The *overhead line* (generic term used by the International Union of Railways) consists of the contact wire and a supporting catenary wire with droppers which connect the two wires, constructed to achieve good high speed current collection (see e.g. [42] Figure 1) - throughout this thesis often referred to as *overhead wire* or *contact line*.

- η ... contact position (carbon contact position as well as overhead wire position).
- F_p ... contact force, intrinsic force acting between pan-head and contact wire.
- The *collector-head* consists of the pan-head and a symmetrical suspension composed of sophisticated torsion springs.
- The *carbon contacts* are establishing the contact to the overhead wire.
- ξ ... pan-head position.
- F_H ... crossbar force, intrinsic force acting between the lower framework (crossbar) and the collector head.
- The *lower framework* consists of several rods with the one on the top of the structure referred to as *crossbar*.
- ζ ... crossbar position.
- The main other rods are named *upper arm* (bended by α), *lower arm* and *thrust rod*.
- The lower arm an the thrust rod are connected via joints to a rigid frame.
- φ_1 ... angle between the railcar and the lower arm.
- The pneumatic actuator mounted on that frame applies a certain torque to the lower arm.
- M_{pa} ... torque acting on lower arm.
- Finally the frame is mounted onto the roof of the railcar via several outdoor post insulators (see Figure 1.1a).

Structure of the white-box model (compare [1, Chapter 2]):

- The white-box model describes the movement of the pantograph in a plane (two-dimensional movement).
- 6 degrees of freedom system (arbitrarily chosen), portrayed as a state vector by equation (2.1) (see Figure 2.2).

$$\mathbf{x}_{WBM} = [\varphi_1, \dot{\varphi_1}, \delta, \dot{\delta}, \xi, \dot{\xi}]. \tag{2.1}$$

- The underlying equations are the equations of motion derived from first principles.

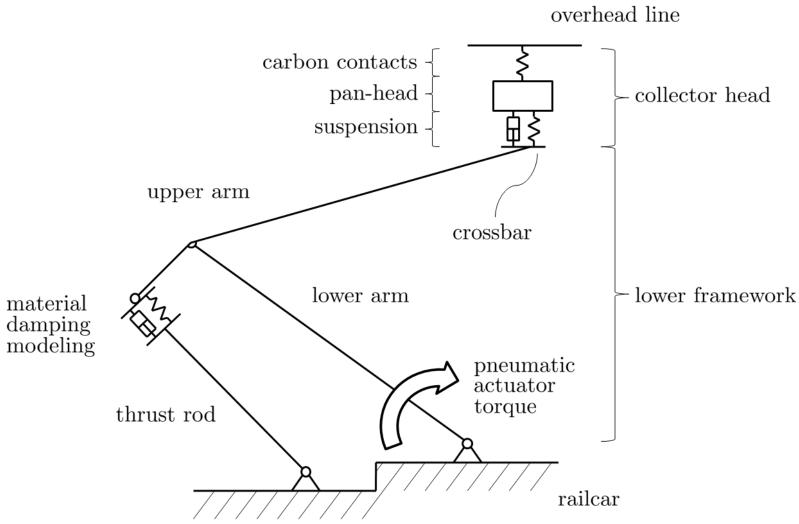


Figure 2.1: Sketch of the white-box pantograph model emphasizing the individual components and associated terms (compare [1, Figure 2.8].

- The **nonlinearity** of the white-box model is given by its **geometry**. Furthermore, regarding nonlinear effects it has to be mentioned, that
  - the material damping incorporated into the model by introducing two DOF $\delta, \dot{\delta}$ and keeping all bars rigid,
  - imperfect joints are not considered,
  - and the **friction** inside the system is **not** considered (eventual hysteresis, detected in pantograph test bench measurement data).
- The collector head is modeled as an one-mass-oscillator (compare [1, Section 2.2]), therefore it can be treated as a linear subsystem of the pantograph framework with the identified parameters ... m_C, k_C and c_C.
- The carbon contacts of the pan-head, which establish the physical contact to the overhead wire, are modeled as a spring with high stiffness (compare [1, Section 2.5]) ... k_E.

Performance of the white-box model:

- **High computational effort** (real-time factor (RTF) value around magnitude 10^2, where $RTF < 1$ implies real-time capability and $RTF > 1$ no real-time capability, see Section 4.6).
- Therefore **not applicable for real-time** feedback control scheme.
- Limitations in accuracy compared to the real-world pantograph due to modeling simplifications.

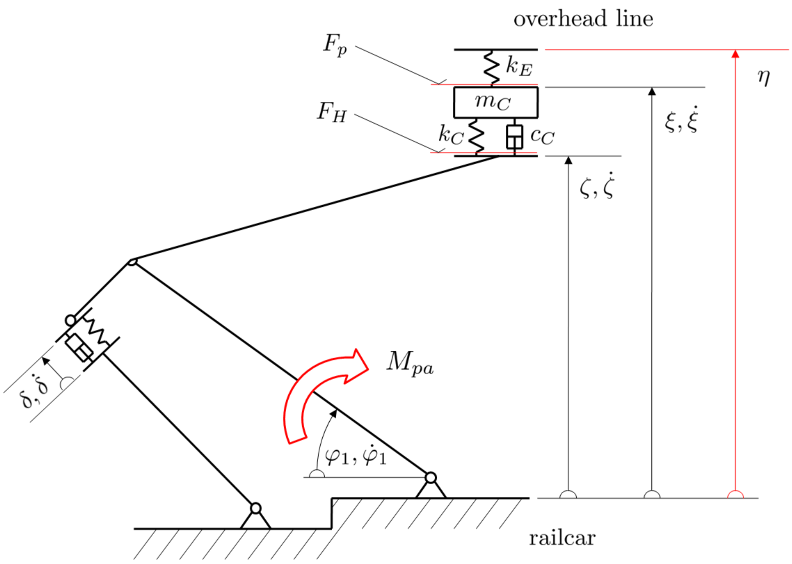


Figure 2.2: Sketch of the white-box pantograph model with the degrees of freedom, parameters of the collector head and internal forces. The two inputs (contact position η and torque M_{pa}) are highlighted (compare [1, Figure 2.8]).

- The parameters (e.g. material damping) of the white-box model where optimized for a certain operating point, therefore additional limitations in accuracy are given if the white-box model is evaluated in other operating points to generate reference data.
- The training data-sets for these optimizations where taken from the test-bench, i.e. are experimental data, therefore these signals are subject to all kind of typical measurement errors (systematic error, random error, sampling error, etc., comp.[1]).

Application/Implementation of the white-box model in this thesis:

- The model was used to generate data-sets for training and validation of the LLMN model.
- Therefore the white-box model represents the reference model to verify the performance of the pantograph LLMNs.
- **Awareness** has to be given to the fact, that the reference signals are already signals of a simplified model (see above) and not experimental data from a test-bench.

Summarizing the above matters, the white-box model from [1] is simulated with the inputs **contact position** η, which acts at the top of the framework, and the **pneumatic actuator torque** M_{pa}, which acts at the bottom, inducing the system. These input signals were modeled as excitation signals due to expert knowledge

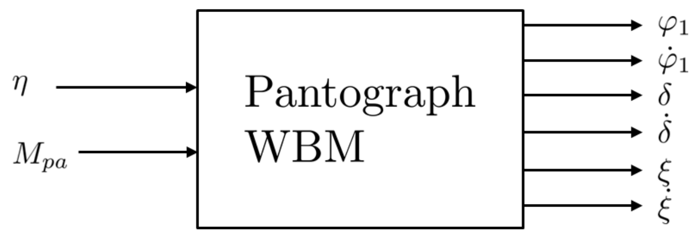


Figure 2.3: Block diagram of the pantograph white-box model illustrating its inputs and outputs. All other signals are derived from these quantities.

(see Section 4.2), therefore generating the reference data for the development of the pantograph LLMNs. The resulting outputs of the white-box model are the six degrees of freedom as portrayed in equation (2.1), which were, together with the inputs, utilized to compute all necessary signals (e.g. crossbar position ζ, see Section 2.2.1) according to the kinematics of the white-box model derived in [1, Section 2.1] (compare with Figure 2.3). The reference data is generated using analytical relations and therefore are reproduceable.

2.1.1 Static Examinations of the Nonlinear Pantograph

Figure 2.4 shows the dependency of the contact force F_p (force between pantograph and overhead wire) of η (position of the overhead wire) and M_{pa} (momentum of the pneumatic actuator). This nonlinear mapping of static quantities of the pantograph was obtained by evaluating the white-box model in several operating points, varying the two inputs contact position η and torque of the pneumatic actuator M_{pa}.

Discussion of Figure 2.4, nonlinear mapping of static quantities of the pantograph white-box model

- There is no significant mapping present between the contact position η and the contact force F_p. The contact force is nearly independent of the position of the contact wire in relation to the roof of the railcar.
- There is a proportional mapping present between the applied torque M_{pa} of the pneumatic actuator and the contact force F_p. The contact force depends on the torque in a nearly linear fashion.
- Additionally it has to be stated that during the examinations, there was no indication that the magnitude of the contact force influences the dynamics of the pantograph system.

2.1.2 Global Linear Model

At this point it would be legitimate to linearize the white-box pantograph model around an operating point (e.g. in the center of the operating range or expected

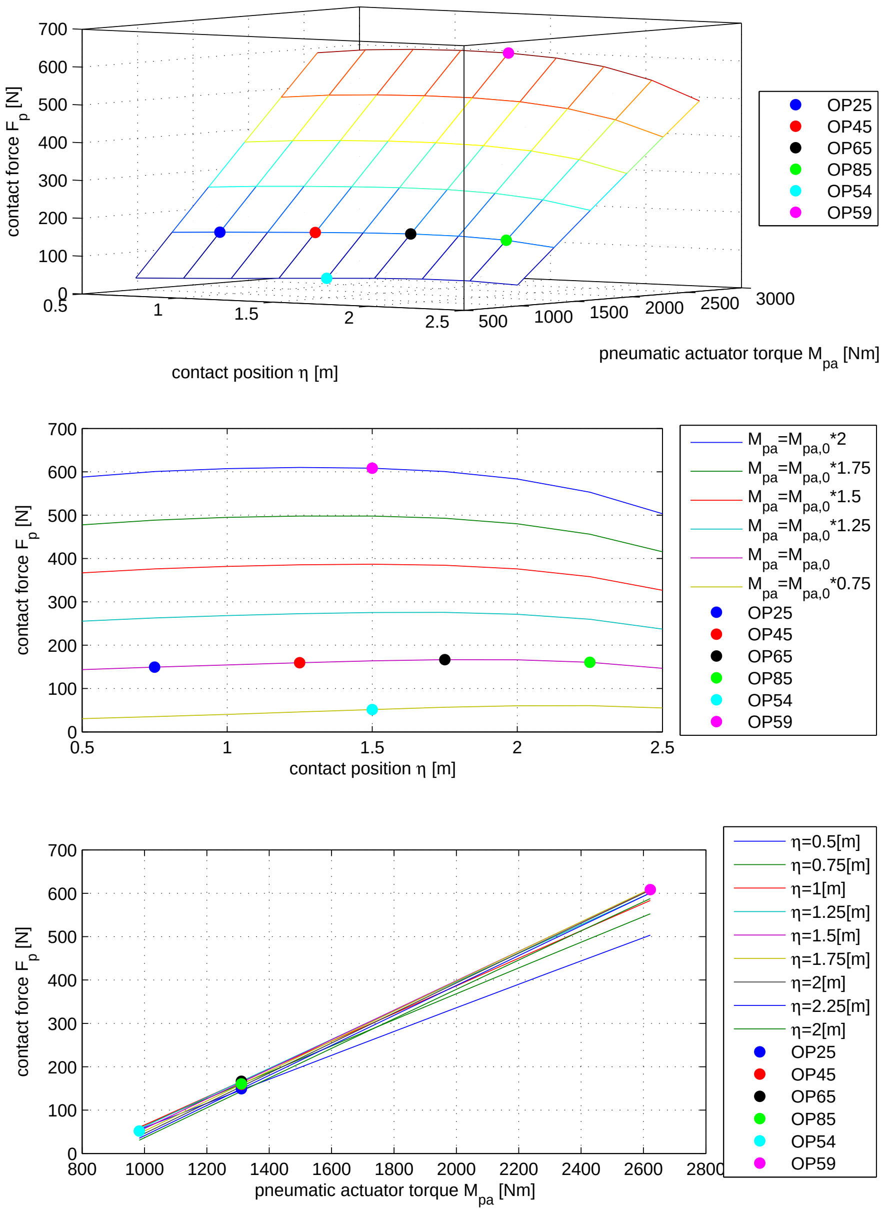


Figure 2.4: These plots show the nonlinear mapping of static quantities of the pantograph obtained by evaluation of the white-box model in several operating points, with the output (contact force F_p) depending on the inputs contact position and torque of the pneumatic actuator as $F_p = f(\eta, M_{pa})$. The value of $M_{pa,0}$ is given as 1310.9 [Nm], see Section 4.2.1.

predominant point of operation) and use this global linear model in the coupled pantograph-overhead line model (compare Figure 1.2).
For the global linear model, with regard to the definition of the model specifications in Sect. 1.2, the following statements can be formulated:

- **Computational efficiency**: Real-time capability of this global linear model is given (see Chap. 4).
- **Structure**: The linearized WBM would represent a global linear model, which could be interpreted as a LLMN model with just one single LLM.
- **Performance**: The achievable performance of this simple model is quite good and acceptable for some applications (obviously the FIT drops the further the pantograph moves away from the selected point of linearization in the operating range).
- **Mathematical formulation**: The state-space matrices of this linearized model can be derived directly from the white-box pantograph model equations (given by [1, equation (2.28)]).

According to the evaluation of the specifications given in Section 1.2.1 the **linearized white-box pantograph model** implemented as a **global linear model** is set as the **default** or **fail save model**.
This thesis however focuses on the modeling goal to achieve an increased performance in the whole operating range by incorporating the nonlinear effects of the pantograph in a LLMN structured model. Arguments that justify the additional effort for this approach can be found in Section 2.1.3.

2.1.3 Dynamic Examinations of the Nonlinear Pantograph

This section aims to investigate the dynamic system behaviour of the pantograph by carving out effects of the system's nonlinearities (engineer's perspective on dynamic system behaviour), which are, as mentioned above, given by the **geometry** of the pantograph. Several examinations were performed utilizing the white-box model.

Frequency Analysis (DFT)

At the beginning of this section the results of a **frequency analysis** of several data sets that were generated by the white-box model will be shown and discussed. The frequency analysis of the resulting signals (outputs of the white-box models) was carried out by a *Discrete Fourier Transformation (DFT)* of the respective signals. The following statements are relevant for the DFT:

- The sampling frequency is given by the selected sampling time as $F_s = 1/T_s = 1000$ [Hz].

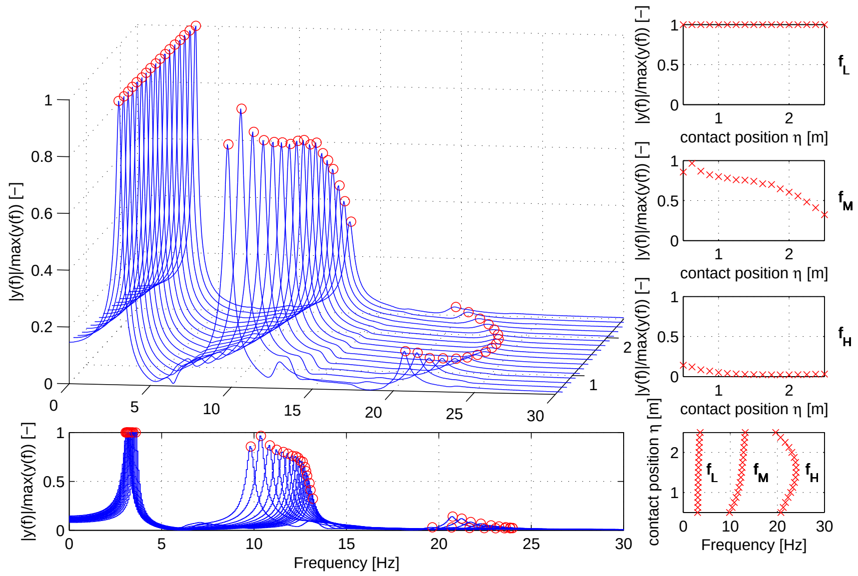


Figure 2.5: Discrete Fourier Transformation (DFT): **pan-head velocity** $\dot{\xi}$ step response of the **white-box model** for the whole operating range. Three modes (resonant frequencies f_L, f_M and f_H are detectable. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

- Therefore the resulting NYQUIST frequency is given as $F_{Ny} = F_s/2 = 500$ [Hz].
- The control range in the frequency domain (controller dynamic) is expected to lie between 0 [Hz] and slightly above 10 [Hz], this range is of primary interest.

From these statements one can see, that the relevant frequencies lie sufficiently far from the NYQUIST frequency. The plots show the frequency shares of the DFT (coefficients of the Fourier transform) over the sample frequency range 0 [Hz] to 30 [Hz] in steps given by the sampling frequency. Additionally these results are portrayed for the whole operating range of the pantograph (0.5 [m] to 2.5 [m]). The analyzed signals are step responses (equidistant steps of the contact position η) of the WBM for different operating points. The position signals (collector head position ξ and pan-head position ζ) show no significant peaks in the frequency domain, which would indicate resonant frequencies, therefore only the DFT of the velocity signals ($\dot{\xi}$ in Figure 2.5 and $\dot{\zeta}$ in Figure 2.6) and the contact force (F_p in Figure 2.7) are depicted.

The resonant frequencies of all examined signals are collected in Table 2.1.

<u>Conclusion</u>:
As it can be seen from the Figures 2.5, 2.6, 2.7 and Table 2.1 the resonant frequencies are not constant over the operating range. This observation indicates, that the dynamic system behaviour is changing correspondingly over the operating range and a

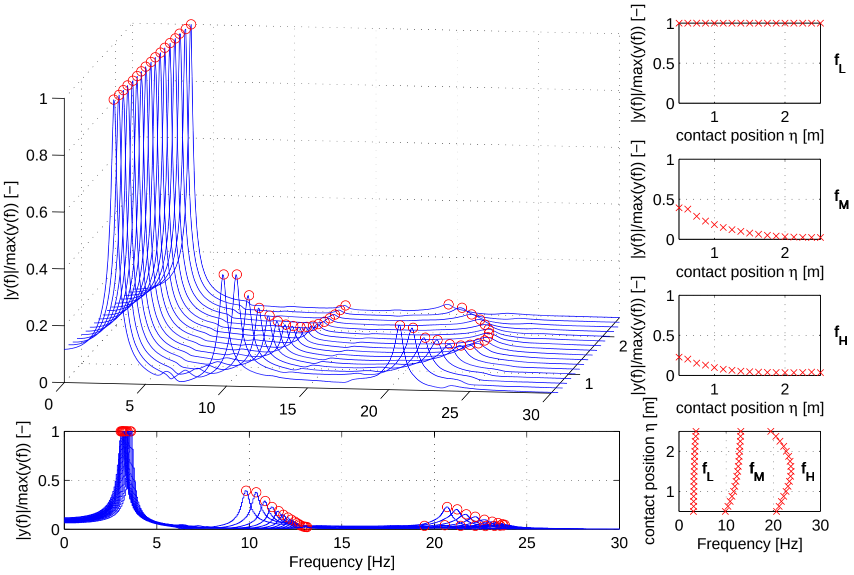


Figure 2.6: Discrete Fourier Transformation (DFT): **crossbar velocity** $\dot{\zeta}$ step response of the white-box model for the whole operating range. Three modes (resonant frequencies f_L, f_M and f_H are detectable. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

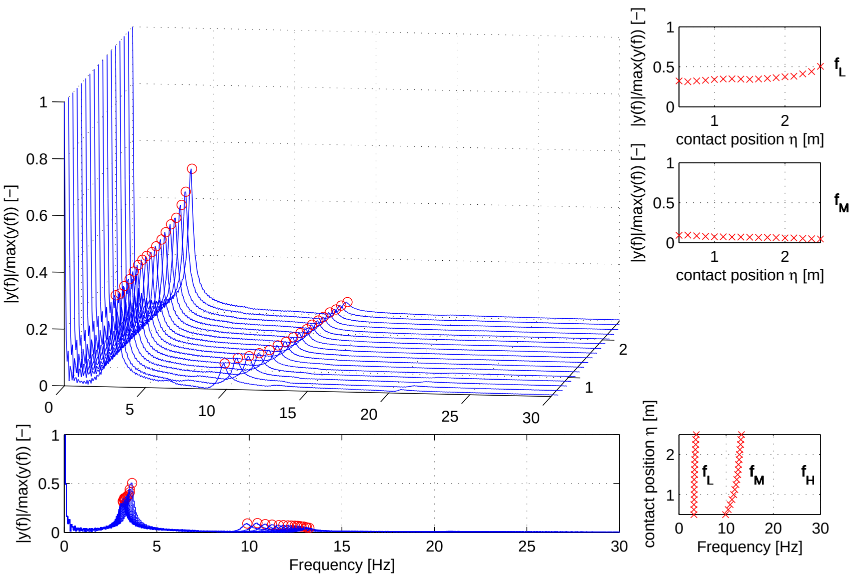


Figure 2.7: Discrete Fourier Transformation (DFT): **contact force** F_p (equ. (3.14)) step response of the white-box model for the whole operating range. Two modes (resonant frequencies f_L, f_M and f_H are detectable. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

Frequencies		Positions		Velocities		Forces	
		ξ	ζ	$\dot{\xi}$	$\dot{\zeta}$	F_H	F_p
f_L [Hz]	min	-	-	3.052	3.052	3.174	3.174
	max	-	-	3.601	3.601	3.662	3.662
	mean	-	-	3.256	3.264	3.325	3.332
	median	-	-	3.235	3.235	3.235	3.296
	mode	-	-	3.113	3.174	3.235	3.235
f_M [Hz]	min	-	-	9.827	9.827	9.827	9.888
	max	-	-	13.184	13.123	13.184	13.245
	mean	-	-	12.078	12.092	12.078	12.107
	median	-	-	12.390	12.451	12.390	12.451
	mode	-	-	-	-	-	-
f_H [Hz]	min	-	-	20.752	20.691	19.592	-
	max	-	-	23.987	23.804	23.804	-
	mean	-	-	22.369	22.332	22.429	-
	median	-	-	22.827	22.705	22.705	-
	mode	-	-	23.865	22.217	23.804	-

Table 2.1: Collected values and statistical averages (mean, median and mode) of the DFT analysis of several signals of the white-box model of the pantograph.

global linearized model will not satisfactory describe the dynamics of the pantograph.

Dynamic Behaviour of the Linearized WBM

Additionally the behaviour of the linearized white-box model over the operating range was examined. Therefore the WBM was linearized around several operating points according to equation [1, Section 2.4, equations (2.28)]. The continuous-time state-space model [1, equations (2.29) and (2.30)] was then discretized by using the MATLAB function `c2d()` with zero-order hold.

By examining the entries of the discrete-time system matrix **A** and input matrix **B**, it can be stated that the entries change their values in a continuous fashion according to the change of the operating point, around which the model was linearized. Some in a more linear fashion, some in a more nonlinear fashion and with strongly differing magnitudes. This continuous change of the linearised white-box model across the operating range can also be detected in the pole-zero map of the eigenvalues of the system matrix **A**, as it is depicted in Figure 2.8.

<u>Conclusion</u>:
Referring to the conclusion in the previous Section 2.1.3 it can be observed, that the entries of the A and B matrix, obtained by linearizations of the white-box model (see Section 2.1.2), vary for different operating points. As a result, also the pole positions, due to the eigenvalues of the system matrix **A**, move inside the unit circle (see Figure 2.8) following certain trajectories. Form these findings it can be stated, that the pantograph inherits a certain nonlinearity and therefore a nonlinear modeling problem is present. The well-behaving linearized state-space matrices would allow implementing a control-scheme based on a **gain scheduling approach** (see e.g. [12,

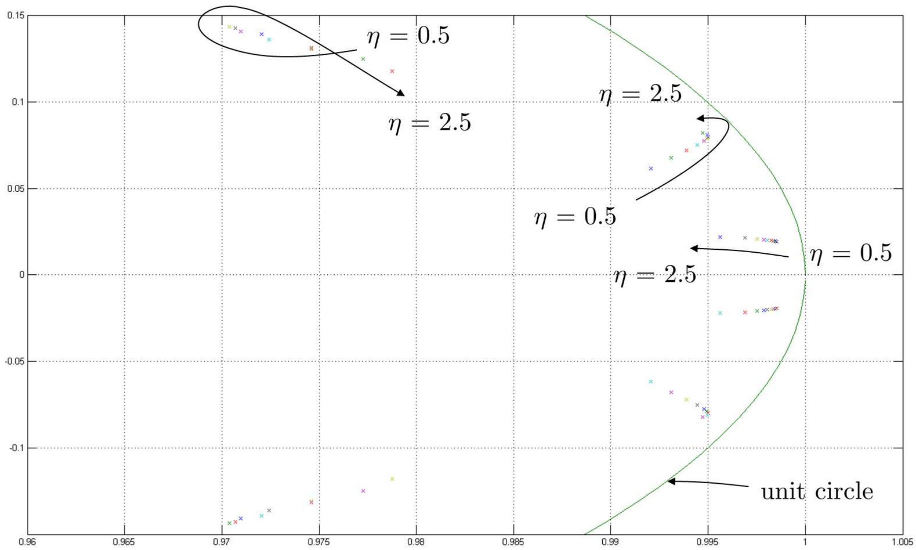


Figure 2.8: Pole-positioning in the $\mathcal{Z}$-domain, given by the eigenvalues of the system matrix $\mathbf{A}$ of the discrete-time state-space system of the linearized white-box model for several operating points across the operating range. The arrows indicate the movement of the poles with increasing operating height.

page 7, Section 2.1]).

Additional Findings

Influence of the pneumatic actuator as the solely controllable input of the state of the art pantograph:

Although the triple shown in Figure 2.4 (η, F_p and M_{pa}) are related in a nonlinear fashion, there was no indication during the examinations, that the contact force or the torque of the pneumatic actuator have an effect on the dynamic behaviour of the pantograph. Therefore the remaining examinations are all realized with M_{pa} set to an operating point at 1310.9 [Nm], which represents a realistic value identified on the test bench in [1]. This constant signal is then superimposed with an excitation signal by utilizing an *amplitude modulated pseudo-random binary sequence* (APRBS, compare e.g. [19, Section 17.7]) resulting in about 6% deviation ($\sim$ 75 [Nm]) form the operating point and employing hold times from 50 [ms] to 200 [ms] (see Section 4.2).

Inference from modes or resonant frequencies f_L, f_M, f_H to current position of the pantograph:

In principle the potential can be recognized, that by looking e.g. at the DFT analysis plots of the velocity signals of the WBM in Figure 2.5, that it would be possible

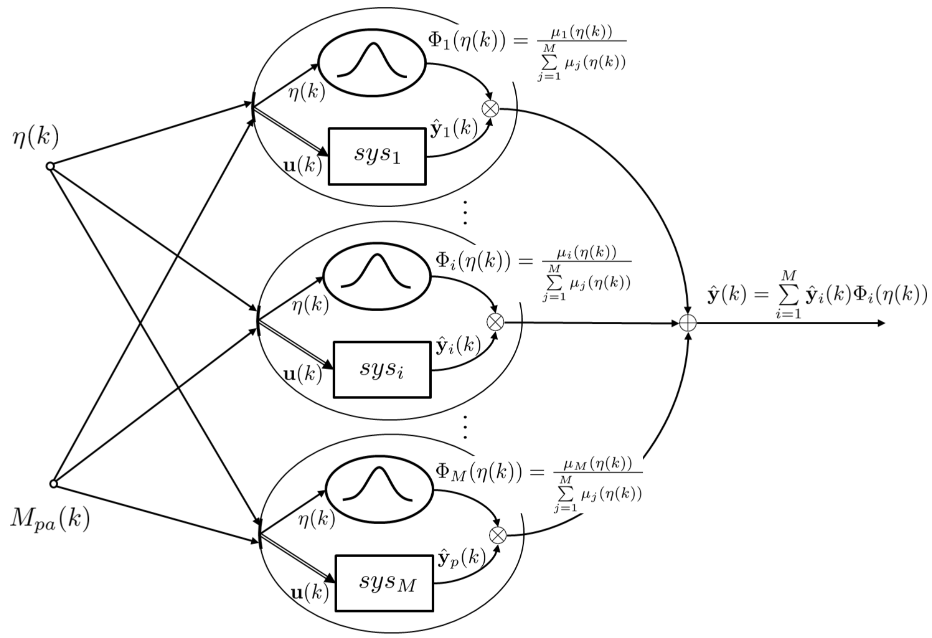


Figure 2.9: Pantograph local linear model network (LLMN) in state-space configuration utilizing output blending with chosen partition variable contact position η.

to extract information about the current operating height of the pantograph from knowledge of the position of the modes at certain operating points of a specific pantograph (with a specific geometry). A scheme could be implemented based on the distinction of cases, by examining f_H (gives two possibilities of the current height: lower or higher) and compare it with the value of f_M (leaves just one possible position) to identify the current operating position (see e.g. the depiction of the peaks of the resonant frequencies over the operating range in the right bottom corner of Figure 2.6).

2.2 Developed Pantograph Model

In this section the **settings for the developed pantograph model** are formulated. The aim was to find a model satisfying the requirements given in Section 1.2 by designing a local linear model network (LLMN) utilizing a local linear Takagi-Sugeno neuro-fuzzy network in state-space configuration. The developed model will be referred to as pantograph LLMN. An illustration of a LLMN in state-space configuration utilizing the output blending method and the chosen inputs (contact position η and pneumatic actuator torque M_{pa}) and partitioning variable (contact position η) can be seen in Figure 2.9, where the sys_i represents the state-space model inside the i-th LLM.

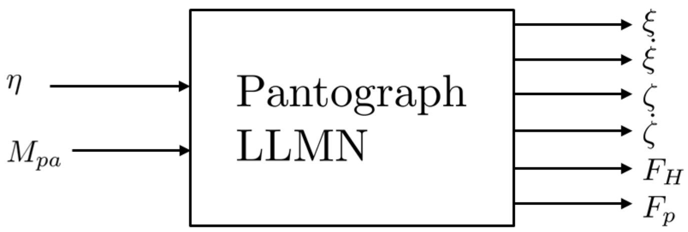


Figure 2.10: Block diagram of the pantograph LLMN model illustrating the set input and output configuration. The inputs are the same as for the WBM, while the outputs are chosen differently (compare with Figure 2.3).

2.2.1 Global Inputs and Outputs of the Developed Pantograph Model

Figure 2.10 shows the pantograph LLMN as a block diagram, with the inputs/outputs set as the following quantities of the pantograph (see Section 2.1 for description of the quantities):

- Pantograph LLMN inputs: contact position η, pneumatic actuator torque M_{pa}

$$\mathbf{u}_{LLMN} = \begin{bmatrix} \eta & M_{pa} \end{bmatrix}^T. \tag{2.2}$$

- Pantograph LLMN outputs: pan-head position ξ and velocity $\dot{\xi}$, crossbar position ζ and velocity $\dot{\zeta}$, crossbar force F_H and contact force F_p

$$\mathbf{y}_{LLMN} = \begin{bmatrix} \xi & \dot{\xi} & \zeta & \dot{\zeta} & F_H & F_p \end{bmatrix}^T. \tag{2.3}$$

A short discussion of the choice of inputs is required here, which are acting at the top - position η or force F_p - and the bottom - pneumatic actuator torque M_{ap} - of the pantograph. The following two choices would be admissible:

- η and M_{pa} (chosen, same input configuration as in the white-box model)
- F_p and M_{pa}

under the following considerations:

- **Partitioning**: The partition space is unambiguous if the contact position η is chosen as partition variable, allowing to determine all nonlinear parameters of the validity function by expert knowledge. The contact force F_p however is unemployable as partition variable without further pre-processing of the force signal (e.g. frequency analysis, scheduling, etc.). Somehow the scheduling variable has to qualify the current operating height of the pantograph to make the developed model perform satisfactory, therefore η was chosen as an input signal and partition variable.

- **Real-World Problem**: Neither η nor F_p are measurable in the current (state of the art) implementation of the pantograph. However the measurement (e.g. optical measurement) of the angle of the lower arm ϕ_1 could be a possibility for future designs, therefore a mapping of one of the positions (ζ, ξ or η) could be realizable.

- **Causality**: Considering the overall model of the coupled pantograph-contact line model (superordinated problem, see Figure 1.2), F_p as an input would be the desired choice because of causality reasons (η is adjusting to F_p).

2.2.2 Local Linear Model Network (LLMN) Setup

What follows are the choices made for the **setup of the LLMN**: (for a detailed explanation of all terms see Chapter 3 and compare [18, 19])

- A **one-dimensional partition space** is used.

- The **contact position** $\eta(k)$ is set as the partition variable (see Section 2.2.1).

- Therefore the height of the contact position (operating range of the pantograph) is set as the **partition space**.

- Several local linear models (LLMs) are equidistantly positioned over the operating range (*centers* of the according MSFs).

- Equal parameters (*spread* and *proportionality factor* k_σ) for the membership functions (MSFs) of the according LLMs are used.

- Blending or interpolation of the LLMs is realized via a membership-function value $\mathbf{\Phi}(\eta(k))$ given by the current positioning of the partition variable (contact position η) in the partition space, with one of these concepts:
  - **Blend the outputs** of the LLMs to generate the global model output (output blending method, default solution).
  - **Blend the system matrices** of the LLMs to generate the global model output (parameter blending method, preferred solution).

Furthermore, as a **simplification**, the pantograph LLMN is set to perform only **one-dimensional movements**. The deviation of the crossbar position ζ in the driving direction of the train over the operating range is depicted in [1, Figure 2.4]. As it can be seen there, the divergence of the crossbar in driving direction is of about ± 0.1 [m] over the whole operating range of 2 [m] and therefore gets neglected to further simplify the model. Hence, as illustrated in Figure 2.11, the pantograph LLMN describes the movement of the pantograph on a line, while the WBM describes it on a plane.

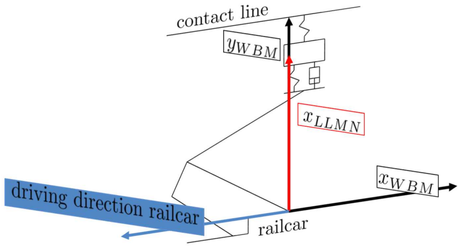


Figure 2.11: Comparison of the pantograph white-box model (two-dimensional movement, see [1, Figure 2.4]) and the pantograph LLMN (one-dimensional movement) regarding their dimension of movement.

2.2.3 Local Linear Model (LLM) Setup

As required in in the specifications for the pantograph models (see Section 1.2), the local linear models (LLMs) shall contain identified linear systems in **state-space formulation**. In this thesis two approaches satisfying this requirement were realized.

- A parametrized mechanical surrogate model (three-mass oscillator, see Section 3.4).
- A subspace identification method named N4SID combined with model reduction method (see Section 3.5).

The main aspects of these two approaches are summarized in Table 2.2.

Remark 2.2.1 (Table 2.2). The physical quantities crossbar force F_H [N] and contact force F_p [N] are post-computed in each time-step k from the global (already blended) output signals of the LLMs according to equations (3.15) and (3.14) (compare LLMN output in equation (2.3)).

Remark 2.2.2 (Table 2.2). The pantograph LLMN (n4sid) was developed as an alternative to the pantograph LLMN (surrogate), which is a computationally similar fast (real-time capable) model. The pantographs producing industry provides frameworks with different geometry, therefore a certain flexibility in the models was sought. The development of a white-box model is time consuming and therefore expensive, but necessary if a linearized global model is to be obtained (linearized from the first principles around an operating point). Furthermore the white-box model will still contain simplifications and therefore will not map the movement of the real-world pantograph accurately. The identification of the surrogate model (three-mass oscillator) is again time consuming and additionally requires good initial values for the parameters (danger of local optima). The subspace identification methods however work very efficiently and user friendly, identifying the model directly from a set of input and output data. Disadvantageously the N4SID approach requires an additional pre- and postprocessing of the inputs and outputs, where static values have to be determined to set up a look-up table.

Properties	**Pantograph LLMN (surrogate)**	**Pantograph LLMN (n4sid)**
Physical interpretability	given for the states and parameters of the state-space system	not given, modes given if state-space system is transformed to modal form
Uniqueness of the state-vector	given	not given
Training strategy	underlying parametrized surrogate model identified by OE method (cost fuction)	subspace identification of the state-space matrices directly from input/output data
Operating height adjustment (offset correction)	state-space system extended with affine terms (see equations (3.33)-(3.34))	preprocessing using a low-pass filter (creating zero-mean values) and postprocessing utilizing a look-up table (static values)
LLM input-vector	$\mathbf{u}_{LLM} = \begin{bmatrix} \eta & M_{pa} \end{bmatrix}^T$	
LLM state-vector	$\mathbf{x}_{surrogate} = \begin{bmatrix} \xi & \dot{\xi} & \zeta & \dot{\zeta} & \delta_M & \dot{\delta_M} \end{bmatrix}^T$	$\mathbf{x}_{n4sid} = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{bmatrix}^T$
LLM output-vector	$\mathbf{y}_{LLM} = \begin{bmatrix} \xi & \dot{\xi} & \zeta & \dot{\zeta} \end{bmatrix}^T$	

Table 2.2: Outlook on the properties of the two developed pantograph models based on local linear models (LLMNs).

Chapter 3

Methodologies of Nonlinear Modeling and Identification

This chapter is dedicated to discuss the in this thesis applied methods on the one hand and on the other hand discusses the structure of the developed pantograph models in detail. In Figure 3.1 an overview of the utilized methods is depicted.

3.1 Introduction

The main content of this thesis is the application of the *local linear neuro-fuzzy network* methodology to a nonlinear modeling problem (i.e. the pantograph, for distinctive features see Section 2.1), often simply referred to as local linear model network (LLMN). This modeling approach (compare [18], [19] and [39]) consists of the attended problem fields:

- **Decomposition Method**: Decomposition of the global nonlinear problem into linear subproblems (local linear models).
- **Structure**: Determine the structure of the local linear neuro-fuzzy network by specifying the **validity functions** (hidden layer or rule premise) and the parametrized state-space systems of the **local linear models** (output layer or rule consequent).
- **Identification**: Determination and/or optimization of the (validity function's) nonlinear and (the local linear models') linear parameters of the model from expert knowledge and/or data.
- **Blending Method**: Method of how to generate a global output of the local linear neuro-fuzzy network, by blending of the outputs or the parameters of the local linear models.

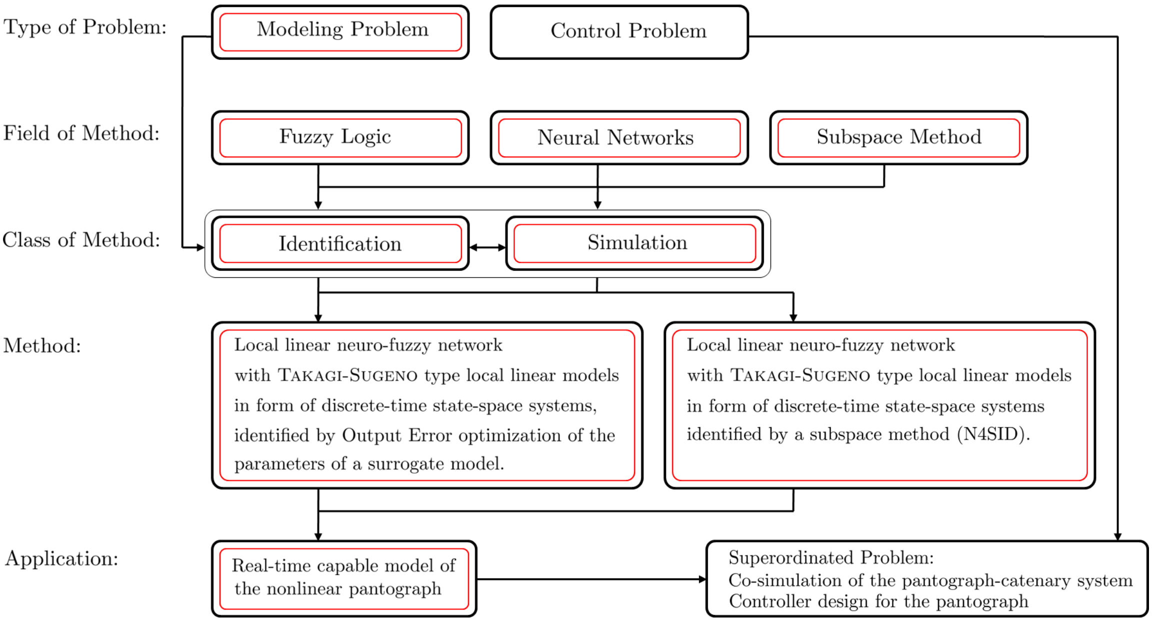


Figure 3.1: Overview and interplay of the applied methods which led to the desired pantograph model. The blocks with red borders were treated in this thesis. Additionally the superordinated problem (coupled system) is indicated.

The remainder of this introduction gives a first idea of how these problem areas where approached to receive the desired pantograph model, a detailed discussion is carried out in the following sections.

The **decomposition** of the nonlinear problem is realized through **operating regime partitioning** of a distinguished input variable to the model. The nonlinear parameters of the validity functions (hidden layer/rule premise) according to this partitioning are fully determined by **expert knowledge** (no nonlinear optimization technique was employed). This approach was possible due to the simple one dimensional partition space of the pantograph's local linear neuro-fuzzy model and the structure of this type of model (see Section 3.3).

The structure is determined by utilizing fully determined **Gaussian membership functions** (MSFs) as validity functions and MIMO discrete-time state-space systems inside the local linear models (LLMs). This thesis follows up with two approaches regarding the configuration of the state-space systems, namely

- a MIMO discrete-time state-space model based on a mechanical **surrogate model** with identifiable parameters (see Section 3.4), and
- a MIMO discrete-time state-space model in innovation form based on the Kalman-filter problem (see Section 3.5), received by the **numerical algorithm for subspace identification** (N4SID) as stated in [33]).

The surrogate model, designed as three-mass oscillator, was chosen for its capabilities in physical **interpretability** (e.g. provides physical states, see Section 3.4). The parameters are positioned due to the oscillator equations of motion and identified by an output error (OE) optimization method. This approach leads to a **light gray box model**, resulting in more interpretability for less performance tradeoff.

The model based on state-space matrices provided by a subspace identification method (N4SID) was chosen with the idea to exploit the great potential of subspace methods to derive a state-space system directly from data in a very **efficient** fashion (see Section 3.5) by incorporating it into the local linear neuro-fuzzy structure. This approach leads to a **dark gray box model**, resulting in less interpretability for more performance tradeoff.

Finally two **methods of blending** are provided to receive a global output:

- **Output Blending**: The global output is received by blending the state-space system outputs of the individual LLMs employing membership functions evaluated in each time-step (see Section 3.6.1 and Figure 3.18). This is the common method for blending in linear model networks and applied when using the LLMs containing the state-space systems identified by the described subspace method.

- **Parameter Blending**: Through blending of the parameters of the state-space matrices of the individual LLMs employing membership functions evaluated in each time step, an interpolated global model is computed in each time-step. Its output represents the global output (see Section 3.6.2 and Figure 3.19). This blending method is implemented with the LLMs based on the surrogate model.

3.2 Local Linear Model Network (LLMN)

This section treats the methodology of the modeling structure applied in this thesis, namely a **local linear neuro-fuzzy model based on first order Takagi-Sugeno type local linear models in state-space configuration**. This model structure combines the methodologies of neural networks and fuzzy logic to - as described above - construct a network that consists of a certain number of simple linear sub-models (neurons containing LLMs) that are primarily valid in a local region of the working range of the problem. By applying this methodology of the - in the literature also referred to as - neuro-fuzzy network type of system or **neuro-fuzzy model**, a nonlinear static model of the pantograph is received. Throughout this thesis, the applied neuro-fuzzy network type of system will be referred to either as local linear model network (LLMN) in state-space configuration, linear model network (LMN) or simply (desired or developed) pantograph model.

In general there exist several different types of nonlinear static models that can be utilized to solve a nonlinear modeling problem (performance of these models depends

Field of Method	Method	Discription	Relevance for pantograph modeling
Classical Nonlinear Modeling	Linear Model	simplest model, the nonlinearity is discounted, the problem is linearized around a certain operating point	compare with the **linearized white-box model** (Section 2.1)
	Polynomial Model	extension of the linear model, more flexible, oscillating interpolation and extrapolation behaviour	
	Look-Up Table Model	dominant nonlinear modeling approach for industrial applications (real-world implementations), low dimensional problems, low computational evaluation demand, parameters are derived directly from measurement, no parameter estimation, non-differential mappings due to linear interpolation rule can be problematic in control tasks	implemented in cooperation with the LLMN based on subspace identified state-space systems (copes only with zero mean signals) to level the offset (see Section 3.5.4)
Neural Networks	Multilayer Perceptron (MLP) Network	most widely applied neural network architecture, high dimensional problems, no interpretation capabilities, expert knowledge cannot be incorporated	
	(Normalized) Radial Basis Function ((N)RBF) Network	local basis functions, low and medium dimensional problems, interpretation capabilities due to construction mechanism, expert knowledge can be incorporated	
Neuro-Fuzzy Models	Singleton Neuro-Fuzzy Model	interpretation of grid-based basis functions as membership functions, comparable with NRBF networks, low dimensional problems	
	Local Linear (Takagi-Sugeno) **Neuro-Fuzzy Model**	linear models in the neurons, good interpretation capabilities, efficient training algorithms low to medium dimensional problems	LOLIMOT algorithm (see Section 3.2.4), the **LLMN pantograph model** is based on this model structure (see Section 3.2.3)

Table 3.1: Overview of possible approaches to receive a static model to a nonlinear problem (compare [19, page 451, Chapter 15]).

on the modeling goals and is in general application specific). Table (3.1) gives an overview of available approaches to nonlinear modeling.

In the following sections a brief introduction to neural network architecture in combination with fuzzy logic is given, focusing on the similarities and differences in these methodologies. Historically the basic principles of neuro-fuzzy models were developed independently in different disciplines, using different specific trems and names but close links to similar model architectures. The following excursion into neural network theory fundamentals can be seen as an attempt to on the one hand bring some light into this at first sight confusing part of the nonlinear modeling world, and on the other hand to give the reader the ability to comprehend *why* the neuro-fuzzy network type of system was chosen as the basis for the pantograph model.

3.2.1 Neural Networks

The motivation for the introduction of artificial neural networks came from the wish to model biological structures (e.g. brains of humans or animals) to imitate nature's information processing techniques, which enable learning and adaptation (e.g. to the environment).
According to the notion in [19], a neural network model is defined as a basis function network with the property that all its basis functions are of the same type and differ

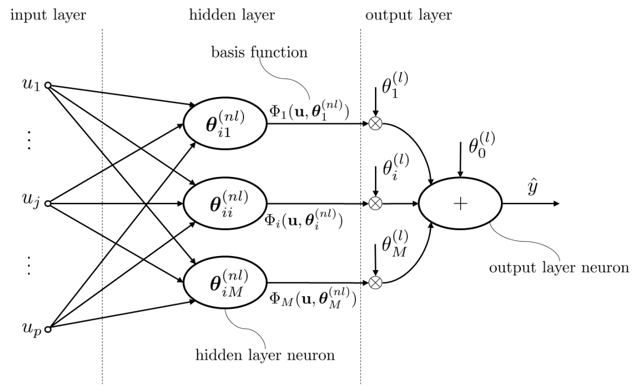


Figure 3.2: Scheme of a *MISO neural network with one hidden layer* defined as a basis function network where all its basis functions are of the same type (compare with equ. (3.1)).

only in their parameters. This kind of model is then referred to as *artificial neural network* (ANN) or simply *neural network* (NN) (see Figure 3.2 for illustration).
Figure 3.2 shows a neural network with a single hidden layer. The hidden layers of a NN each contain a certain amount of nodes/neurons, which embody basis functions Φ_i of a certain type, depending on their nonlinear parameter vector $\boldsymbol{\theta}_i^{(nl)}$ and the input vector $\mathbf{u}$. The output layer of the NN contains the output neuron which commonly is set as a linear combiner, therefore depending on an additional set of linear parameters $\theta_i^{(l)}$.
The hidden layer of the NN can be interpreted as the rule premise structure of a fuzzy model, while the output layer of the NN would correspond to the rule consequent structure of the fuzzy model (see Section 3.2.2).

According to the specific type of the hidden layer neurons' basis functions, there are three classes of neural network architectures presented, which are commonly applied (all belong to the class of *universal approximators*):

- **Multilayer perceptron** (MLP) networks
- **Radial basis function** (RBF) and normalized radial basis function (NRBF) networks
- **Neuro-fuzzy** networks (NF)

The extended basis function formulation as described in [19, page 211, equation (9.3)] for *multi-input single-output* (MISO) continuous-time systems reads as follows:

$$\hat{y} = \sum_{i=0}^{M} \theta_i^{(l)} \Phi_i(\mathbf{u}, \boldsymbol{\theta}_i^{(nl)}), \tag{3.1}$$

with $\boldsymbol{\theta}_i^{(nl)}$ ... nonlinear hidden layer parameters, $\theta_i^(l)$ ... linear output layer parameters and the dummy basis function $\Phi_0(\cdot) = 1$ for the offset parameter $\theta_0^{(l)}$.

The basis functions of these network architectures are dependent on a construction mechanism x_i, which preprocesses the inputs, and a subsequent nonlinear activation function $g(x_i)$, which realizes the basis functions. The **construction mechanisms** of these classes of networks can be divided into:

- **ridge construction** (MLP): projection of the input vector $\mathbf{u}$ onto a nonlinear parameter vector $\boldsymbol{\theta}_i^{(nl)} = [w_{i0}, ..., w_{ij}, ..., w_{ip}]^T$ (compare [19, page 249, equ. (11.14)]) containing the hidden layer weights w_{ij}, with (compare [19, page 254, equ. (11.20)])

$$x_i = \sum_{j=1}^{p} w_{ij} u_j \tag{3.2}$$

- **radial construction** (RBF): computation of a norm (e.g. Euclidean norm or Mahalonobis norm) of the distance between the input vector $\mathbf{u}$ and the centers vector $\mathbf{c}_i$ of the basis functions, with (compare [19, page 265, equ. (11.33) and (11.35)])

$$x_i = \|\mathbf{u} - \mathbf{c_i}\|_{\boldsymbol{\Sigma}_i} = \sqrt{\sum_{j=1}^{p} \left(\frac{u_j - c_{ij}}{\sigma_{ij}} \right)^2} \tag{3.3}$$

- **tensor product construction** (NF): implementation of a certain division strategy (e.g. operating regime decomposition) resulting in univariate functions (depending on several nonlinear parameters) that are defined for each selected input ($\mathbf{z}$) of the input vector $\mathbf{u}$, e.g. membership function (MSF, compare Figure 3.4, equ. (3.11) and according sections).

The activation functions for the MLP network are typically chosen to be of saturation type, e.g. sigmoid functions such as logistic function or hyperbolic tangent. Whereas for the RBF and the NF type of networks the choice of activation functions with a maximum at the center of the neuron ($x_i = 0$) is aspirated, to enhance the neurons validity around it's positioning (local character). The basis function itself however is then computed by (compare [19, page 264, equ. (11.35)])

$$\Phi_i(\mathbf{u}, \theta_i^{(nl)}) = g(x_i), \text{ with } g(\cdot) \text{ ... activation function.} \tag{3.4}$$

A typical choice for an RBF or NF network is the Gaussian function, where the nonlinear activation function reads as (compare [19, page 264, equ. (11.31)])

$$g(x_i) = exp\left(-\frac{1}{2} x_i^2\right). \tag{3.5}$$

Figure 3.3 shows the presented network architectures in linear model network configuration, to illustrate the differences in these approaches. The obvious difference lies in the generation of the neuron output. The conventional network structures (MLP, RBF, compare equation (3.1)) neurons simply multiply their basis function with a constant term (output layer weights, w_i), while the NF network (compare equation (3.98)) provides an e.g. linear function (lin.sys.$_i$) in each neuron, depending on the model input $\mathbf{u}$. The neurons of the NF network are called local linear models (LLMs) and can represent different types of linear systems. This can be e.g. a linear regression model (e.g. ARX) in polynomial form (see LOLIMOT in Section 3.2.4), or a system of differential equations in state-space form (see developed pantograph model as LLMN based on a three-mass oscillator).

Figure 3.4 shows the i-th neuron of a RBF network and a neuro-fuzzy model in comparison. Although there is a difference in the computation path of the basis functions Φ_i, they both deliver the same result if Gaussians are used as activation function in the RBF network neuron, respectively if the MSFs are set as axis-orthogonal Gaussians and the product operator is used for the conjunction of the MSFs. This equality can be shown with equations (3.3) and (3.5) of the RBF network and equations (3.11) and (3.10) of the NF network respectively. Otherwise a strong similarity is still obtained (compare [11]).

Training Procedure: Parameter Optimization/Identification

The strategies for training of a neural network, i.e. identifying the network parameters, are just mentioned here for the sake of completeness, because they mainly deal with the determination of the nonlinear parameters (hidden layer weights, parameters of the activation function). These nonlinear parameters (MSF, rule premise) will be determined due to expert knowledge for the developed pantograph models and not estimated. For further details on the topic consult e.g. [19, page 253, Section 11.2.4] for MLP network training or [19, page 269, Section 11.3.3] for RBF network training.

For a **MLP network** two sets of parameters have to be determined during the training procedure, which are the nonlinear hidden layer weights and the linear output layer weights. In any case an initialization method has to be applied to determine the initial values of the hidden layer weights before the training can be started (e.g. similarly scaled small values), with the limitation that they do not provide any interpretation capabilities. For the training of a MLP network three strategies are commonly applied:

- Regulated Activation Weight Neural Network (RAWN) approach, where the nonlinear hidden layer weights are just initialized and the linear output layer weights are estimated subsequently by a least squares technique. There exist extensions and improvements to this approach. It is recommended for low-dimensional problems only.
- Nonlinear optimization approach, where all weights (hidden and output) are estimated simultaneously by a local or global optimization technique. This represents the most common approach, which utilizes gradient-based learning

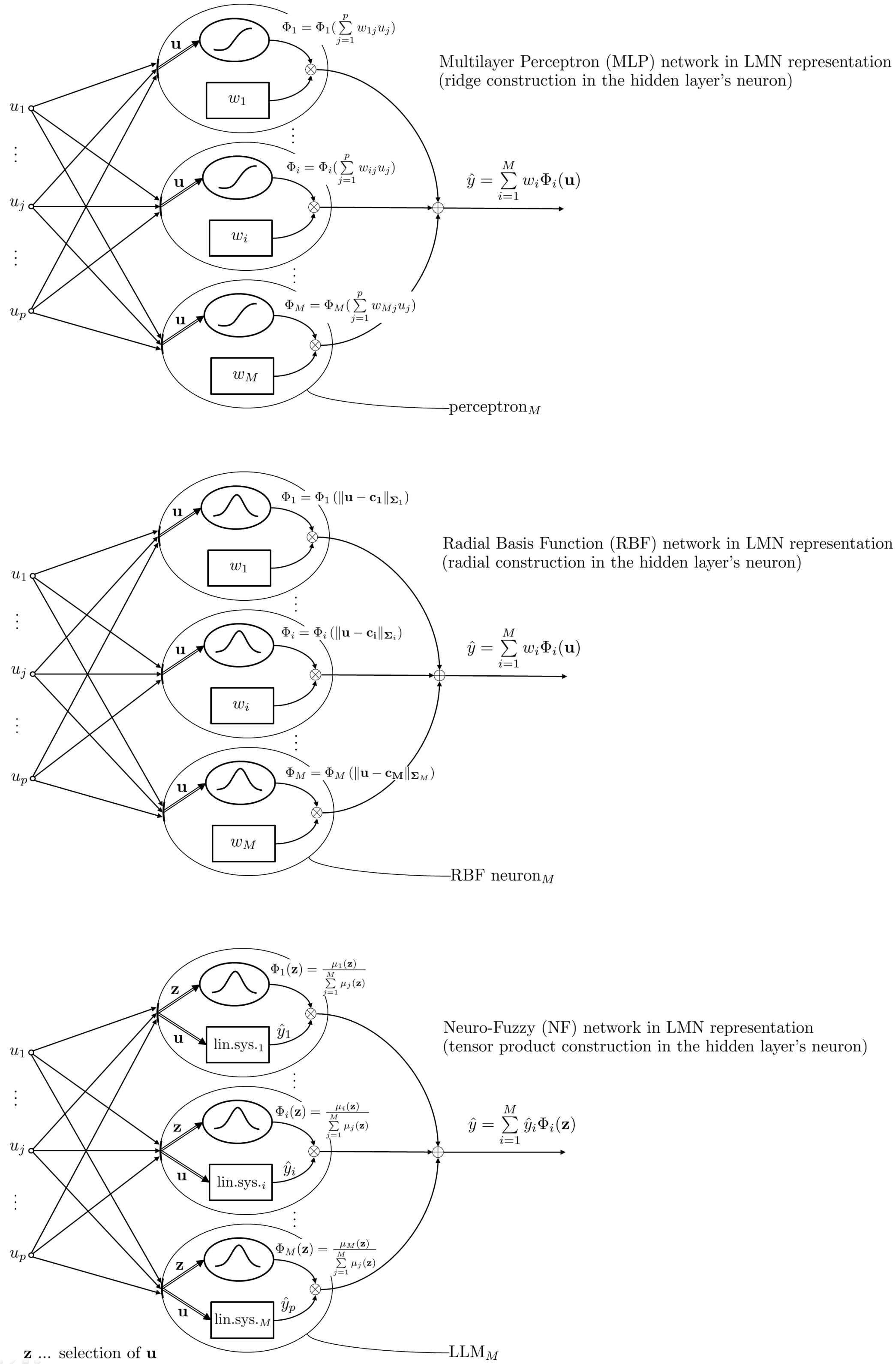


Figure 3.3: A selection of neural network architectures - MLP top, RBF middle, NF bottom - in linear model network (LMN) representation.

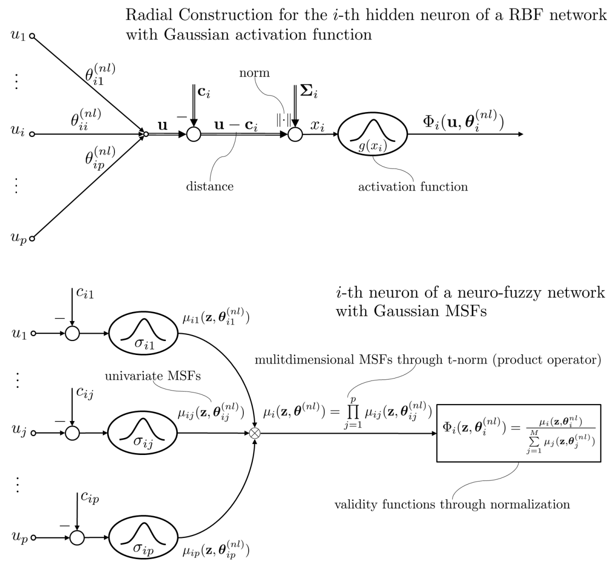


Figure 3.4: Comparison of the structure of the i-th neuron of a MISO radial basis function network and a MISO neuro-fuzzy network, with p inputs and M neurons (compare equations (3.3) and of the RBF network and equations (3.11) and (3.10) of the NF network).

rules where a certain loss function is optimized according to a learning rate (step size).

- Staggered or nested training approach, which is a combination of the two approaches presented above. Here the aim is to exploit the advantages of both approaches by applying the nonlinear optimization only for estimating the hidden layer weights and staggering or nesting in the least squares optimization of the output layer weights.

The training of a **RBF network** demands the determination of the parameters of its basis function (nonlinear hidden layer weights) and a subsequent estimation of the output layer weights, very similar to the training of a MLP network (RAWN approach). The main difference is the geometric interpretation capability of the nonlinear parameters which can be exploited. Commonly the positioning of the center of the basis function is determined first, while all the other parameters are determined subsequently. Possible approaches are:

- Random Center Placement, where the centers are randomly determined (RAWN approach).
- Clustering for Center Placement, which is based on clustering techniques (see e.g. [19, page 142, Section 6.2]), where groups of data are searched out of a data set, that possess some kind of similarity. Improvement of the random approach.

rule premise	rule consequent
same for all fuzzy systems	fuzzy system dependent
fuzzification → aggregation → activation → accumulation → defuzzification	

Figure 3.5: Inference of a fuzzy system, by deriving an output fuzzy set given the fuzzy rules and the known inputs.

- Complexity Controlled Clustering for Center Placement, where the complexity of the underlying model is incorporated for further improvement.
- Grid-Based Center Placement, which is an alternative to the clustering based approaches where the centers are placed on a certain grid of the input space. Recommended for low dimensional problems only.
- Subset Selection for Center Placement, which is based on a subset selection technique (see e.g. [19, page 67, Section 3.4]).
- Nonlinear Optimization for Center Placement, which is a straightforward nonlinear optimization of the hidden layer parameters, with good initial values due to interpretation capabilities.

3.2.2 Fuzzy Systems

Fuzzy logic was invented as an extension to *Boolean logic* by allowing the assignment of any value in the interval $[0, 1]$ to a variable, instead of either 0 ($false$) or 1 ($true$). This "fuzzy" assignment of a variable was inspired by human thought patterns and human communication, which often is based on vague and uncertain, maybe insufficient information, resulting in unprecise (fuzzy) statements. Therefore this type of systems incorporate a great potential for interpretation, because it somehow contains the spirit of human nature. Fuzzy systems are, as well as the neural networks, part of the class of universal approximators (compare [15] and [41]).

The developed pantograph model is designed as a *neuro-fuzzy network based on linear local models*, which are realized as first order Takagi-Sugeno fuzzy systems. In general however a neuro-fuzzy network architecture could be based on a variety of fuzzy systems. A fuzzy system can be divided into two major parts, the *rule premise* and the *rule consequent* (compare Figure 3.5). The output is referred to as *inference* of a fuzzy system. *Approximate reasoning mechanisms* based on fuzzy logic were consequently developed to cope with linguistic statements in a rule-based form (see Figure 3.6), which will be discussed here (see example in Figure 3.6 for illustration).

In the *fuzzification* stage (see Figure 3.5) a nonlinear transformation of the inputs (called linguistic variables, e.g. operating height) from a crisp value to a fuzzy value

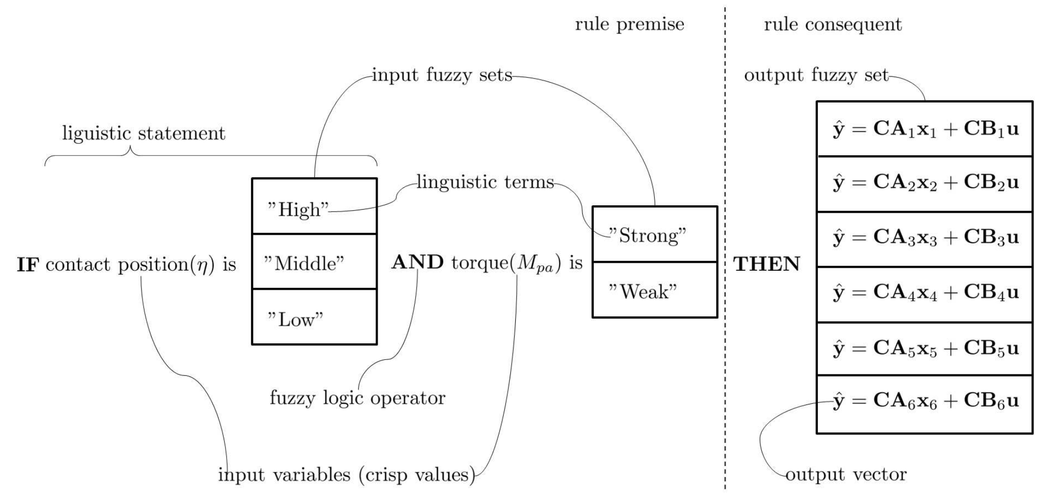


Figure 3.6: Example of a MIMO Takagi-Sugeno fuzzy system with two inputs $\mathbf{u} = [\eta, M_{pa}]^T$ and six outputs $\hat{\mathbf{y}} = [\xi, \dot{\xi}, \zeta, \dot{\zeta}, \delta_M, \dot{\delta}_M]^T$ illustrated for the pantograph (compare Section 3.4.3).

is performed, utilizing the univariate ***membership functions*** (MSFs):

$$\mu_i(u_j) : \mathbb{R}^1 \rightarrow [0, 1], \tag{3.6}$$

with $i = 1, ..., M$ denoting the i-th of M partitions (number of linguistic terms) of the j-th input (number of linguistic variables).
The fuzzy value is referred to as the *degree of membership* to an according *linguistic term* (e.g. "Low'), which is defined by its MSF $\mu_i(u_j)$ (e.g. $\mu_1(\eta)$).
If there are multiple linguistic variables available in a fuzzy system, they have to be combined using fuzzy logic operators (*t-norms*, e.g. AND) determining the *degree of rule fulfillment*.
A *fuzzy rule* is then formulated by assigning certain linguistic terms of the input fuzzy sets to an output fuzzy set. Completing this step, as mentioned above, is setting up the inference of the fuzzy system, which is connecting the rule premise to the rule consequent. A selection of the commonly utilized fuzzy systems is given by the following list (these systems differ only in their rule consequent):

- **linguistic fuzzy systems** (Mamdani fuzzy systems): in the rule consequent first the output activation of all rules is computed (activation) utilizing arbitrary output membership functions and fuzzy operators. Then these output activations are joined (accumulation) using fuzzy operators and eventually a crisp out value is generated by a final defuzzification step applying a certain method (e.g. center of gravity) to the joined output MSFs. Alternativley a fuzzy output is resulting.

- **singleton fuzzy systems**: simplification to linguistic fuzzy systems by using singleton output MSFs, therefore the output fuzzy set contains constant values

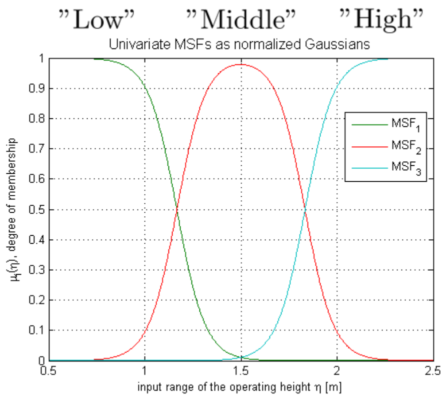


Figure 3.7: According membership functions $\mu_i(\eta)$ to the example Takagi-Sugeno fuzzy model in Figure 3.6, defining the linguistic terms "Low", "Middle" and "High" of the linguistic variable contact position η.

s_i, which determine the position of the trivial output MSF. The singleton fuzzy output is then computed by

$$\hat{y} = \frac{\sum_{i=1}^{M} s_i \mu_i(\mathbf{u})}{\sum_{i=1}^{M} \mu_i(\mathbf{u})} \tag{3.7}$$

- Takagi-Sugeno **fuzzy systems**: extension to singleton fuzzy systems where the rule consequent does not contain fuzzy sets, but linear functions (a zero-th order consequent would deliver again a singleton fuzzy system). The output of a first-order Takagi-Sugeno fuzzy system is computed by

$$\hat{y} = \frac{\sum_{i=1}^{M} f_i(\mathbf{u}) \mu_i(\mathbf{u})}{\sum_{i=1}^{M} \mu_i(\mathbf{u})} \tag{3.8}$$

An example for a possible first-order Takagi-Sugeno fuzzy rule could read as follows (compare Figure 3.6):

R_2: **IF** contact position is "Low" **AND** torque is "Strong" **THEN** $\hat{\mathbf{y}} = \mathbf{C}\mathbf{A}_2\mathbf{x}_2 + \mathbf{C}\mathbf{B}_2\mathbf{u}$.

This example illustrates the main mechanisms that will be utilized in the local linear model network (LLMN). The rule consequent's output fuzzy set can be interpreted as a *local linear model* (LLM) containing a state-space system. The fuzzyfication step (determination of degree of membership utilizing a membership function) in the rule premise can be interpreted as the (operating regime) partitioning of an input variable using the LLMN structure.

Training Procedure: Parameter Optimization/Identification

Like in Section 3.2.1, the training strategies for RBF networks are also just mentioned. For further details consult e.g. [19, page 313, Section 12.3.3].
In general the training of fuzzy models is similar that of RBF networks commonly realized by a grid based center placement approach. It is recommended however to determine the nonlinear premise parameters (MSF) according to prior knowledge to keep the strengths of the fuzzy model's transparency in regard to interpretation of its parameters. This is also the approach taken for the development of the presented pantograph LLMNs. If however little prior knowledge is available, the premise parameters can be estimated employing nonlinear local or global optimization techniques. If absolutely no expert knowledge is present, a global nonlinear optimization method is suggested, e.g. a genetic algorithm, to prevent ending up in a local optimum. If however good initial values can be found, a local nonlinear optimization is the way to go. The rule consequent parameters (linear) can be estimated by a least squares technique subsequently to the determination of the rule premise parameters.

Another possibility provided by fuzzy models is the optimization of the rule structure (see e.g. LOLIMOT algorithm in Section 3.2.4), where the optimal model complexity is sought. Here the nonlinear global search methods play an important role. Finally it has to be mentioned, that other components of a fuzzy system can be optimized as well, e.g. the fuzzy operators or the defuzzification method. This section is ended up by a list of possible schemes which optimize several of the discussed parts of different type neuro-fuzzy networks (see [19, page 323, Section 12.4] for details):

- Nonlinear Local Optimization
- Nonlinear Global Optimization
- Orthogonal Least Squares Learning
- Fuzzy Rule Extraction by a Genetic Algorithm (FUREGA)
- Nested Least Squares Optimization of the Singletons
- Constrained Optimization of the Input Membership Functions
- Adaptive Spline Modeling of Observation Data (ASMOD)

3.2.3 Local Linear Neuro-Fuzzy Model

This section now treats the methodology of the implemented nonlinear modeling architecture - a **local linear neuro-fuzzy model based on local linear models** implemented as first-order Tagaki-Sugeno fuzzy systems - furthermore referred to as *local linear model network* (LLMN) *in state-space configuration*.

This model structure is based on an neural network in the neuro-fuzzy architecture (see Section 3.2.1 and Figure 3.3 bottom) where the fuzzy system's rule premise

neural networks	**fuzzy system**	**linear model network** (local linear neuro-fuzzy model based on first order Takagi-Sugeno fuzzy systems)	type of **parameters** present in that part of the structure
output layer	rule consequent	local linear models (e.g. in state-space system configuration)	linear parameters
hidden layer	rule premise	operating regime partitioning using fuzzy logic	nonlinear parameters

Table 3.2: Summary of specific terms for the presented model architectures: neural networks, fuzzy systems and the linear model network.

and consequent methodology is incorporated into its hidden layer neurons (compare Section 3.2.2 and Figure 3.6). In this configuration the hidden layer's neurons now contain both the hidden layer's nonlinear and the output layer's linear parameters. The nonlinear hidden layer parameters are the nonlinear parameters of a fuzzy rule's premise structure, which incorporates Gaussian membership functions. The output layer's linear parameters are the linear parameters of a Takagi-Sugeno fuzzy system's output fuzzy set, which incorporates linear (in case of the surrogate model partial differential) equations in state-space form. The output layer neuron simply sums up the outputs of the individual hidden layer neurons (output blending see Section 3.6.1.
A slightly different approach is also implemented, where the neuro-fuzzy network is implemented parallel to a plant model, having the parameters of this plant model as its output (parameter blending see Section 3.6.2).

Remark 3.2.1. The decomposition of the nonlinear problem into linear subproblems is commonly done by an operating regime decomposition approach. In case of the applied local linear neuro-fuzzy network, the incorporated fuzzy logic can lead to a smooth partitioning of the input space (desired behaviour in this application case) if the activation function is chosen appropriately (e.g. Gaussian function). An overview of alternative local-linear neuro-fuzzy model architectures can be found in [18, page 3, Chapter 1], which includes a discussion of different operating regime approaches in [18, page 5, Section 1.2]. For a detailed discussion of the applied decomposition see Section 3.3.

Remark 3.2.2. With regard to the stability of the local linear neuro-fuzzy network in state-space configuration, it has to be stated, that only conservative, necessary but not sufficient conditions exist, with which the stability of a LLMN can be examined. This topic will be further discussed in Section 3.6.3.

A summary of the specific terms of the presented model architectures can be found in Table 3.2.

Training Procedure: Parameter Optimization/Identification

A great strength of fuzzy models is, that their defining parameters can be specified either by qualitative expert knowledge, by data-driven identification by measurement data or any combination of these two extreme approaches. Therefrom a ***interpretability-performance tradeoff*** results. Data-driven identification works

more accurate if the model is more flexible. The flexibility however is limited to keep the interpretability capabilities, while the expert knowledge is incorporated to increase the performance where possible.
For the developed pantograph model the main motivation for applying this type of systems were:

- The exploitation of expert knowledge: incorporation of expert knowledge into the model before and during identification (engineering cycle).
- Improved understanding of the model process: interpretation of the model obtained by identification, therefore additional information about the real-world nonlinear pantograph can be extracted from the model.

Therefore the model was developed by setting all nonlinear parameters of the model (operating regime partitioning) due to prior knowledge, while identifying the linear parameters of the model (LLMs) in a data-driven fashion (output error optimization method). In the first approach, additionally the linear parameters where kept as few as possible by exploiting the expert knowledge when generating the parametrized state-space systems (inside the LLMs, see Section 3.4). In the second approach the implementation of subspace methods leads to an iteration-free determination of the state-space matrices (see Section 3.5).
Finally it can be mentioned, that neuro-fuzzy network modeling is utilized in advanced-control solutions (e.g. Fuzzy-MPC) and already applied in some industry branches (for criteria on successful applications see [22], for an overview on successful applications see [23] and [24]). It beholds a great potential as a tool to examine nonlinear problems and develop efficient models.

3.2.4 Local Linear Model Tree (LOLIMOT) Algorithm

An algorithm based on local-linear neuro-fuzzy models that can be used for the identification of MISO systems is called LOLIMOT, postulated by Nelles. This algorithm was utilized in a first attempt to receive the desired pantograph model and led to certain insights and conclusions. The main findings of applying this methodology to the pantograph problem are presented in this section in a narrative fashion.

The *Local Linear Model Tree* (LOLIMOT) algorithm - as proposed by Nelles in [19, page 365, Section 13.3.1] - is an incremental tree-construction algorithm with axis-orthogonal input space partitioning, based on linear local models in ARX-form (see equation (3.9)), whose parameters are identified with a local weighted least squares algorithm from data. The premise structure is optimized by an iterative heuristic search method. LOLIMOT starts in an outer loop with an initial model structure (e.g. 2 LLMs equidistantly positioned on the partition space) and estimates the according LLM parameters in an inner loop with a regression method from data. After a certain threshold value of accuracy or iterations is reached (early stopping), the

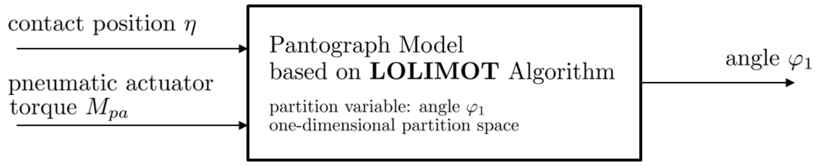


Figure 3.8: Input/Output configuration for the pantograph model based on the LOLIMOT algorithm.

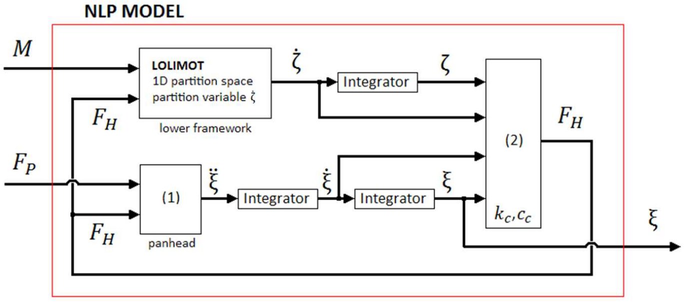
partition space - according to the worst performing LLM - is split in half and the optimization of the parameters of the newly set LLMs starts and so on. For a more detailed description see [19, 20, 21].

Definition of the *autoregressive with exogenous input* (ARX) model (compare e.g. [19, page 466, equation (16.10)]):

$$y(k) = \frac{B(q)}{A(q)}u(k) + \frac{1}{A(q)}v(k), \text{ with } v(k) \ \ldots \text{ white noise} \tag{3.9}$$

The algorithm, as described by Nelles is considerably fast (no nonlinear structure optimization, heuristic search method) and can cope with multidimensional inputs but only single outputs (MISO systems only). This algorithm was used in a first modeling approach to the nonlinear pantograph problem. Due to prior knowledge, several choices like the order of the ARX-polynomials (corresponding to the amount of DOF of the white-box model) as well as the input and output signals could be made. In a first attempt, the identification of the whole mechanism was carried out, using the inputs as specified in equation (2.2), but an arbitrary, eventually measurable variable as output like the angle of the lower bar φ_1 as sketched in Figure 3.8. This kind of application of the LOLIMOT algorithm employed output partioning in an *external dynamics approach with global state feedback* (for further information see [19, page 603, Fig. 20.1a]). To incorporate even more expert knowledge to the model (to make it more white), in a second attempt a pantograph model was developed, where the LOLIMOT algorithm was only used to identify the nonlinear part of the pantograph (the collector head is a known linear one-mass oscillator). This early draft of the pantograph LLMN can be seen in Figure 3.9. Additionally, to fulfill the specifications as defined in Section 1.2.1 a further attempt was made to alter the algorithm in a way to receive the identified LLMs in state-space system form instead of ARX polynomials. This was realized by a conversion of the identified ARX-models polynomials to *non-minimal state-space* (NMSS) systems (the system matrices are build from the coefficients of the numerator and denominator polynomials according to [12, page 63, Section 6.3]). These state-space systems were used to replace the ARX-models in the local linear models before simulation.

However the simulation results where not satisfactory (see Section 4.3), because the generated output of the model configured in the described form became unstable (see Figure 4.6), even if the converted NMSS systems themselves where all stable



$$m_c\ddot{\xi} = -m_c g + F_H - F_P \qquad (1)$$

$$F_H = k_c(\zeta - \xi) + c_c(\dot{\zeta} - \dot{\xi}) \qquad (2)$$

Figure 3.9: Scheme of an early draft of the pantograph model based on the LOLIMOT algorithm.

(see Figure 4.7). Additionally, in some cases the conversion of the ARX-polynomials delivered unstable state-space systems, which led to an indefensible simulation performance.
Another issue is that LOLIMOT is a tree-construction algorithm, where the user has no influence on the partitioning of the selected input space. In practice the algorithm tends to split the local linear models in regions where the most data points are present in the training data set (based on the idea that more data holds more information). This property of the algorithm is not desirable in this specific application, because therefore the partitioning becomes dependent of the training data set employed, while the expert knowledge of the nonlinear pantograph demands a fixed partitioning.

Finally the states and system matrices allow no physical interpretation due to the NMSS form of the LLMs and solely MISO systems are realizable.
At that point, with the issues described above, it came clear that the local linear neuro-fuzzy models based on ARX-models with regression methods for estimation are not suitable to develop a pantograph model with the specifications set in Section 1.2.1 and a different path had to be pursued.

3.3 Operating Regime Decomposition, Partitioning

This section treats the implementation of the concept of **operating regime decomposition** for the pantograph problem as the first step of constructing a local linear neuro-fuzzy network. To reveal the idea behind the decomposition approach, the ***divide-and-conquer strategy*** as stated in [18] is repeated here:

"A complex problem is somehow partitioned into a number of simpler subproblems that can be solved independently, and whose individual solutions yield the solution of the original complex problem." ([18, page 4])

With other words the *divide and conquer* approach to a nonlinear problem requires a decomposition into linear subproblems that are easier to solve. Therefore the whole operating range of the nonlinear system gets partitioned (via a suitable partition variable, e.g. contact position η) into smaller operating regimes in which the according linear subsystems (local linear models, LLMs) are valid (determined via a membership function). The main idea could be formulated as a term equation:

Global approximation = Interpolation of the local approximations

The **key** to this decomposition problem is to find a suitable quantity that characterizes the operating range (could be high-dimensional in general, for the pantograph problem the operating height is set as an one-dimensional partition space) and along which the nonlinear problem can be partitioned into several operating regimes. These operating regimes then form a complete partition of the operating range, without any overlap. The interpretability is good as long as the number of partition variables (rule premise and number of fuzzy rule sets, see Table 3.2) is small.

There exist two major techniques of how to determine the partitioning of the input space, that are in general applied complementary:

- Expert knowledge.
- Nonlinear structure identification from data.

For the development of the pantograph models in this thesis, the partitioning is determined **solely by expert knowledge**, therefore no nonlinear structure identification of the premise parameters is carried out (compare Section 3.2.3).

What follows is the definition of the validity functions Φ_i. They are chosen as ([19, page 343, equ. (13.4)])

$$\Phi_i(\mathbf{z}(k)) = \frac{\mu_i(\mathbf{z}(k))}{\sum\limits_{j=1}^{M} \mu_i(\mathbf{z}(k))}, \tag{3.10}$$

with M the number of LLMs and time step k. The activation functions $\mu_i(\mathbf{z}(k)) = \mu_i(\eta(k))$ are chosen as *normalized Gaussians* and are explicitly given by (compare with [19, page 343, equ. (13.5)] for one-dimensional partition space)

$$\mu_i(\eta(k)) = \exp(-\frac{1}{2}(\frac{(\eta(k) - center_i)^2)}{(k_\sigma \cdot spread_i)^2}), \tag{3.11}$$

where $center_i$ (center coordinates) and $spread_i$ (standard deviations) are the nonlinear parameters of the according weighting function μ_i (see Table 3.3). k_σ represents a tuning parameter introduced by [19, page 362, equ. (13.36)] for the LOLIMOT

number of LLMs	$center_i$	$spread_i$	k_σ [1]
1: LLM_1	0.5	1	
2: LLM_2	0.75	0.5	
LLM_1	0.25	0.5	
4: LLM_4	0.875	0.25	$\frac{1}{3}$
LLM_3	0.625	0.25	
LLM_2	0.375	0.25	
LLM_1	0.125	0.25	

Table 3.3: Parameters of the local linear models for LLMNs with different amount of LLMs (compare with Figure 3.12) for an equidistant partitioning approach.

algorithm, which acts as a proportionality factor between the membership functions extension and the standard deviation and therefore is closely related to the $spread_i$ parameter. In general the product $k_\sigma spread_i$ determines the tradeoff between smoothness of transition between one LLM and another and the locality of each LLMs validitiy. A small value creates step-like validity functions (non-smooth,strong locality), while bigger values blur the area of validity until all LLMs are always valid over the whole partition space (smooth, no locality).
Finally it can be stated, that by choosing the validity functions as described by equation (3.10), they form a *partition of unity*, i.e. all contributions of all LLMs sum up to 100 [%] at each time instance, therefore

$$\sum_{i=1}^{M} \Phi_i(\mathbf{z}(k)) = 1, \tag{3.12}$$

which is necessary to allow proper interpretation of the validity functions.

The validity function vector can then be written as the column vector of all validity function values of all LLMs with

$$\mathbf{\Phi}(\mathbf{z}(k)) = \begin{bmatrix}\Phi_1(\mathbf{z}(k)) & \cdots & \Phi_M(\mathbf{z}(k))\end{bmatrix}^T . \tag{3.13}$$

As it can be seen from equation (3.10) the membership functions (MSFs) are set as non-strictly local equidistantly positioned normalized Gaussians with identical width. An illustration of such a LLMN with four LLMs is depicted in Figure 3.10. This configuration of the premise parameters in general represents a restriction to the neuro-fuzzy model with regard to achievable accuracy, but is executed to guarantee interpretability capabilities and prevent unexpected and undesired normalization side effects (e.g. reactivation, compare [19, page 316, Section 12.3.4]). Furthermore it realizes smooth transitions between the LLMs.

[1] Proportionality factor for the weighting functions extension (steepness). Can be interpreted as an additional tuning parameter according to Nelles LOLIMOT algorithm [19, page 365, Section 12.3.1]. In this thesis not tuned, but set to a value recommended in [19].

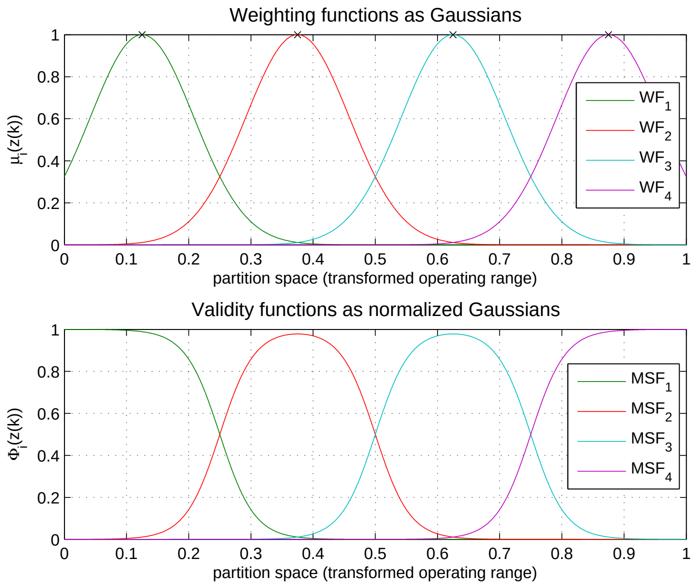


Figure 3.10: The weighting functions $\mu_i(\mathbf{z}(k))$ (upper plot, denoted as WF_i) are chosen as equidistantly positioned Gaussian functions with identical width ($spread_i$). The weighting functions are normalized (see equation (3.10)) to receive the membership functions $\Phi_i(\mathbf{z}(k))$ (lower plot, denoted as MSF_i). Depicted for a linear local model network with 4 local linear models, where the abscissa represents the transformed operating range (partition space).

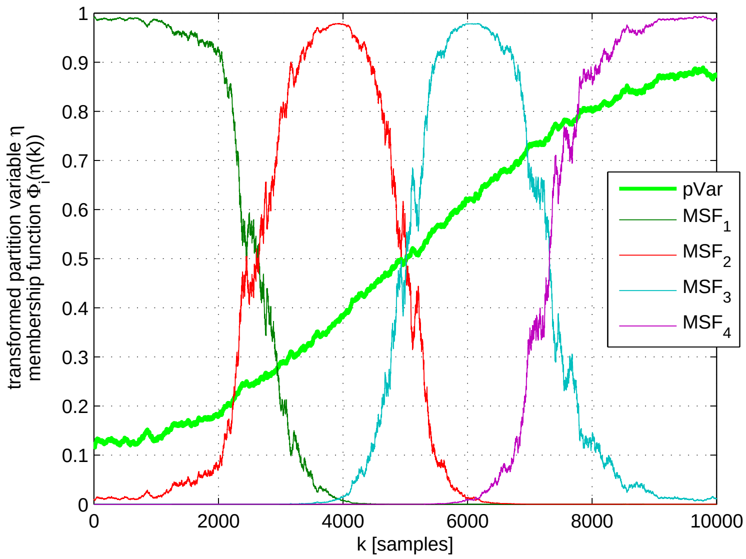


Figure 3.11: Membership functions $\Phi_i(\eta(k))$ for a local linear model network (LLMN) with 4 local linear models (LLMs) over time steps k of a simulation over the whole operating range (sampling time $T_s = 1$ [ms]).

Alternative types of membership functions could be singleton, triangular (linear interpolation) or trapezoidal shaped functions (strictly local).

In Figure 3.11 the membership functions $\Phi_i(\eta(k))$ of a simulation of the LLMN model with 4 LLMs over the whole operating range is shown for every time step k. The green graph (pVar) shows the movement of the partition variable (in this case the contact position η) for each time step, transformed to the range $[0, 1]$. It is clear to see how the LLMs are valid for the global model over the simulation time according to their membership function value. If e.g. $\Phi_i(\eta(k)) = 1$ the associated LLM_i would be 100% valid in this time step. The validity functions can be interpreted as operating point dependent weighting factors. Furthermore an illustration of the operating range, its partitioning and the positioning of the local-linear models can be seen in Figure 3.12.

In this thesis simply the case of a single rule premise (operating height, one-dimensional MSF) is treated and therefore a discussion of fuzzy logic operators (e.g. *t-norms* and *t-conorms*) which are utilized to combine several fuzzy rules (multi-dimensional MSFs) is obsolete. For further information on the topic consult e.g. [19, page 302, Section 12.1.2]).

As described above the input variable $u_1 = \eta$ (contact position) was chosen as the partition variable. This realization is referred to as **input partitioning**, where an input signal to the LLMN is utilized to determine the activation of the LLMs. An alternative route would be to choose an output variable as the partitioning variable,

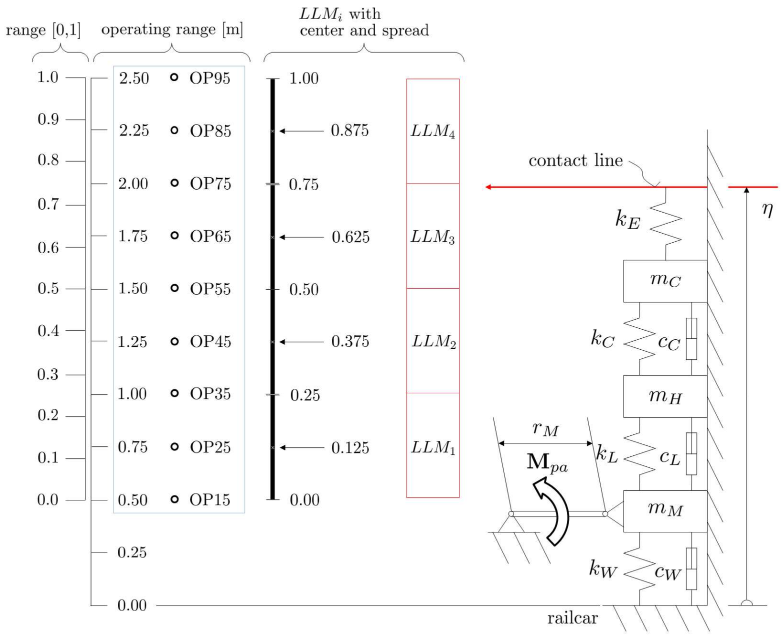


Figure 3.12: Illustration of the partition space (operating range/height of the pantograph) and the equidistant positioning of the local-linear models along this axis, with the operating points OPXY as defined in equation (4.1).

e.g. the crossbar position ζ. In that case, referred to as **output partitioning**, the selected variable (global LLMN output) is fed back to the LLMN input (global state feedback, external dynamics approach). For further details on that matter see [19, page 601, Chapter 20] and [19, page 603, Figure 20.1a]. Regarding stability of the LLMN, the output partitioning approach has to be seen more critical, as introduces all kind of issues associated with closed loop dynamics to the partitioning procedure.

3.4 Surrogate Model (Parametrized Three-Mass Oscillator)

This section treats the configuration of the state-space models that are based on a mechanical surrogate model of the pantograph, namely a parametrized three-mass oscillator (a scheme thereof is depicted in Figure 3.13). This section will be opened with an introduction to and discussion of general issues that arise in modeling using a parametrized model structure (see Section 3.4.1). Following the equations of motion of the three mass oscillator will be derived (see Section 3.4.2) and transcribed into a state-space system (see Section 3.4.3). Furthermore the parameter vector (see Section 3.4.4) and the according optimization method will be defined (see Section 3.4.5).

3.4.1 Modeling Issues with Parametrized Models

A brief introduction to some concepts and approaches to issues (mainly tradeoffs) that arise in nonlinear modeling problems using parametrized model structures is given here.

model complexity (compare [19, page 157, Section 7.1])
The model complexity is related to the number of parameters that the model possesses (number of parameters can be a measure for model complexity) and give the model its flexibility. However not every parameter has the same influence on model behaviour or is equally important (parameter sensitivity, Fisher information). The key is to determine the *optimal model complexity*, where neither under-fitting nor over-fitting is occuring. By approaching the nonlinear modeling problem by utilizing a local linear Takagi-Sugeno neuro-fuzzy model, another perspective has to be taken into account , which is the ***flexibility-interpretability tradeoff***. The flexibility strengthens the models accuracy and robustness, while the interpretability capability inherited by the model can be used to gain *a posteriori* knowledge about the examined problem (e.g.: Is the model behaviour physically plausible?, compare [18, page 52, Section 1.6]).

bias/variance tradeoff (compare [19, page 158, Section 7.2])
As mentioned above the parameter influence the model performance by giving it flexibility. The model error can be decomposed into the *bias error* and the *variance error*:

- *bias error*: Due to structural inflexibility of the model.
- *variance error*: Due to uncertainties in the estimated parameters.

According to that contemplation the following statements can be made:

- overfitting/overtraining: high variance error
- underfitting/undertraining: high bias error
- simple model (few parameters): high *bias error* but low *variance error*
- complex model (many parameters): low *bias error* but high *variance error*

curse of dimensionality (compare [19, page 190, Section 7.6.1])
This issue is in general relevant for high dimensional problems, which is not the case for the models presented in this thesis (which employ a one-dimensional partition space). This is due to the fact, that a linear increase in the input dimensionality causes an exponential increase of the required data amount. Therefore a common problem in nonlinear modeling is the desire to decrease the input dimensionality. Some concepts for reducing the data amount are given by this list:

- non-reachable regions in the input space

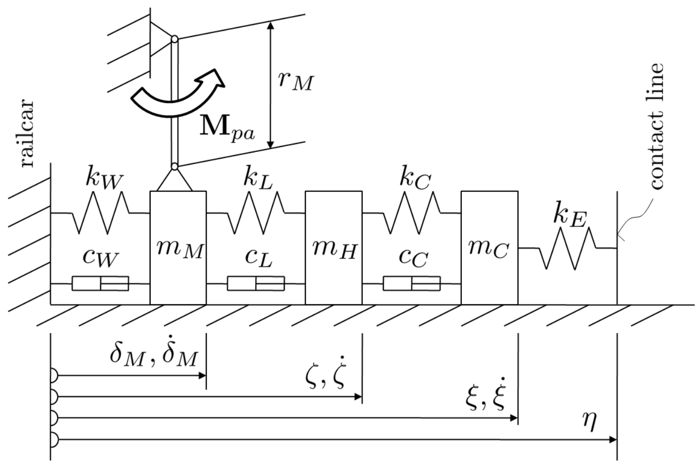


Figure 3.13: Scheme of the three-mass oscillator surrogate model, which represents the underlying model of the pantograph LLMN (surrogate).

- correlated or redundant inputs
- smooth behaving inputs require just sparse data (significantly less data necessary)
- specific application demands different accuracies in different operating conditions (inaccurate model behaviour might be acceptable in some regions)

This introduction into nonlinear modeling with parametrized models is concluded with a quotation from [19, page 192, Section 7.6.1] regarding the relationship between the *curse of dimensionality* and the *bias/variance tradeoff*:

> "Each additional input makes the model more complex. Although each additional input may provide the model with more information about the process this does not necessarily improve the model performance. Only if the benefit of the additional information exceeds the variance error caused by the additional model parameters, will the overall effect of this input be positive. Thus, discarding inputs *can* improve the model performance." ([19], page 192)

With regard to the developed pantograph model, the above statement was taken into account by keeping the model as simple as possible (using a one-dimensional partition space) and incorporating expert knowledge where possible.

3.4.2 Equations of Motion of the Three-Mass Oscillator

In Figure 3.13 a scheme of a three-mass oscillator without the influence of gravity can be seen. The three masses m_C, m_H and m_M are modeled as lumped masses. This surrogate model serves as the underlying system for the local linear models (LLMs) of the LLMN. The incorporated state-space matrices of the LLMs are derived from

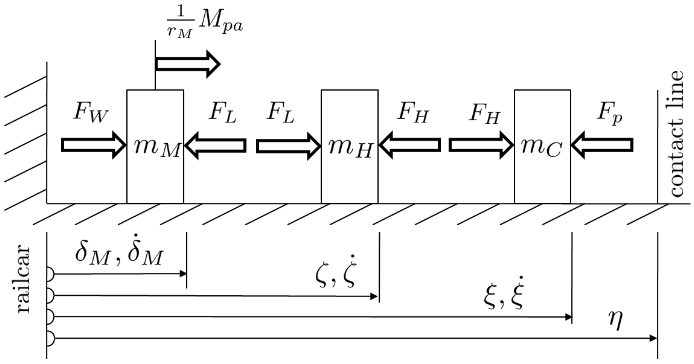


Figure 3.14: Free body diagram for the three-mass oscillator surrogate model, determining the orientations of the cut forces.

the equations of motion of this multiple mass oscillator, while the parameters of this mechanism represent the parameters that will be optimized in a cost function.

For the setting up the surrogate model, first the white-box model gets separated into two parts, which are the (compare with Figure 2.1)

- collector head, identified/known one-mass oscillator, **linear subsystem**, and the
- lower framework, **nonlinear subsystem**.

In the next step the lower framework (nonlinear) part is surrogated by a two-mass oscillator with unknown parameters. The torque of the pneumatic actuator is applied via an additional bar, resulting in an additional force that acts on the lower of the two masses. By coupling this two oscillators and bounding one end two a wall, the three-mass oscillator surrogate model for the pantograph is received, which is a linear system with 7 free parameters moving in a single dimension.

With the orientations of the forces defined in the free body diagram of the three-mass oscillator given in Figure 3.14, the following relations hold:

$$F_p = k_E(\xi - \eta) \tag{3.14}$$

$$F_H = k_C(\zeta - \xi) + c_C(\dot{\zeta} - \dot{\xi}) \tag{3.15}$$

$$F_L = k_L(\delta_M - \zeta) + c_L(\dot{\delta}_M - \dot{\zeta}) \tag{3.16}$$

$$F_W = k_W(-\delta_M) + c_W(-\dot{\delta}_M) \tag{3.17}$$

Hence the **center of mass theorem** for the 3 masses can be denoted by the following equations of motion (compare with the free body diagram of the three-mass oscillator in Figure 3.14):

collector head, mass m_C:

$$\ddot{\xi}\ m_C = F_H - F_p$$

$$\ddot{\xi} = -\frac{k_C}{m_C}\ \xi - \frac{c_C}{m_C}\ \dot{\xi} + \frac{k_C}{m_C}\ \zeta + \frac{c_C}{m_C}\ \dot{\zeta} - \frac{1}{m_C}\ F_p \tag{3.18}$$

mass m_H:

$$\ddot{\zeta}\ m_H = F_L - F_H$$

$$\ddot{\zeta} = \frac{k_C}{m_H}\ \xi + \frac{c_C}{m_H}\ \dot{\xi} - \frac{k_C + k_L}{m_H}\ \zeta - \frac{c_C + c_L}{m_H}\ \dot{\zeta} + \frac{k_L}{m_H}\ \delta_M + \frac{c_L}{m_H}\ \dot{\delta}_M \tag{3.19}$$

mass m_M:

$$\ddot{\delta}_M\ m_M = F_W - F_L + \frac{1}{r_M}\ M_{pa}$$

$$\ddot{\delta}_M = \frac{k_L}{m_M}\ \zeta + \frac{c_L}{m_M}\ \dot{\zeta} - \frac{k_L + k_W}{m_M}\ \delta_M - \frac{c_L + c_W}{m_M}\ \dot{\delta}_M + \frac{1}{r_M m_M}\ M_{pa} \tag{3.20}$$

3.4.3 Equations of Motion in State-Space Form

In this section the equations of motion of the three-mass oscillator surrogate model are rewritten in state-space form. The chosen state vector for a single LLM is given by

$$x_{TMO}(t) = \begin{bmatrix} \xi & \dot{\xi} & \zeta & \dot{\zeta} & \delta_M & \dot{\delta}_M \end{bmatrix}^T. \tag{3.21}$$

From the equations of motion (compare equations (3.18), (3.19) and (3.20)) and the definition of the state vector in equation (3.21) the **system matrix $\mathbf{A}_{TMO}$** is given as

$$\mathbf{A}_{TMO} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{(k_E+k_C)}{m_C} & -\frac{c_C}{m_C} & \frac{k_C}{m_C} & \frac{c_C}{m_C} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{k_C}{m_H} & \frac{c_C}{m_H} & -\frac{(k_C+k_L)}{m_H} & -\frac{(c_C+c_L)}{m_H} & \frac{k_L}{m_H} & \frac{c_L}{m_H} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{k_L}{m_M} & \frac{c_L}{m_M} & -\frac{(k_L+k_W)}{m_M} & -\frac{(c_L+c_W)}{m_M} \end{bmatrix}, \tag{3.22}$$

and with the chosen inputs - contact position η and torque M_{pa} - the resulting **input matrix $\mathbf{B}_{TMO}$** is given by

$$\mathbf{B}_{TMO} = \begin{bmatrix} 0 & 0 \\ \frac{k_E}{m_C} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{1}{r_M m_M} \end{bmatrix}. \tag{3.23}$$

With the choice of the **output matrix $\mathbf{C}_{TMO}$** and a **zero matrix $\mathbf{D}_{TMO}$** (no direct feedthrough)

$$\mathbf{C}_{TMO} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.24}$$

$$\mathbf{D}_{TMO} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \tag{3.25}$$

the three-mass oscillator surrogate model state equation and output equation are found by

$$\dot{\mathbf{x}}_{TMO}(t) = \mathbf{A}_{TMO}\ \mathbf{x}_{TMO}(t) + \mathbf{B}_{TMO}\ \mathbf{u}(t), \tag{3.26}$$

$$\hat{\mathbf{y}}_{TMO}(t) = \mathbf{C}_{TMO}\ \mathbf{x}_{TMO}(t) + \mathbf{D}_{TMO}\ \mathbf{u}(t). \tag{3.27}$$

This state-space system is then discretized utilizing the MATLAB command `c2d()` with *zero-order hold* to discrete time with a sampling time of $T_s = 0.001$ [s] $= 1$ [ms] to the form

$$\mathbf{x}_{TMO}(k+1) = \mathbf{A}_{TMO}\ \mathbf{x}_{TMO}(k) + \mathbf{B}_{TMO}\ \mathbf{u}(k), \tag{3.28}$$

$$\hat{\mathbf{y}}_{TMO}(k) = \mathbf{C}_{TMO}\ \mathbf{x}_{TMO}(k) + \mathbf{D}_{TMO}\ \mathbf{u}(k). \tag{3.29}$$

This state-space system can be extended with an affine term, giving it the possibility to balance out offsets. This extension is necessary if signals are to be identified, which do not possess a zero-mean. Therefore the **affine term state-space formulation** is states as:

With the definition of the affine term state vector x_0 as

$$\mathbf{x}_{TMO,0} = \begin{bmatrix} x_{0,1} & x_{0,2} & x_{0,3} & x_{0,4} & x_{0,5} & x_{0,6} \end{bmatrix}^T, \tag{3.30}$$

the affine term vector u_0 as

$$\mathbf{u}_{TMO,0} = \begin{bmatrix} u_{0,1} & u_{0,2} \end{bmatrix}^T \tag{3.31}$$

and the affine term output vector y_0 as

$$\mathbf{y}_{TMO,0} = \begin{bmatrix} y_{0,1} & y_{0,2} & y_{0,3} & y_{0,4} & y_{0,5} & y_{0,6} \end{bmatrix}^T, \tag{3.32}$$

the affine term state-space system and output equation (in discrete time) can be stated as

$$\mathbf{x}_{TMO}(k+1) = \mathbf{A}_{TMO} \; (\mathbf{x}_{TMO}(k) - \mathbf{x}_{TMO,0}) + \mathbf{B}_{TMO} \; (\mathbf{u}(k) - \mathbf{u}_{TMO,0}), \tag{3.33}$$

$$\hat{\mathbf{y}}_{TMO}(k) = \mathbf{C}_{TMO} \; (\mathbf{x}_{TMO}(k) - \mathbf{x}_{TMO,0}) + \mathbf{D}_{TMO} \; (\mathbf{u}(k) - \mathbf{u}_{TMO,0}) + \mathbf{y}_{TMO,0}. \tag{3.34}$$

3.4.4 Parameter Vector

The parameter vector $\boldsymbol{\theta}_i$ for a single LLM for the surrogate model without affine terms is given as

$$\boldsymbol{\theta}_{i,TMO} = \begin{bmatrix} m_{H,i} & m_{M,i} & k_{L,i} & c_{L,i} & r_{M,i} & k_{W,i} & c_{W,i} \end{bmatrix}^T. \tag{3.35}$$

The parameter vector solely for the affine terms of the surrogate model system equations as well as the parameter vector for the surrogate model with affine terms is given by

$$\boldsymbol{\theta}_{i,AT} = \begin{bmatrix} x_{0,1,i} \\ x_{0,2,i} \\ x_{0,3,i} \\ x_{0,4,i} \\ x_{0,5,i} \\ x_{0,6,i} \\ u_{0,1,i} \\ u_{0,2,i} \\ y_{0,1,i} \\ y_{0,2,i} \\ y_{0,3,i} \\ y_{0,4,i} \\ y_{0,5,i} \\ y_{0,6,i} \end{bmatrix}, \boldsymbol{\theta}_{i,\overline{TMO}} = \begin{bmatrix} \boldsymbol{\theta}_{i,TMO} \\ \boldsymbol{\theta}_{i,AT} \end{bmatrix}. \tag{3.36}$$

The Parameter vector θ for a single LLM for the extended surrogate model with free identifiable pan-head without affine terms is given as

$$\boldsymbol{\theta}_{i,ETMO} = \begin{bmatrix} m_H & m_M & k_L & c_L & r_M & k_W & c_W & m_C & k_C & c_C \end{bmatrix}^T. \tag{3.37}$$

3.4.5 Output Error (OE) Optimization

The free parameters of the surrogate model (see equations (3.35) and (3.37)) are optimized for each LLM using the MATLAB function `fmincon()`.This MATLAB function executes an output error optimization with the cost function given by equation (3.38) by varying the parameters between each simulation run and additionally supports the setting of boundaries for each parameter of the parameter vector. Therefore expert knowledge can be incorporated into the optimization process (compare with Figure 3.13) by the following statements.

- All parameters represent physical entities that need to be greater than or equal to zero (no negative mass, spring stiffness or damping).
- From the knowledge of the magnitude of the forces involved in the mechanism, boundaries for the length of the lever arm r_m of the pneumatic actuator torque M_{pa} can be set.
- From the knowledge of the total mass of the pantograph mechanism, boundaries for the two masses of the lower framework m_H and m_M can be set.
- The upper boundaries for the spring stiffness and damping factors are set to sufficient high values, while 0 would indicate that the spring or damper has no effect on the result.

Table 4.2 gives an overview of the set parameter constraints for the MATLAB function `fmincon()` for a certain identification run.

Remark 3.4.1. The typical output error methods exploit the gradient and Hessian matrix of the loss function for the search of the optimal parameters. For the parameter optimization of the pantograph LLMN (surrogate) these tasks (gradient, Hessian) are carried out by the Matlab function `fmincon()` internally. Just the cost function is handed over.

The employed cost function or **optimization criterion** for the output error (OE) optimization method is given in the form

$$\mathcal{J} = \min_{e} \|\mathbf{Y}_{data} - \hat{\mathbf{Y}}\|_2^2 \tag{3.38}$$

with $\mathbf{Y}_{data}$ and $\hat{\mathbf{Y}}$ containing the variables that are being used for optimization utilizing the Euclidean norm $\|\cdot\|_2$. Additionaly the cost function can be tuned by applying weights to those signals that are more important (e.g. contact force Q_{F_p}) or to balance out different magnitudes of variables (e.g position in 10^0 [m] and forces in 10^3 [N], see weights Q_i). The optimization criterion used in most cases during the development of the pantograph LLMN and which was used for the results shown in Section 4.4 is given by equation (3.39).

$$\begin{aligned}
\mathcal{J}_{TMO} = Q_p \left(\min_e \|\xi_{data} - \hat{\xi}\|_2^2 + \min_e \|\zeta_{data} - \hat{\zeta}\|_2^2 \right) \\
+ Q_v \left(\min_e \|\dot{\xi}_{data} - \hat{\dot{\xi}}\|_2^2 + \min_e \|\dot{\zeta}_{data} - \hat{\dot{\zeta}}\|_2^2 \right) \\
+ Q_f \left(\min_e \|F_{H,data} - \hat{F}_H\|_2^2 + Q_{F_p} (\min_e \|F_{p,data} - \hat{F}_H\|_2^2) \right)
\end{aligned} \tag{3.39}$$

with the weighting factors Q_i set as the mean of the standard deviations of the respective type of signals with $i = p$ ... positions, v ... velocities, f ... forces. Q_{F_p} is an additional weight to lay the focus of the optimization on the contact force signal.

3.5 Subspace Identification Methods

This section is devoted to give some insight into the functionality of subspace identification methods. The applied *numerical algorithm for subspace identification* (N4SID) [2] [33] as proposed by Overschee and Moor in [33] is implemented in the MathWorks Matlab System Identification Toolbox (see e.g. the according User's Guide by Lennart Ljung for further information) as `n4sid()` and utilized for the **pantograph LLMN (n4sid)**.

Beforehand it is mentioned, that the N4SID represents a superior structure implementing all previous developed subspace identification methods. The Matlab function `n4sid()` as it is currently implemented is based on a *unifying theorem* proposed in [34] which allows the interpretation of different subspace identification methods as singular value decompositions of a weighted matrix. These are the Canonical Variate Analysis (CVA) method (see [16]), the group of MOESP methods (see [38], [36] and [37]) and the N4SID (see [33]). Thus throughout this thesis the term N4SID has to be understood as a generic term or umbrella term.

The methods described here, are open-loop identification methods, suitable for the pantograph modeling problem as it is treated in this thesis. In Section 3.5.4 a brief discussion of issues arising with closed-loop subspace identification is carried out.

3.5.1 Introduction to Subspace Identification Methods

This subsection aims to elucidate the used terminology and discuss some basic knowledge that is helpful when dealing with subspace identification methods.

[2] N4SID - "Numerical algorithms for Subspace State Space Identification. Read as a Californian license plate: *enforce id.*"[33]

column space of $\mathbf{A} \in \mathbb{R}^{n \times m}$ $range(\mathbf{A}) = y \in \mathbb{R}^m : y = \mathbf{A}x$ for some $x \in \mathbb{R}^n$	Orthogonal complements in $\mathbb{R}^m$
left null space of $\mathbf{A} \in \mathbb{R}^{n \times m}$ $ker(\mathbf{A}^T) = y \in \mathbb{R}^m : \mathbf{A}^T y = 0$	
row space of $\mathbf{A} \in \mathbb{R}^{n \times m}$ $range(\mathbf{A}^T) = x \in \mathbb{R}^n : x = \mathbf{A}^T y$ for some $y \in \mathbb{R}^m$	Orthogonal complements in $\mathbb{R}^n$
left null space of $\mathbf{A} \in \mathbb{R}^{n \times m}$ $ker(\mathbf{A}) = x \in \mathbb{R}^n : \mathbf{A}x = 0$	

Table 3.4: The four fundamental subspaces of a matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$.

Subspaces of a Matrix and Linear Least-Squares

In general it can be stated, that a matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$ defines a linear transformation from the vector space $\mathbb{R}^n$ to the vector space $\mathbb{R}^m$. Each of these two vector spaces consists of two subspaces, thus defining the **four fundamental subspaces** related to a matrix $A \in \mathbb{R}^{n \times m}$. Table 3.4 summarizes the mathematical definitions of the subspaces of a matrix.

By keeping the relation of subspaces of a matrix in mind, let's take a look at the set of linear equations

$$\mathbf{A}\mathbf{x} = \mathbf{y}. \tag{3.40}$$

Geometrically speaking one seeks the linear combinations of the columns of the matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$ of rank r that equal the vector $\mathbf{y} \in \mathbb{R}^m$, therefore a solution $\mathbf{x} \in \mathbb{R}^n$ only exists provided that the vector $\mathbf{y}$ lies in the column space of the matrix $\mathbf{A}$. If this condition is fulfilled, the set of linear equations 3.40 are called consistent, otherwise inconsistent. To solve an inconsistent set of linear equations, the **linear least-squares** problem can be utilized (compare [39, page 28, Section 2.6]):

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_F^2, \tag{3.41}$$

with $\|\cdot\|_F$ as the Frobenius norm, which in this case ($\mathbf{y}$ defined as a vector) is identical to the Euclidean norm.

Figure 3.15 illustrates the linear-least squares method, where, as described above, linear combinations of the columns of the matrix $\mathbf{A}$ are sought, that determine the vector $\hat{\mathbf{y}}$ with a minimized residual $\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}}$. Or with other words a vector $\mathbf{x}$ is sought, that minimizes the residual $\mathbf{e}$. In the depicted case the vector $\mathbf{y}$ lies outside the column space (plane spanned by the basis vectors $\mathbf{v}_{\mathbf{A},1}$ and $\mathbf{v}_{\mathbf{A},2}$) of the matrix

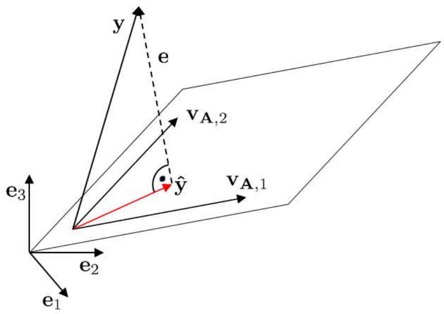


Figure 3.15: Illustration of the solution to the linear least-sqaures problem. The two basis vectors $\mathbf{v}_{\mathbf{A},1}$ and $\mathbf{v}_{\mathbf{A},2}$ spanning a two-dimensional plane in the three-dimensional Euclidean space ($\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{e}_1$), representing the column space of the matrix $\mathbf{A}$. The estimated $\hat{\mathbf{y}}$ is found as the orthogonal projection of the true $\mathbf{y}$ with the minimized residual $\mathbf{e}$ lying in the orthogonal space of the column space of $\mathbf{A}$ (compare with [39, page 30, Fig. 2.1]).

$\mathbf{A}$ and the minimal residual $\mathbf{e}$ is found lying in the orthogonal space of the column space of matrix $\mathbf{A}$. The result of applying this procedure can also be understood as a **projection** of the vector $\mathbf{y}$ onto the plane given by the basis vectors of the column space of matrix $\mathbf{A}$.
The solution $\hat{\mathbf{x}}$ to the least-squares problem (3.41) can be found by the so-called normal equations (compare [39, page 29, equation (2.10)])

$$\mathbf{A}^T\mathbf{A}\hat{\mathbf{x}} = \mathbf{A}^T\mathbf{y}, \tag{3.42}$$

where $\hat{\mathbf{x}}$ is *unique* if the matrix $\mathbf{A}$ has full column rank n. Then $\mathbf{A}^T\mathbf{A}$ is square and invertible and the estimate is found as (compare [39, page 32, equation (2.12)])

$$\hat{\mathbf{x}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y}. \tag{3.43}$$

The matrix $(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$ is referred to as the *pseudo-inverse* of the matrix $\mathbf{A}$. Furthermore the matrix $\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T \in \mathbb{R}^{m\times m}$ yields the *orthogonal projection* of a vector in $\mathbb{R}^m$ onto the space spanned by the columns of the matrix $\mathbf{A}$. This projection is denoted by (compare [39, page 32, Section 2.6.1])

$$\mathbf{\Pi}_{\mathbf{A}} \coloneqq \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T, \tag{3.44}$$

with the properties given in [39, page 32, Section 2.6.1].

Singular Value Decomposition

The subspaces of a matrix can be (numerically efficient) determined by the utilization of a singular value decomposition (SVD) (see e.g. [39, page 26, Theorem 2.6]) and an appropriate partitioning. In general every matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ can be decomposed as (compare [39, page 26, Theorem 2.6])

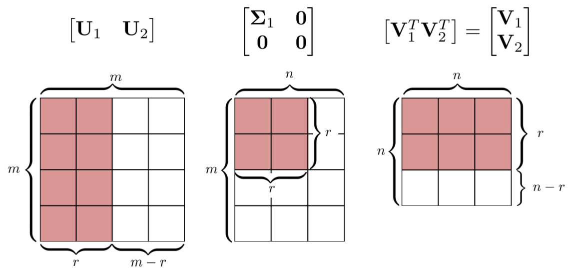


Figure 3.16: Illustration of the partitioning of the SVD of a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $m > n > r = rank(\mathbf{A})$ and $\mathbf{U}_1 \in \mathbb{R}^{m \times r}$, $\mathbf{U}_2 \in \mathbb{R}^{m \times (m-r)}$, $\mathbf{\Sigma}_1 \in \mathbb{R}^{r \times r}$, $\mathbf{V}_1 \in \mathbb{R}^{n \times r}$ and $\mathbf{V}_1 \in \mathbb{R}^{n \times (n-r)}$, where $m = 4 > n = 3 > r = 2$ (compare with equation (3.46)).

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \tag{3.45}$$

with the orthogonal matrices $\mathbf{U} \in \mathbb{R}^{m \times m}$ and $\mathbf{V} \in \mathbb{R}^{n \times n}$ and the matrix $\mathbf{\Sigma} \in \mathbb{R}^{m \times n}$ with the singular values on its main diagonal listed in descending order ($\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_k = 0$ where $rank(\mathbf{A} = r)$ and $min(m.n) = k$). The singular values can be utilized to determine the rank of a matrix, which is implemented in the MATLAB System Identification Toolbox function `n4sid()` for automatic determination of the order of the identified system. This is realized by counting the nonzero singular values the system order n is obtained, in case where no noise is present. If however noise is present on the system all singular values will differ from zero, but a gap between the nth and $(n + 1)$th singular value should be detectable (see [39, page 311, Figure 9.4]).

If a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ has rank r, where $r < m$ and $r < n$ the SVD from equation (3.45) can be partitioned to (compare [39, page 27, Section 2.5])

$$\mathbf{A} = \begin{bmatrix} \mathbf{U}_1 & \mathbf{U}_2 \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^T \mathbf{V}_2^T \end{bmatrix}, \tag{3.46}$$

with $\mathbf{U}_1 \in \mathbb{R}^{m \times r}$, $\mathbf{U}_2 \in \mathbb{R}^{m \times (m-r)}$, $\mathbf{\Sigma}_1 \in \mathbb{R}^{r \times r}$, $\mathbf{V}_1 \in \mathbb{R}^{n \times r}$ and $\mathbf{V}_1 \in \mathbb{R}^{n \times (n-r)}$. This partitioning of the SVD is illustrated for a matrix $\mathbf{A} \in \mathbb{R}^4$ with $r = 2$ in Figure 3.16. From equation (3.46), Figure 3.16 and Table 3.4 it can be seen, that the columns of the received matrices $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}_1$ and $\mathbf{V}_2$ provide orthogonal bases for all four fundamental subspaces of a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $m > n > r = rank(\mathbf{A})$. These results are collected in Table 3.5.

QR Factorization

The QR factorization is utilized by the subspace identification methods for improving the numerical efficiency when estimating state-space system matrices from input and output data.

column space of $\mathbf{A}$:	$range(\mathbf{A}) = range(\mathbf{U}_1)$
left null space of $\mathbf{A}$:	$ker(\mathbf{A}^T) = range(\mathbf{U}_2)$
row space of $\mathbf{A}$:	$range(\mathbf{A}^T) = range(\mathbf{V}_1)$
null space of $\mathbf{A}$:	$ker(\mathbf{A}) = range(\mathbf{V}_2)$

Table 3.5: The resulting matrices $\mathbf{U}_1$, $\mathbf{U}_2$, $\mathbf{V}_1$ and $\mathbf{V}_2$ of the partitioning of the SVD of a matrix $\mathbf{A}$ deliver the four fundamental subspaces of this matrix (compare with Table 3.4, equation (3.46) and Figure 3.16).

column space of $\mathbf{A}$:	$range(\mathbf{A}) = range(\mathbf{Q}_1)$
left null space of $\mathbf{A}$:	$ker(\mathbf{A}^T) = range(\mathbf{Q}_2)$
row space of $\mathbf{A}$:	$range(\mathbf{A}^T) = range(\mathbf{R}_1^T)$

Table 3.6: The resulting matrices $\mathbf{Q}_1$, $\mathbf{Q}_2$ and $\mathbf{R}_1$ of the partitioning of the QR factorization of a matrix $\mathbf{A}$ deliver again some of the fundamental subspaces of this matrix (compare with Table 3.4, equation (3.48) and Section 3.5.1.

In general every matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $m > n$ can be decomposed into (compare [39, page 27, Theorem 2.6])

$$\mathbf{A} = \mathbf{QR}, \tag{3.47}$$

with the orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{m\times m}$ and the augmented upper-triangular matrix $\mathbf{R} \in \mathbb{R}^{m\times n}$ (zero rows at the bottom). This procedure is referred to as *QR factorization* (compare [39, page 27, Theorem 2.7]). As above for a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $m > n > r = rank(\mathbf{A})$ a partitioning of the QR factorization of matrix $\mathbf{A}$ in equation (3.47) can be carried out in the form (compare [39, page 27, Section 2.5])

$$\mathbf{A} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 & \mathbf{R}_2 \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{3.48}$$

with $\mathbf{Q}_1 \in \mathbb{R}^{m\times r}$, $\mathbf{Q}_2 \in \mathbb{R}^{m\times(m-r)}$, $\mathbf{R}_1 \in \mathbb{R}^{r\times r}$ and $\mathbf{R}_2 \in \mathbb{R}^{r\times(n-r)}$. In analogy to the illustrations presented in the previous section the matrices received by the partitioned QR-decomposition can be utilized to deliver some subspaces of the matrix $\mathbf{A}$, as is summarized in Table 3.6.

Remark 3.5.1. The RQ factorization of a matrix $\mathbf{A}$ is related to the QR-decomposition of the matrix $\mathbf{A}^T$ as is shown in [39, page 28, Section 2.5].

Kalman-Filter Problem

In general the Kalman-filter represents a computational scheme for reconstructing the state vector $\mathbf{x}(k)$ of a given state-space model in a statistically optimal manner. The well known Kalman-filter problem (see e.g. [39, page 134, Section 5.3]) is a minimum-error variance estimation problem (compare defintions [39, page 110, Definition 4.15] and [39, page 110, Definition 4.16]). The Kalman-filter is also referred to as the optimal-statistical-state-observer and is a member of the class of filters used to reconstruct missing information, such as part of the state vector from measured quantities in a state-space model.

In relation to the subspace identification methods, it can be stated that an observer is a filter that approximates the state vector of a dynamical system from measurements of the input and output sequences and in general requires a model of the system under consideration (formulation of the Kalman filter problem). Furthermore by formulating the Kalman filter problem in a recursive manner (keywords: one-step-ahead predicted states, time update, measurement update; compare [39, page 135, Section 5.4]), it can be interpreted as a stochastic least-squares problem. Thereby a connection between the Kalman filter problem, a least-squares problem and, as described above, a projection (into a subspace) is found. This mutually interpretation possibilities are utilized in deriving the N4SID equations as described in [33] and Section 3.5.2. For further information see [33, page 9, Chapter 4].
It is also mentioned here, that the introduction to the functionality of the subspace methods done in Section 3.5.2 is in regard to classical subspace identification, where the system matrices of the data generating system are identified. The N4SID approach however differs in its implementation, by first estimating the Kalman states of the underlying system, and subsequently determining the system matrices (see illustration in [33, page 47, Figure 1] and remarks throughout Section 3.5.2).

State-Space System Representations

The subspace identification methods are based on different forms of state-space systems, which will be briefly discussed here.

In general the stochastic (state-space) model of interest for the identification of the pantograph is given in **process form** by

$$\hat{\mathbf{x}}(k+1) = \mathbf{A}(k)\mathbf{x}(k) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{w}(k), \tag{3.49}$$

$$\mathbf{y}(k) = \mathbf{C}(k)\mathbf{x}(k) + \mathbf{D}(k)\mathbf{u}(k) + \mathbf{v}(k), \tag{3.50}$$

with the matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{q\times n}$, $\mathbf{D} \in \mathbb{R}^{q\times m}$, the state vector $\mathbf{x} \in \mathbb{R}^{n}$, the input vector $\mathbf{u} \in \mathbb{R}^{m}$, the output vector $\mathbf{y} \in \mathbb{R}^{q}$, the process noise $\mathbf{w} \in \mathbb{R}^{n}$ and the measurement noise $\mathbf{v} \in \mathbb{R}^{q}$ - which both have to be white and uncorrelated sequences - and with n the number of states, m the number of inputs and q the number of outputs. For this system (has to be observable) a Kalman filter can be designed using a stochastic input denoted as a innovation sequence by

$$\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{C}(k)\hat{\mathbf{x}}(k|k-1), \tag{3.51}$$

which is a zero-mean white-noise sequence and independent of past input and output data (for further information regarding the stochastic properties see [39, page 152, Section 5.5.5]). The innovation represents the difference between the observed value and the predicted value, therefore implementing a predictive observer based disturbance model. Together with the Kalman gain $\mathbf{K}$ (which can be obtained by solving an algebraic Ricatti equation) the state-update and output equations can be written as

$$\hat{\mathbf{x}}(k+1|k) = \mathbf{A}(k)\hat{\mathbf{x}}(k|k-1) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{K}(k)\mathbf{e}(k), \tag{3.52}$$

$$\mathbf{y}(k) = \mathbf{C}(k)\hat{\mathbf{x}}(k|k-1) + \mathbf{e}(k) \tag{3.53}$$

which is referred to as the innovation representation of a state-space system or state-space system in **innovation form**.

3.5.2 Subspace Identification

This section treats the approach of identifying linear time-invariant (LTI) state-space models from input and output data of a dynamic system via subspace identification methods. The content of the following sections aims to give the reader a basic understanding of functionality of these methods and an overview of the common approaches. For further information on the presented methods and mathematical proofs consult [39, Chapter 9], for the derivation of the n4sid equations see [33] as well as [35]. Furthermore, as mentioned in the introduction of this Section, there exists a *unifying theorem* (see [34]) showing the similarities of several subspace identification methods. An overview of the state of the art subspace identification methods (for open-loop and closed-loop systems) can be found in [25].

The general conception is to store the available input and output data in structured block Hankel matrices with which the so-called data equation (see equation (3.60)) or a Kalman filter can be formulated. Then, by solving a number of simple linear-algebra problems (SVD and QR factorization and solution of a linear least-squares problem), it is possible to retrieve certain information of the underlying state-space system (which generated the data) either by the column space of the observability matrix of the data equation or the row space of the Kalman filter in a non-iterative fashion (no nonlinear optimization required). The bottom line is, that the system matrices of a LTI system can be retrieved up to a similarity transformation solely from input and output data, which can be utilized to generate a pantograph LLMN (n4sid) without any knowledge about the pantograph under investigation (see Section 3.5.4 for a discussion of issues that arise when implementing this approach).

The subspace identification will be explained starting with deterministic (i.e. noise-free) systems, moving on gradually ending up with models where white process and measurement noise is present.

The state-update and output equation of a minimal (reachable and observable) deterministic system are given by

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \tag{3.54}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k), \tag{3.55}$$

with $\mathbf{x}(k) \in \mathbb{R}^n$, $\mathbf{u}(k) \in \mathbb{R}^m$ and $\mathbf{y}(k) \in \mathbb{R}^l$. For the subspace ID the unknowns of that equations are the system matrices ($\mathbf{A}, \mathbf{B}, \mathbf{C}$ and $\mathbf{D}$) and the initial state vector $\mathbf{x}(k=0)$.

By inserting the the state-update equations of all time steps k from $k = 0$ up to $k = s-1$ with $s > n$ into each other, the output equations of these time steps can be written in the following formulation, giving a relationship between the input data batch and the initial vector to the output data batch as (compare [39, page 295, equation (9.4)])

$$\begin{bmatrix} \mathbf{y}(0) \\ \mathbf{y}(1) \\ \mathbf{y}(2) \\ \vdots \\ \mathbf{y}(s-1) \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{s-1} \end{bmatrix}}_{\mathcal{O}_s} \mathbf{x}(0) + \underbrace{\begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & & \ddots & \ddots & \vdots \\ \mathbf{CA}^{s-2}\mathbf{B} & \mathbf{CA}^{s-3}\mathbf{B} & \cdots & \mathbf{CB} & \mathbf{D} \end{bmatrix}}_{\mathcal{T}_s} \begin{bmatrix} \mathbf{u}(0) \\ \mathbf{u}(1) \\ \mathbf{u}(2) \\ \vdots \\ \mathbf{u}(s-1) \end{bmatrix}, \tag{3.56}$$

where $\mathcal{O}_s$ is referred to as the **extended observability matrix** (compare [33, page 4, Section 2.1.1]).

As the underlying system is assumed to be time-invariant arbitrary time-shifts can be employed to equation (3.56), while keeping the same matrices $\mathcal{O}_s$ and $\mathcal{T}_s$, e.g. for a shift over k samples (compare [39, page 295, equation (9.5)]):

$$\begin{bmatrix} \mathbf{y}(k) \\ \mathbf{y}(k+1) \\ \vdots \\ \mathbf{y}(k+s-1) \end{bmatrix} = \mathcal{O}_s \mathbf{x}(k) + \mathcal{T}_s \begin{bmatrix} \mathbf{u}(k) \\ \mathbf{u}(k+1) \\ \vdots \\ \mathbf{u}(k+s-1) \end{bmatrix}. \tag{3.57}$$

If now equations (3.56) and (3.57) are combined for different time-shifts (dependent on the availability of according data) the data equation can be defined as (compare [39, page 296, equation (9.5)])

$$
\begin{bmatrix}
\mathbf{y}(0) & \mathbf{y}(1) & \cdots & \mathbf{y}(N-1) \\
\mathbf{y}(1) & \mathbf{y}(2) & \cdots & \mathbf{y}(N) \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{y}(s-1) & \mathbf{y}(s) & \cdots & \mathbf{y}(N+s-2)
\end{bmatrix} = \\
= \mathcal{O}_s \mathbf{X}_{0,N} + \mathcal{T}_s
\begin{bmatrix}
\mathbf{u}(0) & \mathbf{u}(1) & \cdots & \mathbf{u}(N-1) \\
\mathbf{u}(1) & \mathbf{u}(2) & \cdots & \mathbf{u}(N) \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{u}(s-1) & \mathbf{u}(s) & \cdots & \mathbf{u}(N+s-2)
\end{bmatrix}, \tag{3.58}
$$

where (compare [39, page 296, Section 9.2.1)])

$$
\begin{aligned}
\mathbf{X}_{i,N} &= \begin{bmatrix} \mathbf{x}(i) & \mathbf{x}(i+1) & \cdots & \mathbf{x}(i+N-1) \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{A}^i \mathbf{x}(i) & \mathbf{A}^{i+1} \mathbf{x}(i) & \cdots & \mathbf{A}^{i+N-1} \mathbf{x}(i) \end{bmatrix}
\end{aligned} \tag{3.59}
$$

with $n < s < N$.

Remark 3.5.2 (Equation (3.59)). $\mathbf{X}_{i,N}$ as denoted in equation (3.59) only depends on the initial state $\mathbf{x}(0)$ and the system matrix $\mathbf{A}$ if i is set to 0.

The **data equation** can be denoted in compact form as (compare [39, page 296, equation (9.7)])

$$
\mathbf{Y}_{0,s,N} = \mathcal{O}_s \mathbf{X}_{0,N} + \mathcal{T}_s \mathbf{U}_{0,s,N}, \tag{3.60}
$$

with the known matrices $\mathbf{Y}_{0,s,N}$ and $\mathbf{U}_{0,s,N}$ as block Hankel matrices containing the input and output data, the unknown matrices $\mathcal{O}_s$ as the extended observability matrix and the lower block triangular Toeplitz matrix $\mathcal{T}_s$ (compare [33, page 5, Section 2.1.1]) containing the state-space matrices of the underlying system up to a similarity transformation, as well as the unknown initial condition $\mathbf{x}(0)$ contained in the state matrix $\mathbf{X}_{0,N}$.

Subspace Identification for Autonomous Systems

By looking at an autonomous system (special case of the deterministic system) the basic operations of the subspace identification methods can be demonstrated. The data equation (3.60) becomes then (matrices $\mathbf{B}$ and $\mathbf{D}$ equal zero) (compare [39, page 297, equation (9.8)])

$$
\mathbf{Y}_{0,s,N} = \mathcal{O}_s \mathbf{X}_{0,N}. \tag{3.61}
$$

Equation (3.61) shows, that the columns of the block Hankel matrix of the output data $\mathbf{Y}_{0,s,N}$ (as defined in equation (3.58)) are linear combinations of the columns of the extended observability matrix $\mathcal{O}_s$ (as defined in equation (3.56)). To see this compare with Section 3.5.1 equation (3.40). If additionally $N \geq s > n$ it can be shown that under some mild conditions ($\mathbf{X}_{0,N}$ full row rank n, minimal system, application of Sylvester inequality, compare [39, page 297, Section 9.2.2]) the column spaces of $\mathbf{Y}_{0,s,N}$ and $\mathcal{O}_s$ are equal. Hence by applying a SVD (see Section 3.5.1 equation (3.46)) to the matrix $\mathbf{Y}0, s, N$ the system matrices $\mathbf{A}$ and $\mathbf{C}$ can be determined up to a similarity transformation (compare [39, page 298, equation (9.9)]):

$$\mathbf{Y}_{0,s,N} = \mathbf{U}_n \mathbf{\Sigma}_n \mathbf{V}_n^T, \tag{3.62}$$

with $\mathbf{\Sigma}_n \in \mathbb{R}^{n \times n}$ with $rank(\mathbf{\Sigma}_n) = n$.

The matrix $\mathbf{C}_T = \mathbf{CT}$ (similarity transformation of matrix $\mathbf{C}$) can now be received from (compare [39, page 298, Section 9.2.2])

$$\mathbf{U}_n = \mathcal{O}_s \mathbf{T} = \begin{bmatrix} \mathbf{CT} \\ \mathbf{CT}(\mathbf{T}^{-1}\mathbf{AT}) \\ \vdots \\ \mathbf{CT}(\mathbf{T}^{-1}\mathbf{AT})^{s-1} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_T \\ \mathbf{C}_T\mathbf{A}_T \\ \vdots \\ \mathbf{C}_T\mathbf{A}_T^{s-1} \end{bmatrix} \tag{3.63}$$

by taking the first l rows of $\mathbf{U}_n$ i.e. (compare [39, page 299, Section 9.2.2])

$$\mathbf{C}_T = \mathbf{U}_n(1:l, :), \tag{3.64}$$

with l as the number of outputs as defined in equation (3.55). Furthermore the matrix $\mathbf{A}_T$ can be computed from the equality resulting from equation (3.63) as (compare [39, page 299, equation (9.10)])

$$\mathbf{U}_n(1:(s-1)l, :)\ \mathbf{A}_T = \mathbf{U}_n(l+1, sl, :), \tag{3.65}$$

which due to $s > n$ has a unique solution. Additionally the number of **nonzero singular values determines the order of the underlying system**, which becomes more relevant in the presence of noise disturbance as will be shown later on.

Figure 3.17 depicts two block Hankel matrices of the output data of a state-space system with $n = 2$ states and $N >> s = 3 > n$. Therefore, each column of the block Hankel matrix (see e.g. equation (3.58)) can be represented by a point in the three dimensional space, where the two trajectories represent output data sets of simulations realized with different initial states $\mathbf{x}(0)$. This illustration of the column space of the block Hankel matrix allows a **geometric interpretation** due to the fact that both curves lie in the same two-dimensional subspace (a plane). This plane therefore has to be characteristic for the matrix pair $(\mathbf{A}, \mathbf{C})$. A different autonomous state-space system would deliver state trajectories that lie in a different plane, therefore this visualization shows how output data (i.e. the column space of the block Hankel matrix) contains information about the underlying system (state-dimension n).

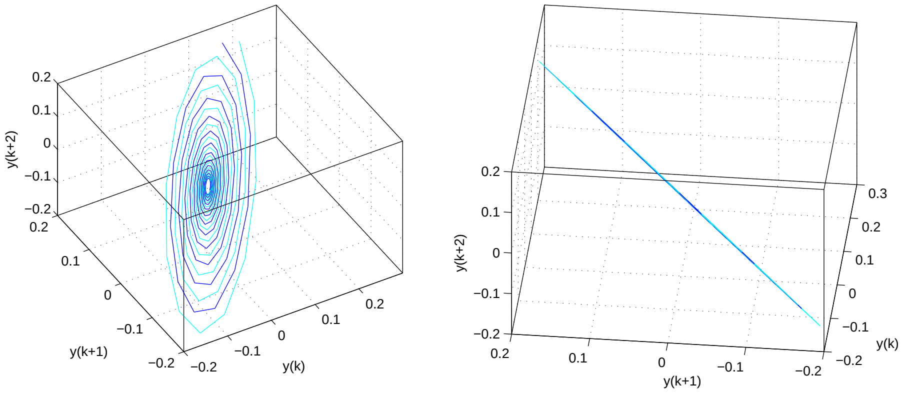


Figure 3.17: Illustration of two block Hankel matrices containing the output data of the same autonomous state-space system for two different initial values $x(0)$ (blue and cyan). The state trajectories are lying in a two-dimensional subspace of a three-dimensional ambient space, therefore revealing information of the underlying system.

Subspace Identification for General Input Sequences

As shown in the previous section, the subspace identification aims to retrieve a matrix whose column space is equal to the column space of the extended observability matrix to subsequently determine the system matrices $\mathbf{A}$ and $\mathbf{C}$ of the underlying system up to a similarity transformation. In case of the presence of a general input to the system, this approach would suggest subtracting the term $\mathcal{T}_s \mathbf{U}_{0,s,N}$ from $\mathbf{Y}_{0,s,N}$ in equation (3.60) and perform a SVD to determine $\mathbf{A}$ and $\mathbf{B}$ as shown in equations (3.64) and (3.65). However the matrix $\mathcal{T}_s$ is unknown and therefore has to be estimated. This can be done by formulating an according linear least-squares problem (compare [39, page 301, Section 9.2.4])

$$\min_{\mathcal{T}_s} \|\mathbf{Y}_{0,s,N} - \mathcal{T}_s \mathbf{U}_{0,s,N}\|_F^2, \tag{3.66}$$

which can be reformulated in analogy to equation (3.44) as the orthogonal projection of the block Hankel matrix $\mathbf{Y}_{0,s,N}$ onto the column space of the block Hankel matrix $\mathbf{U}_{0,s,N}$ utilizing a projection matrix. Thus the influence of the input on the output can be removed in the data equation (3.60) which then can be written as (compare [39, page 302, equation (9.17)])

$$\mathbf{Y}_{0,s,N} \mathbf{\Pi}^{\perp}_{U_{0,s,N}} = \mathcal{O}_s \mathbf{X}_{0,N} \mathbf{\Pi}^{\perp}_{U_{0,s,N}}, \tag{3.67}$$

with the

orthogonal projection matrix (compare [39, page 302, equation (9.16)] and equation (3.44))

$$\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}} = \mathbf{I}_N - \mathbf{U}^T_{0,s,N}(\mathbf{U}_{0,s,N}\mathbf{U}^T_{0,s,N})^{-1}\mathbf{U}_{0,s,N}, \tag{3.68}$$

with the properties (compare [39, page 295, Section 9.2.4])

$$\mathbf{U}_{0,s,N}\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}} = \mathbf{0}, \tag{3.69}$$

$$\text{rank}\left(\mathbf{Y}_{0,s,N}\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}}\right) = n. \tag{3.70}$$

The subspace identification can only be carried out if the input $\mathbf{u}(k)$ is such that (compare with [39, page 302, Lemma 9.1], based on the definition of input sequences that are *persistently exiting* of order n in [39, page 358, Definition 10.1])

$$rank\left(\left[\mathbf{X}_{0,N}\mathbf{U}_{0,s,N}\right]\right) = n + sm, \tag{3.71}$$

Only then the column space of $\mathbf{Y}_{0,s,N}\mathbf{\Pi}_U$ in equation (3.67) is contained in the column space of the extended observability matrix and therefore the system matrices $\mathbf{A}_T$ and $\mathbf{C}_T$ can be retrieved (compare [39, page 303, equation (9.19)]):

$$\text{range}\left(\mathbf{Y}_{0,s,N}\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}}\right) = \text{range}\left(\mathcal{O}_s\right). \tag{3.72}$$

As it can be shown, the numerical efficiency of the subspace identification computation can be significantly reduced by using the following RQ factorization (see Lemma [39, page 304, Lemma 9.2] and Theorem [39, page 305, Theorem 9.1]):

$$\begin{bmatrix}\mathbf{U}_{0,s,N}\\ \mathbf{Y}_{0,s,N}\end{bmatrix} = \begin{bmatrix}\mathbf{R}_{11} & \mathbf{0} & \mathbf{0}\\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{Q}_1\\ \mathbf{Q}_2\\ \mathbf{Q}_3\end{bmatrix}, \tag{3.73}$$

with $\mathbf{R}_{21} \in \mathbb{R}^{sm\times sm}$ and $\mathbf{R}_{22} \in \mathbb{R}^{sl\times sl}$, $\mathbf{Q}_2 \in \mathbb{R}^{sl\times N}$. Applying this factorization simplifies (computationally) the relation of the column spaces to (compare [39, page 305, equation (9.22)])

$$\text{range}\left(\mathbf{Y}_{0,s,N}\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}}\right) = \text{range}\left(\mathbf{R}_{22}\right) = \text{range}\left(\mathcal{O}_s\right). \tag{3.74}$$

Applying the SVD to the term $\mathbf{R}_{22}$ retrieves the system matrices $\mathbf{A}_T$ and $\mathbf{C}_T$ as shown above. Thereby the construction of the projection matrix, which is considerably big and involves a matrix inversion computation, can be avoided.

Finally the matrices $\mathbf{B}_T$ and $\mathbf{D}_T$ as well as the initial state vector $\mathbf{x}_T(0)$ can be computed setting up a linear least-squares problem (see [39, page 307, equation (9.25)]) or directly from $\mathbf{R}_{11}$ and $\mathbf{R}_{21}$ by exploiting the structure of the matrix $\mathcal{T}_s$ in equation (3.56). For further information consult [38].

Instrumental Variables

For real world problems, the requirement of a deterministic (noise-free) system is hardly ever fulfilled. Therefore concepts to deal with **noise disturbance** were developed, were in the following sections all noise sequences are assumed to be *ergodic* (strong law of large numbers, see [39, page 104, Section 4.3.4]) stochastic processes. There are three cases of subspace identification utilizing instrumental variables that will be discussed separately:

- subspace identification with measurement noise (white - MOESP, colored - PI-MOESP)
- subspace identification with process and measurement noise (PO-MOESP)
- subspace identification with process and measurement noise based on a least-squares problem (N4SID)

First the system equations (3.54)-(3.55) are reformulated for **measurement noise** by

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \tag{3.75}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) + \mathbf{v}(k). \tag{3.76}$$

The data equation (3.60) for measurement noise is extended to (compare [39, page 307, equation (9.28)])

$$\mathbf{Y}_{0,s,N} = \mathcal{O}_s \mathbf{X}_{0,N} + \mathcal{T}_s \mathbf{U}_{0,s,N} + \mathbf{V}_{0,s,N}, \tag{3.77}$$

with $\mathbf{V}_{0,s,N}$ the block Hankel matrix constructed for the measurement noise sequence $v(k)$.

The aim now is to carry out the subspace identification in the same fashion as shown before, receiving unbiased estimates of the system matrices $\mathbf{A}_T, \mathbf{B}_T, \mathbf{C}_T$ and $\mathbf{D}_T$ in the presence of measurement noise. To be able to do so, the influence of the input $\mathbf{U}_{0,s,N}$ and additionally of the measurement noise $\mathbf{V}_{0,s,N}$ have to be eliminated from equation (3.77). To remove the influence of the block Hankel matrix constructed from the noise sequence $v(k)$ the so called *instrumental variables* matrix (as proposed in [30]) was introduced, which posses the following properties (compare [39, page 314, equations (9.37) and (9.38)]):

Properties of the **instrumental-variables matrix** $\mathbf{Z}_N$:

$$\lim_{N\to\infty} \frac{1}{N} \mathbf{V}_{i,s,N} \mathbf{\Pi}^{\perp}_{\mathbf{U}_{i,s,N}} \mathbf{Z}_N^T = 0, \tag{3.78}$$

$$\operatorname{rank}\left(\lim_{N\to\infty} \frac{1}{N} \mathbf{X}_{i,N} \mathbf{\Pi}^{\perp}_{\mathbf{U}_{i,s,N}} \mathbf{Z}_N^T \right) = n. \tag{3.79}$$

Due to property (3.78) it is possible to again retrieve the system matrices of the underlying system by multiplying the data equation (3.77) on the right first by the projection matrix $\mathbf{\Pi}^{\perp}_{\mathbf{U}_{i,s,N}}$ and then by the instrumental variables matrix $\mathbf{Z}_N^T$, yielding (compare [39, page 314, Section 9.4])

$$\lim_{N\to\infty} \frac{1}{N}\mathbf{Y}_{0,s,N}\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}}\mathbf{Z}_N^T = \lim_{N\to\infty} \frac{1}{N}\mathcal{O}_s\mathbf{X}_{i,N}\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}}\mathbf{Z}_N^T. \tag{3.80}$$

Because of property (3.79), the column space of the term on the left-hand side of equation (3.80) is again contained in the column space of the extended observability matrix $\mathcal{O}_s$ by (compare [39, page 314, equation (9.39)])

$$\text{range}\left(\lim_{N\to\infty} \frac{1}{N}\mathbf{Y}_{0,s,N}\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}}\mathbf{Z}_N^T\right) = \text{range}\left(\mathcal{O}_s\right). \tag{3.81}$$

In the case were measurement noise $\mathbf{v}(k)$ and **process noise** $\mathbf{w}(k)$ are present, the system equations (3.54)-(3.55) have to be reformulated, utilizing the innovation form as purposed in Section 3.5.1 by equations (3.52)-(3.53), yielding the data equation (3.60) extended for measurement and process noise as (compare [39, page 321, equation (9.52)])

$$\mathbf{Y}_{0,s,N} = \mathcal{O}_s\mathbf{X}_{0,N} + \mathcal{T}_s\mathbf{U}_{0,s,N} + \mathcal{S}_s\mathbf{E}_{0,s,N}, \tag{3.82}$$

where $\mathbf{E}_{0,s,N}$ is a block Hankel matrix constructed from the innovation sequence $\mathbf{e}(k)$ and with the weighting matrix (compare with [39, page 322, Section 9.6])

$$\mathcal{S}_s = \begin{bmatrix} \mathbf{I}_l & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CK} & \mathbf{I}_l & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CAK} & \mathbf{CK} & \mathbf{I}_l & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{CA}^{s-2}\mathbf{K} & \mathbf{CA}^{s-3}\mathbf{K} & \cdots & \mathbf{CK} & \mathbf{I}_l \end{bmatrix}. \tag{3.83}$$

Here the innovation sequence $\mathbf{e}(k)$ is a white-noise sequence and $\mathbf{K}$ is the Kalman gain (compare Section 3.5.1). By formulating the data equation in this way, the properties of the instrumental variables matrix are the same as in case of subspace identification with just measurement noise, if $\mathbf{V}_{i,s,N}$ in equation (3.78) gets replaced with $\mathbf{E}_{i,s,N}$.

Finally an alternative route to subspace identification as proposed in [33] for the derviation of the N4SID method will be introduced. Here the instrumental variable matrix $\mathbf{Z}_N$ is utilized to derive the extended observability matrix $\mathcal{O}_s$ by constructing a linear-least squares problem (projection, compare [39, page 330, equation (9.63)] and [33, page 7, Chapter 3])

$$\begin{bmatrix}\hat{\mathbf{L}}_N^u \hat{\mathbf{L}}_N^z\end{bmatrix} = \arg\min_{\mathbf{L}^u,\mathbf{L}^z} \|\mathbf{Y}_{s,s,N} - \begin{bmatrix}\mathbf{L}^u & \mathbf{L}^z\end{bmatrix}\begin{bmatrix}\mathbf{U}_{s,s,N} \\ \mathbf{Z}_N\end{bmatrix}\|_F^2, \tag{3.84}$$

where (compare [39, page 330, equation (9.64)])

$$\lim_{N\to\infty} \hat{\mathbf{L}}_N^z = \mathcal{O}_s\mathcal{L}_s + \mathcal{O}_s(\mathbf{A}-\mathbf{KC})^s\Delta_z \tag{3.85}$$

yields the extended observability matrix $\mathcal{O}_s$ (for details on the involved matrices consult [39]). The data equation (3.60) can be written for this formulation for $i = s$ as (compare [39, page 330, equation (9.67)])

$$\mathbf{Y}_{s,s,N} = \mathcal{O}_s\mathcal{L}_s\mathbf{Z}_N + \mathcal{T}_s\mathbf{U}_{s,s,N} + \mathcal{S}_s\mathbf{E}_{s,s,N} + \mathcal{O}_s(\mathbf{A}-\mathbf{KC})^s\mathbf{X}_{0,N} \tag{3.86}$$

The remaining question that needs to be answered is how to choose the instrumental variable $\mathbf{Z}_N$, which is not unambiguous. Therefore several subspace identification methods were developed, which will be discussed briefly in the upcoming section.

Subspace Identification Methods

Table 3.7 and 3.8 give an overview of common subspace identification methods that incorporate different choices of the instrumental variable $\mathbf{Z}_n$, acoording to the equations introduced in the previous sections.

Remark 3.5.3 (Table 3.8, subspace method N4SID). The range equation of the N4SID method (made computationally efficient by the RQ factorization) can be used to compute the extended observability matrix as shown before. Alternatively the matrix $\mathbf{X}_{s,N}$ contains the state-sequence of a Kalman filter, which can also be estimated by the SVD as

$$\hat{\mathbf{X}}_{s,N} = \mathbf{\Sigma}_n^{\frac{1}{2}}\mathbf{V}_n^T \tag{3.87}$$

according to equation (3.45). Taking the path of estimating the state-sequence of a Kalman filter also allows to determine the state-space matrices $\mathbf{A}_T, \mathbf{B}_T, \mathbf{C}_T$ and $\mathbf{D}_T$ by solving the least-squares problem (compare [39, page 332, equation (9.69)])

$$\min_{\mathbf{A}_T,\mathbf{B}_T,\mathbf{C}_T\mathbf{D}_T} \| \begin{bmatrix} \hat{\mathbf{X}}_{s+1,N} \\ \mathbf{Y}_{s,1,N-1} \end{bmatrix} - \begin{bmatrix} \mathbf{A}_T & \mathbf{B}_T \\ \mathbf{C}_T & \mathbf{D}_T \end{bmatrix} \begin{bmatrix} \hat{\mathbf{X}}_{s,N-1} \\ \mathbf{U}_{s,1,N-1} \end{bmatrix} \|_F^2, \tag{3.88}$$

and was first proposed by [33]. These estimated system matrices $\mathbf{A}_T, \mathbf{B}_T, \mathbf{C}_T$ and $\mathbf{D}_T$ can furthermore be used to estimate the Kalman gain of the innovation form state-space system as defined in equation (3.52) by solving a according Riccati equation (for further details consult [39, page 333, Section 9.6.3]).

Finally it is mentioned, that there exists a possibility to related the different approaches to subspace identification based on the formulation of the least-squares problem in equation (3.84). By solving this problem in an alternative manner, it can be shown (see [39, page 334, Section 9.6.4]) that the extended observability matrix $\mathcal{O}_s$ can be received from the SVD of a weighted matrix given by

$$\mathbf{W}_1\left((\mathbf{Y}_{s,s,N}\mathbf{\Pi}^{\perp}_{\mathbf{U}_{s,s,N}}\mathbf{Z}_N^T)(\mathbf{Z}_N\mathbf{\Pi}^{\perp}_{\mathbf{U}_{0,s,N}}\mathbf{Z}_N^T)^{-1}\right)\mathbf{W}_2 = \mathbf{U}_n\mathbf{\Sigma}_n\mathbf{V}_n^T. \tag{3.89}$$

MOESP	Multivariable Output-Error State-sPace method ([38])
System	given by equations (3.75)-(3.76) resulting in data equation (3.77) with $i = 0$, **white** measurement noise
Conditions	with $\mathbf{v}(k)$ as an ergodic white-noise sequence with variance $\sigma^2\mathbf{I}_l$ that is uncorrelated with the ergodic sequence $\mathbf{u}(k)$ with $\mathbf{u}(k)$ as an ergodic sequence, such that condition (3.71) is satisfied.
$\mathbf{Z}_N$	**no** instrumental variable necessary due to properties of the noise sequence as it is shown in [39, page 307, Section 9.3]
RQ factorization	$\begin{bmatrix} \mathbf{U}_{0,s,n} \\ \mathbf{Y}_{0,s,N} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{0} \\ \mathbf{R}_{21} & \mathbf{R}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix}$
SVD	$\lim_{N\to\infty} \frac{1}{\sqrt{N}}\mathbf{R}_{22} = \begin{bmatrix} \mathbf{U}_n & \mathbf{U}_2 \end{bmatrix} \begin{bmatrix} \sqrt{\mathbf{\Sigma}_n^2 + \sigma^2\mathbf{I}_n} & \mathbf{0} \\ \mathbf{0} & \sigma\mathbf{I}_{sl-n} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^T \\ \mathbf{V}_2^T \end{bmatrix}$ $\text{range}(\mathbf{U}_n) = \text{range}(\mathcal{O}_s)$
PI-MOESP	Past Inputs Multivariable Output-Error State-sPace method ([36])
System	given by equations (3.75)-(3.76) resulting in data equation (3.77) with $i = s$, **colored** measurement noise
Conditions	with $\mathbf{v}(k)$ as an ergodic noise sequence that is uncorrelated with the ergodic sequences $\mathbf{x}(j)$ and $\mathbf{u}(j)$ for all $k, j \in \mathbb{Z}$ with $\mathbf{x}(k)$ and $\mathbf{u}(k)$ such that the rank conditions (compare [39, page 317, equation (9.44)] and [39, page 318, equation (9.45)]) $\text{rank}\left(\lim_{N\to\infty} \frac{1}{N} \begin{bmatrix} \mathbf{X}_{s,N} \\ \mathbf{U}_{s,s,N} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{0,s,N}^T & \mathbf{U}_{s,s,N}^T \end{bmatrix}\right) = n+sm$ and $\lim_{N\to\infty} \frac{1}{N}\mathbf{U}_{0,s,N}\mathbf{U}_{0,s,N}^T$ has full rank, are satisfied
$\mathbf{Z}_N$	the past inputs data block Hankel matrix is used with $\mathbf{Z}_N = \mathbf{U}_{0,s,N}$
RQ factorization	$\begin{bmatrix} \mathbf{U}_{s,s,n} \\ \mathbf{U}_{0,s,n} \\ \mathbf{Y}_{s,s,N} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{0} \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \\ \mathbf{Q}_3 \end{bmatrix}$
SVD	$\text{range}\left(\lim_{N\to\infty} \frac{1}{\sqrt{N}}\mathbf{R}_{32}\right) = \text{range}(\mathcal{O}_s)$

Table 3.7: Subspace Identification Methods with $N \geq s > n$, Part 1.

PO-MOESP	Past Outputs Multivariable Output-Error State-sPace method ([37])
System	given by equations (3.52)-(3.53) resulting in data equation (3.82) with $i = s$, white measurement noise and white process noise
Conditions	with $\mathbf{e}(k)$ as an ergodic white-noise sequence that is uncorrelated with the ergodic sequences $\mathbf{x}(j)$ and $\mathbf{u}(j)$ for all $k, j \in \mathbb{Z}$ with $\mathbf{e}(k)$ and $\mathbf{u}(k)$ such that the rank conditions (compare [39, page 325, equation (9.58)] and [39, page 326, equation (9.59)]) $\operatorname{rank}\left(\lim_{N\to\infty} \frac{1}{N} \begin{bmatrix} \mathbf{X}_{s,N} \\ \mathbf{U}_{s,s,N} \end{bmatrix} \begin{bmatrix} \mathbf{Y}^T_{0,s,N} & \mathbf{U}^T_{0,s,N} & \mathbf{U}^T_{s,s,N} \end{bmatrix}\right) = n + sm$ and $\operatorname{rank}\left(\lim_{N\to\infty} \frac{1}{N} \begin{bmatrix} \mathbf{X}_{0,N} \\ \mathbf{U}_{0,2s,N} \end{bmatrix} \begin{bmatrix} \mathbf{X}^T_{0,N} & \mathbf{U}^T_{0,2s,N} \end{bmatrix}\right) = n + 2sm$ are satisfied
$\mathbf{Z}_N$	the past input and output data block Hankel matrices are used with $\mathbf{Z}_N = \begin{bmatrix} \mathbf{U}_{0,s,N} \mathbf{Y} 0, s, N \end{bmatrix}$
RQ factorization	$\begin{bmatrix} \mathbf{U}_{s,s,n} \\ \begin{bmatrix} \mathbf{U}_{0,s,n} \\ \mathbf{Y}_{0,s,n} \end{bmatrix} \\ \mathbf{Y}_{s,s,N} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{0} \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \\ \mathbf{Q}_3 \end{bmatrix}$
SVD	$\operatorname{range}\left(\lim_{N\to\infty} \frac{1}{\sqrt{N}} \mathbf{R}_{32}\right) = \operatorname{range}(\mathcal{O}_s)$
N4SID	Numerical Algorithm for Subspace Identification ([33])
System	given by equations (3.52)-(3.53) resulting in data equation (3.86) with $i = s$, white measurement noise and white process noise
Conditions	identical with the conditions required for the PO-MOESP method, see there
$\mathbf{Z}_N$	the past input and output data block Hankel matrices are used with $\mathbf{Z}_N = \begin{bmatrix} \mathbf{U}_{0,s,N} \mathbf{Y} 0, s, N \end{bmatrix}$
RQ factorization	$\begin{bmatrix} \mathbf{U}_{s,s,n} \\ \begin{bmatrix} \mathbf{U}_{0,s,n} \\ \mathbf{Y}_{0,s,n} \end{bmatrix} \\ \mathbf{Y}_{s,s,N} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{0} \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \\ \mathbf{Q}_3 \end{bmatrix}$
SVD	$\operatorname{range}\left(\lim_{N\to\infty} \mathbf{R}_{32}\mathbf{R}_{22}^{-1} \begin{bmatrix} \mathbf{U}_{0,s,N} \\ \mathbf{Y}_{0,s,N} \end{bmatrix}\right) \approx \operatorname{range}\left(\mathcal{O}_s \mathbf{X}_{s,N}\right)$ for large enough s

Table 3.8: Subspace Identification Methods with $N \geq s > n$, Part 2.

Subspace ID Method	Weighting matrices	
PO-MOESP	$\mathbf{W}_1 = \mathbf{I}_{sl}$	$\mathbf{W}_2 = (\mathbf{Z}_N \mathbf{\Pi}^{\perp}_{\mathbf{U}_{s,s,N}} \mathbf{Z}_N^T)^{\frac{1}{2}}$
N4SID	$\mathbf{W}_1 = \mathbf{I}_{sl}$	$\mathbf{W}_2 = (\mathbf{Z}_N \mathbf{Z}_N^T)^{\frac{1}{2}}$
CVA	$\mathbf{W}_1 = (\mathbf{Y}_{s,s,N} \mathbf{\Pi}^{\perp}_{\mathbf{U}_{s,s,N}} \mathbf{Y}^T_{s,s,N})^{-\frac{1}{2}}$	$\mathbf{W}_2 = (\mathbf{Z}_N \mathbf{\Pi}^{\perp}_{\mathbf{U}_{s,s,N}} \mathbf{Z}_N^T)^{\frac{1}{2}}$

Table 3.9: Weighting matrices of equation (3.89) for the weighted SVD setting different approaches to subspace identification into relation.

By the choice of the nonsingular weighting matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ several subspace identification matrices can be related as can be seen in Table 3.9. There the Canonical Variate Analysis (CVA) based approach to subspace identification follows the idea, that the matrix on which the SVD is performed is obtained by an canonical correlation analysis (see [16] and [25]). The CVA based approach is implemented in the MATLAB function `n4sid()` and was chosen in the majority of cases if the choice of the weighting matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ in the `n4sidOptions` was set to `'auto'` (automatic determination). See the MATLAB `n4sid()` documentation for further details.

3.5.3 State-Space Model with N4SID

The approach, which leads to the pantograph LLMN (n4sid), implements state-space systems identified directly form input and output data by employing the subspace identification method N4ISD through the MATLAB function `n4sid()`, where an arbitrary high system order is chosen. Subsequently an order reduction achieved by `reduce()` is carried out to match the dimensions of the pantograph LLMN (surrogate) systems. The results shown in Section 4.5 are based on a (6×1) state space vector (noninterpretable), a (2×1) input vector (contact position η and pneumatic actuator torque M_{pa}) and an (4×1) output vector (collector head and crossbar positions and velocities, $\mathbf{y} = \begin{bmatrix} \xi & \dot{\xi} & \zeta & \dot{\zeta} \end{bmatrix}^T$. Details of the derivation of the system matrices in this approach are discussed in the following section.

By employing the MathWorks® MATLAB function `n4sid()` one obtains the system matrices $\mathbf{A}_T, \mathbf{B}_T, \mathbf{C}_T, \mathbf{D}_T$ and $\mathbf{K}_T$ of a state-space system in innovation representation (compare Section 3.5.1) in discrete time as

$$\mathbf{x}_T(k+1) = \mathbf{A}_T \mathbf{x}_T(k) + \mathbf{B}_T \mathbf{u}(k) + \mathbf{K}_T \mathbf{e}_T, \tag{3.90}$$

$$\mathbf{y}_T(k) = \mathbf{C}_T \mathbf{x}_T(k) + \mathbf{D}_T \mathbf{u}(k) + \mathbf{e}_T \tag{3.91}$$

where $\mathbf{D}_T = \mathbf{0}$ by default (no direct feedthrough) and index $_T$ indicates again the similarity transformed matrices. This result of `n4sid()` is given in form of an `idss` model, which is defined as a state-space model with identifiable parameters (supports additional functions like subreferencing). The disturbance $\mathbf{e}_T$ is defined as described in Section 3.5.1. For the identification of the system matrices for the pantograph LLMN (n4sid) using `n4sid()`, there were three options of special interest that were changed from their default settings (see Matlab documentation of `n4sid()` and `n4sidOptions()`):

- `InitialState = 'zero'`: The initial state is fixed to zero and not treated as an independent estimation parameter.
- `DisturbanceModel = 'none'`: The Kalman gain matrix is fixed to zero and not treated as an independent estimation parameter (therefore the disturbance model is switched off).
- `Form = 'modal'`: The type of canonical form is set to modal form, decreasing the number of free parameters for estimation (the default setting `'free'` allows all matrix entries to be estimated).

All these settings decrease the performance of the `n4sid()`. The first two options were deactivated, because only the system matrices $\mathbf{A}_T, \mathbf{B}_T, \mathbf{C}_T$ and $\mathbf{D}_T$ are implemented in the pantograph LLMN, without any disturbance model and independent of the initial state. The choice of a modal form of the system matrix $\mathbf{A}_T$ is done to make the non-interpretable matrices resulting from the N4SID more similar in their structure and therefore more compatible to each other. This is relevant in case the parameter blending method (see Section 3.6.2) is utilized when a pantograph LLMN (n4sid) with multiple LLMs is constructed. However through the subsequent order reduction procedure this form gets lost, and has to be restored by a transformation back to the modal form (realized with the Matlab function `canon()`) which again weakens the performance. If instability occurs during parameter blending, the output blending method (see Section 3.6.1) has to be employed.

By setting the options as described above it is also possible to reduce the order of the identified system using the Matlab function `reduce()` (see Matlab documentation), which employs Hankel singular values based model reduction functions. This function is implemented in its default setting (additive error method) and used to receive a state-space model of order $n = 6$, based on the expert knowledge of the dimension of the underlying system (DOF, see Section 2.1). This proceeding allows the application of the N4SID with an overpowering high order (e.g. $n = 100$) yielding simulation results with astonishing performance (FIT over 90[%] for zero-mean operation point data sets, see Chapter 4), while receiving a set of matrices in a dimension matching the expert knowledge about the pantograph system and therefore keeping the pantograph LLMN (n4sid) simple and fast from a computational point of view.

Summarizing the presented pantograph LLMN (n4sid) incorporates sets of identified state-space matrices $\mathbf{A}_T, \mathbf{B}_T, \mathbf{C}_T$ and $\mathbf{D}_T$, derived by the subsequent application of the MATLAB functions `n4sid()` and `reduce()` on local data sets (around the center of the LLMs), into the LLMs of the LLMN structure according to equations (3.28) and (3.29). The dimensions are given with number of states $n = 6$, number of inputs $m = 2$ and number of outputs $q = 4$, where the input and output data (sequences) are given as

$$\mathbf{u}_{n4sid} = \begin{bmatrix} \eta & M_{pa} \end{bmatrix}^T , \tag{3.92}$$

$$\mathbf{y}_{n4sid} = \begin{bmatrix} \xi & \dot{\xi} & \zeta & \dot{\zeta} \end{bmatrix}^T , \tag{3.93}$$

for $k = 1 \dots N$.

3.5.4 Modeling Issues with Subspace ID Methods

In this section a brief discussion of some issues that arise when using the pantograph LLMN (n4sid) based on subspace identification methods is carried out.

Primarily the identified state-space matrices provide no interpretability possibilities, which makes the pantograph model a **dark-gray model**. The main advantage of this approach however is the very efficient way of determining a working model just by input/output data sets, which therefore can be adapted to another pantograph geometry very quickly, yielding strong performance.

A great **computational disadvantage** is the incorporation of a preprocessing (low-pass filter) and postprocessing (look-up table) to the pantograph model (makes the model considerably slower, compare Figure 3.18 see Section 4.6). This is necessary due to the fact, that the `n4sid()` algorithm can only cope with zero-mean signals. Furthermore an online capable implementation of that approach introduces a slight phase shift to the input signals. Subsequently, due to the input partitioning, also the partitioning variable carries a phase shift, distorting the blending procedure, even if the more robust output blending method is applied.

Furthermore it is mentioned here, that applying the subspace methods to **closed loop data** is a topic of its own. In the case of closed loop identification the conditions presented in the previous sections are violated and the presented methods have to be modified. For further information consult e.g. [39, page 336, Section 9.7] or see [6], [13], [26] and [5].

3.6 LLMN Blending Methods

This section discusses the applied blending methods of the local linear model networks in state-space configuration.

As mentioned in the previous sections, the aim of this thesis is to develop a LLMN in state-space configuration for the nonlinear pantograph. In Section 3.1 two possible blending procedures were mentioned, that enable the LLMN to generate a global, blended output for each of the MIMO-system's outputs. These are the

- **Parameter Blending Method**, utilized for the **pantograph LLMN (surrogate)** and the
- **Output Blending Method**, utilized for the **pantograph LLMN (n4sid)**.

The underlying discrete-time state-space models for both pantograph LLMNs (surrogate and n4sid) are given in the following form:

$$\mathbf{x}_i(k+1) = \mathbf{A}_i \ \mathbf{x}_i(k) + \mathbf{B}_i \ \mathbf{u}(k), \tag{3.94}$$

$$\hat{\mathbf{y}}_i(k) = \mathbf{C}_i \ \mathbf{x}_i(k) + \mathbf{D}_i \ \mathbf{u}(k), \tag{3.95}$$

with the global (LLMN model) input given as

$$\mathbf{u}(k) = \begin{bmatrix} \eta(k) & M_{pa}(k) \end{bmatrix}^T. \tag{3.96}$$

Remark 3.6.1 (Equations (3.94)-(3.95)). Index $_i$ indicates an affiliation with the local linear model LLM_i, where $i = 1 \ ... \ M$.

Remark 3.6.2. The contact position η is chosen as the input partition variable, hence the one-dimensional partition space for the computation of the according validity function is defined as

$$\mathbf{\Phi}(\mathbf{z}(k)) = \mathbf{\Phi}(\eta(k)). \tag{3.97}$$

Remark 3.6.3. For the pantograph LLMN (surrogate) the matrices $\mathbf{A}_i$, $\mathbf{B}_i$, $\mathbf{C}_i$ and $\mathbf{D}_i$ are given according to equations (3.22) to (3.25) (see Section 3.4.3). For the pantograph LLMN (n4sid) these matrices are given by several identification runs using the Matlab function `n4sid()` as similarity transformed matrices (see Section 3.5.3).

3.6.1 Output Blending

The output blending is the more commonly applied method, especially in connection with LLMNs in ARX configuration. An illustration of the output blending method implemented for the LLMN in state-space configuration can be seen in Figure 3.18. In this case the overall output $\hat{\mathbf{y}}(k)$ of the LLMN model is realized by blending the outputs of the respective LLMs. The system equations (3.94) and (3.95) are evaluated at each time step k for each LLM separately providing the according local outputs $\hat{\mathbf{y}}_i(k)$. The evaluation of the validity functions collected in $\mathbf{\Phi}(\eta(k))$ (see equation 3.13) determine the interpolation of the local outputs $\hat{\mathbf{y}}_i(k)$, generating the global output $\hat{\mathbf{y}}(k)$. Therefore an additional output blending equation, which is also evaluated at each time step k, is received as (compare [19, page 342, equ. (13.3)]):

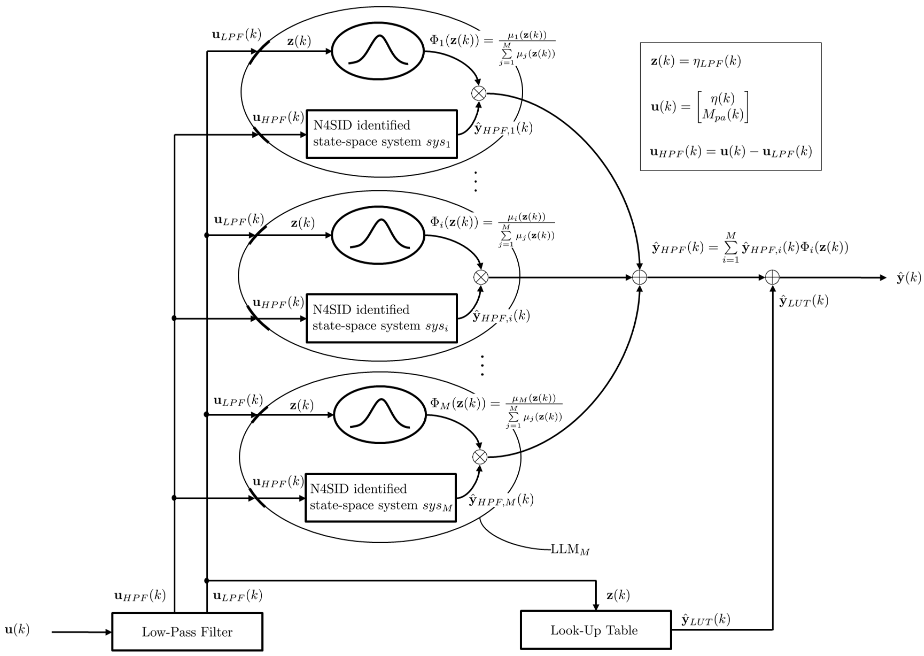


Figure 3.18: Illustration of a LLMN structure utilizing the output blending method, implemented for the pantograph LLMN (n4sid).

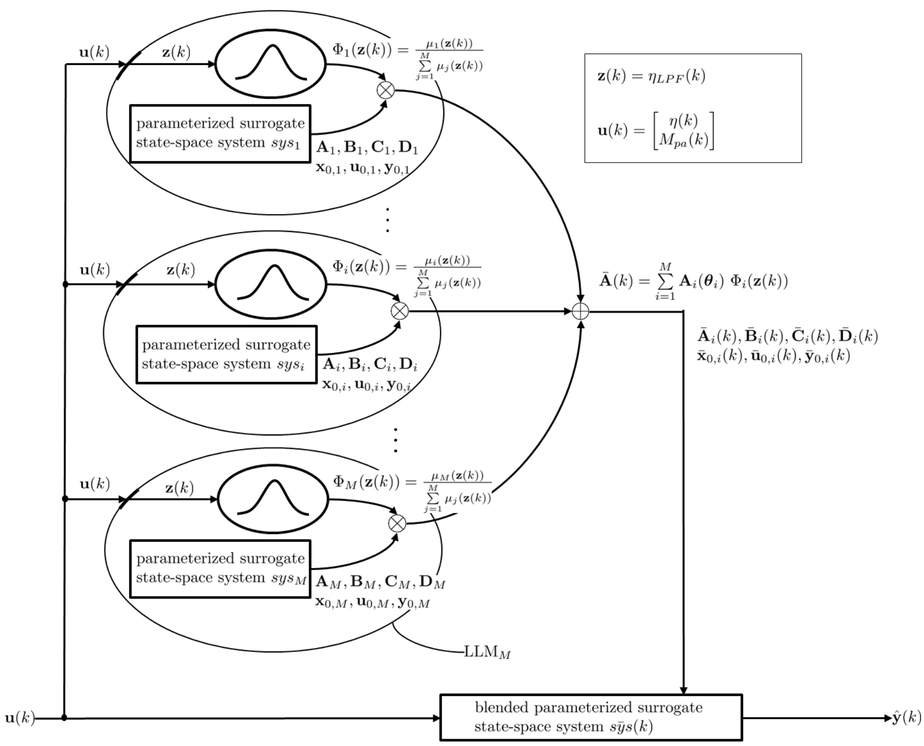


Figure 3.19: Illustration of a LLMN structure utilizing the parameter blending method, implemented for the pantograph LLMN (surrogate).

$$\hat{\mathbf{y}}(k) = \sum_{i=1}^{M} \hat{\mathbf{y}}_i(k)\ \Phi_i(\mathbf{z}(k)) \tag{3.98}$$

If the the state-space system equations (3.94) and (3.95) are inserted into the output blending equation (3.98) it then reads as

$$\hat{\mathbf{y}}(k) = \sum_{i=1}^{M} \{\mathbf{C}_i\ \mathbf{A}_i\ \mathbf{x}_i(k-1) + \mathbf{C}_i\ \mathbf{B}_i \mathbf{u}(k-1) + \mathbf{D}_i\ \mathbf{u}(k)\}\ \ \Phi_i(\mathbf{z}(k)). \tag{3.99}$$

3.6.2 Parameter Blending

In the alternative approach, utilizing the parameter blending method, the matrices of the local state-space systems get interpolated in every time step k, again determined by the evaluation of the according validity functions $\Phi_i(\eta(k))$. Therefore a blended state-space system is received for each time step k (linear time-variant system), which delivers the global output $\hat{\mathbf{y}}(k)$. This method therefore only evaluates one state-system in each time step and does not require an additional output blending equation. An illustration of the parameter blending method implemented for the LLMN in state-space configuration can be seen in Figure 3.19. The resulting state and output equation of the blended system are given as (compare [18, page 295, equ. (12.5)] and [19, page 342, equ. (13.3)])

$$\mathbf{x}(k+1) = \underbrace{\left(\sum_{i=1}^{M} \mathbf{A}_i(\boldsymbol{\theta}_i)\ \Phi_i(\mathbf{z}(k))\right)}_{\bar{\mathbf{A}}(k)}\ \mathbf{x}(k) + \underbrace{\left(\sum_{i=1}^{M} \mathbf{B}_i(\boldsymbol{\theta}_i)\ \Phi_i(\mathbf{z}(k))\right)}_{\bar{\mathbf{B}}(k)}\ \mathbf{u}(k), \tag{3.100}$$

$$\hat{\mathbf{y}}(k) = \underbrace{\left(\sum_{i=1}^{M} \mathbf{C}_i(\boldsymbol{\theta}_i)\ \Phi_i(\mathbf{z}(k))\right)}_{\bar{\mathbf{C}}(k)}\ \mathbf{x}(k) + \underbrace{\left(\sum_{i=1}^{M} \mathbf{D}_i(\boldsymbol{\theta}_i)\ \Phi_i(\mathbf{z}(k))\right)}_{\bar{\mathbf{D}}(k)}\ \mathbf{u}(k). \tag{3.101}$$

Remark 3.6.4 (Equations (3.100)-(3.101)). This type of system can be interpreted as a linear parameter varying (LPV) system, where the validity functions $\Phi_i(\eta(k))$ represents the time-varying parameter vector of the LPV system (compare [18, page 295, equation (12.4)]).

Remark 3.6.5 (Equations (3.100)-(3.101)). The state-space systems implemented in the pantograph LLMN (surrogate) contain the affine terms $\mathbf{x}_{TMO,0}$, $\mathbf{u}_{TMO,0}$ and $\mathbf{y}_{TMO,0}$ (compare equations (3.30) to (3.32)). These additional vectors also have to be blended in every time-step (see Figure 3.19). The extension of the parameter blending method to the extended state-space system is straight forward (compare equations (3.33)-(3.34)), but not carried out in the equations (3.100)-(3.101) for better readability.

Remark 3.6.6. The state-space system matrices are denoted with their dependency of the pantograph LLMN's (surrogate) parameter vector $\boldsymbol{\theta}_i$, because the parameter blending method is applied in that way for just that model. The general formulation is obtained by omitting this dependency, e.g. $\mathbf{A}_i$ instead of $\mathbf{A}_i(\boldsymbol{\theta}_i)$.

Remark 3.6.7. The application of the parameter blending method to the LLMN structure results in a single unique state-vector, which in case of the pantograph LLMN (surrogate) is also physically interpretable.

3.6.3 Stability Considerations regarding LLMNs in State-Space Configuration

The literature currently provides conservative stability proofs for the implemented model structures, which are local linear models (LLMN) in state-space configuration (see e.g. [7] and the following paragraph). However a stability analysis for the presented models is not performed and subsequently stability for the pantograph LLMNs cannot be guaranteed. To compensate for that lack of proof a discussion of the observations gained during the development of the pantograph models regarding their stability is carried out in Section 5.1.2. In general it can be stated, that the systems inside the local linear models (LLMs) are modeled in such a way, that they are all stable. According to the blending procedure and the form of the implemented state-space systems some kind of stability preserving behaviour can be observed. Examinations on blending of different types of system matrices (controllability, jordan, modal, etc. formulation) can be found in [28].

In general the stability analysis of neuro-fuzzy systems is mainly based on Lyapunov stability theory, realized as numerical methods (linear matrix inequality). There exist three popular methods which test if a Lyapunov function of a certain type can be found for the examined system. They all represent sufficient but not necessary conditions on the stability of the system and are listed here starting with the most conservative function as

- the **common** (or global) quadratic Lyapunov functions (see [7, page 53, Section 4.2]), also referred to as Lyapunov's direct method (see [19, page 616, Section 20.4.2]),
- the **piecewise** quadratic Lyapunov functions (see [7, page 58, Section 4.3], [43]) and
- the **fuzzy** (or nonquadratic) Lyapunov functions (see [7, page 66, Section 4.4]).

Lyapunov's direct method (see [19, page 616, Section 20.4.2]), which represents the most conservative approach of the three mentioned concepts, is briefly described here. It is carried out by the search for a common (or global) quadratic Lyapunov function which fulfills the stability condition given in equation (3.102). According to [31] and [32, page 27, Theorem 1, equation (2.25)] a local linear neuro-fuzzy model with M rules (LLMN with M LLMs) is guaranteed stable by fulfilling (Lyapunov stability theorem)

$$\mathbf{A}_i^T \mathbf{P} \mathbf{A}_i - \mathbf{P} < 0, \tag{3.102}$$

for $i = 1, 2, \ \dots \ , M$, a positive definite quadratic matrix $\mathbf{P}$ and the system matrices $\mathbf{A}_i$ of the local linear models.

Remark 3.6.8. This stability condition follows from the local linear model structure simply as an generalization of the stability condition of the individual LLMs state-space systems according to Lyapunov's direct method, where the determination of an individual positive definite matrix $\mathbf{P}_i$ is sufficient to guarantee the stability of the local model.

Remark 3.6.9. The stability condition 3.102 represents a linear matrix inequality, which can be solved by numerical optimization very efficiently and due to the fact, that the optimization problem is convex, finding a globally optimal solution is guaranteed [8]. The stability condition in equation (3.102) however is a sufficient but not necessary condition and considerably conservative. If no quadratic Lyapunov function can be found, no statement on the stability of the LLMN can be made utilizing this method. Further information on LMI approach can be found throughout [32].

Summarizing it has to be stated, that the topic of the stability of the pantograph LLMNs would require further research and is only treated in this thesis by a brief discussion of experienced phenomena in Section 5.1.2.

Chapter 4

Numeric Studies of Pantograph Models

This chapter is devoted to present and discuss the simulation results obtained by the developed pantograph models, and evaluate these models according to the model specifications defined in Section 1.2.1. Therefore five different model configurations (see Section 4.1) were trained and validated using two different types of reference data sets (generated by the white-box model, see Section 2.1 and Section 4.2).
Before the simulation results are presented, the simulated model configurations, the mainstays of the developed program code and the employed excitation signals will be introduced.

4.1 Introduction

In Table 4.1 an overview of the examined pantograph LLMN configurations is given. In Figure 4.1 the main components of the utilized program code are depicted. The code is written in a script based form using MathWorks® MATLAB® and selected toolboxes (e.g. System Identification Toolbox™).

For all presented results of all model configurations, as well as all the reference data sets the chosen **sampling time** is always set to $T_s = 1$ [ms] (consequentially equidistant time steps).

4.2 Excitation Signals

In this section the excitation signals for the inputs of the developed pantograph models are discussed. The same signals were used as inputs to the white-box pantograph model from [1] to generate the reference data sets. As mentioned in Section 2.2.1 equation (2.2) the chosen inputs are as follows.

Examined pantograph LLMN configurations					
Approach:	Pantograph LLMN (**surrogate**)		Pantograph LLMN (**n4sid**)		
Peculiarities:	State-space system with affine term extension, see equations (3.33)-(3.34).		Preprocessing (low pass filter) and postprocessing (look-up table) of input/output signals, additional transformation of the identified and post-processed system matrices for the approach utilizing the parameter blending method.		
Number of LLMs:	1 LLM	4 LLMs	1 LLM	4 LLMs	4 LLMs
Blending Method:	-	Parameter Blending (**PB**), see Section 3.6.2	-	Output Blending (**OB**), see Section 3.6.1	Parameter Blending (**PB**), see Section 3.6.2
Training data set:	Operation point (**OP**) data set	Whole range (**WR**) data set	Operation point (**OP**) data set	Four operation point (**OP**) data sets for local identification of the LLMs	
	10 [s] at OP65, see Figure 4.3	10 [s], see Figure 4.5	42 [s] at OP65, see Figure 4.2	42 [s] at OP25/OP45/OP65/OP85, see e.g. Figure 4.4 and 3.12	
Validation data set:	Operation point (**OP**) data set	Whole range (**WR**) data set	Operation point (**OP**) data set	Whole range (**WR**) data set	
	42 [s] at OP65, similar to Figure 4.2	120 [s], similar to Figure 4.4	42 [s] at OP65, similar to Figure 4.2	120 [s], similar to Figure 4.4	

Table 4.1: Carried out training and validation runs using five pantograph LLMN configurations and two types of data sets.

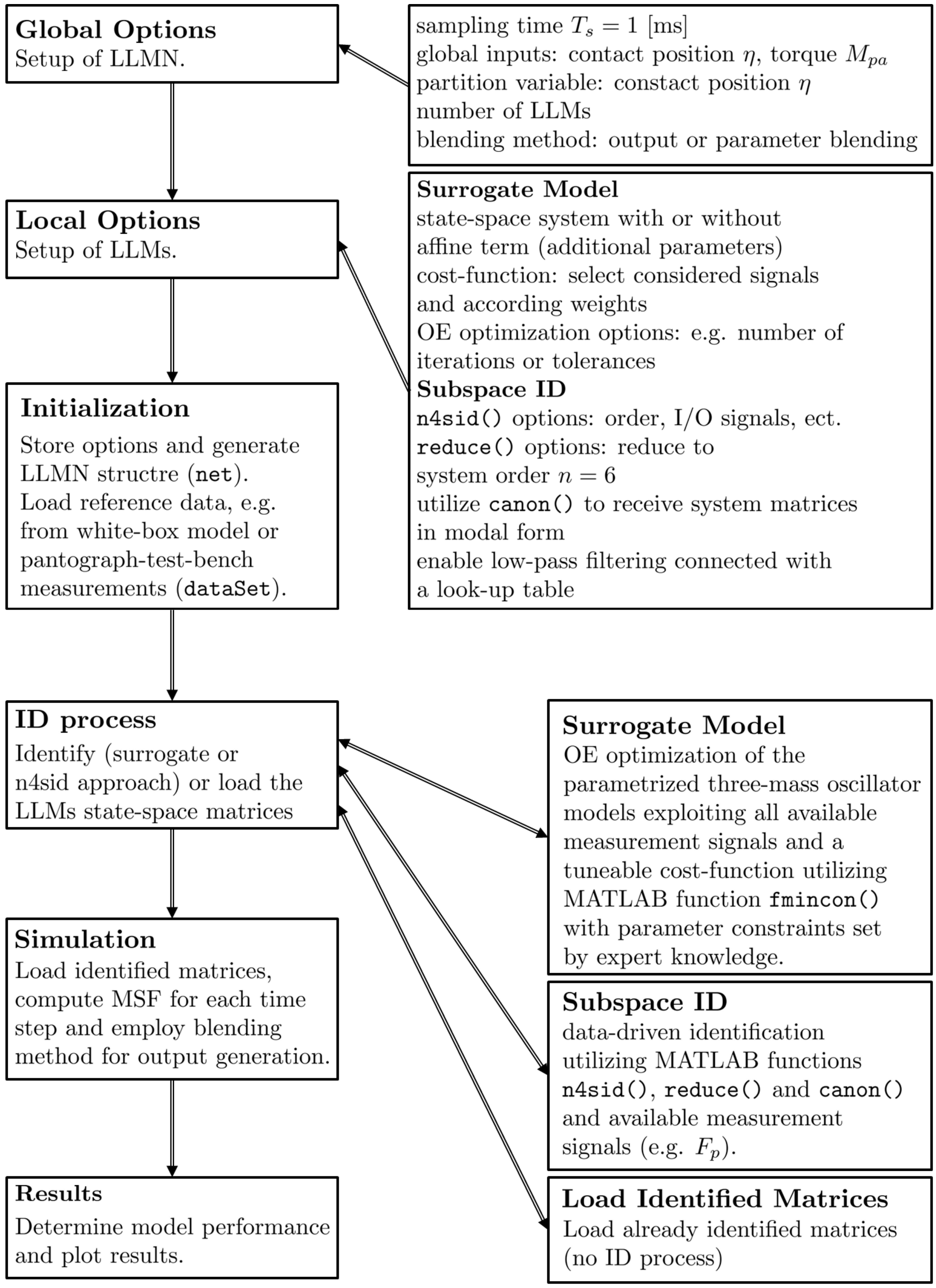


Figure 4.1: Main elements of the pantograph local linear model network program code.

- The **contact position** η [m], which has its operating range determined in the interval of [0.5, 2.5] [m] due to expert knowledge, and
- the **pneumatic actuator torque** M_{pa} [Nm], which is set to an operating point of 1310.9 [Nm] (discussed in Section 2.1.3).

Remark 4.2.1. The abbreviation of operating points, that will be used in this section, is given in a code like fashion, namely OPXY, where X denotes the operating height of the contact position (for their definition see Figure 3.12) and Y denotes the magnitude of the torque, which will be set to the previously stated OP denoted as $Y = 5$ in all simulations. Exemplary:

$$\text{OP65} \coloneqq \Big[\eta = 1.75[m] \quad M_{pa} = 1310.9[Nm]\Big]^T . \tag{4.1}$$

Remark 4.2.2. For all identification runs there were two data sets generated for which the underlying signal was of the same shape, but employed a different superimposed noise signal (see Section 4.2.1).

Remark 4.2.3. In general, while going through the engineering cycle, the observation was made, that the identification runs resulted in good performing pantograph LLMNs if the training data set contained about minimum 10000 samples (corresponds to 10 [s] with a sampling time of $T_s = 1$ [ms]). For the pantograph LLMN (surrogate) larger amounts of samples did not improve the performance significantly in most cases, while extending the computational time required for the optimization of the state-space system parameters. Therefore the training data was extracted accordingly (cut out) from the available data sets (compare Figures 4.3 and 4.5) for the training of the pantograph LLMN (surrogate).

Remark 4.2.4. The state-space matrices of the LLMs of the pantograph LLMN (n4sid) were identified using four operation point data sets, where the OPs match the positions of the centers of the LLMs of the model network (compare Figure 3.12). The simulation results of the pantograph LLMN (n4sid) for the whole range data set therefore are just validation runs and not training/identification runs (compare with Table 4.5).

4.2.1 Noise excitation

The modeled noise excitations are similar for each of the input data sets (slight parameter variations). Two types of noise sequences were utilized to excite the input signals (these are the expected types of excitation for the real-world pantograph due to expert knowledge):

- Contact position η: **white noise sequence**

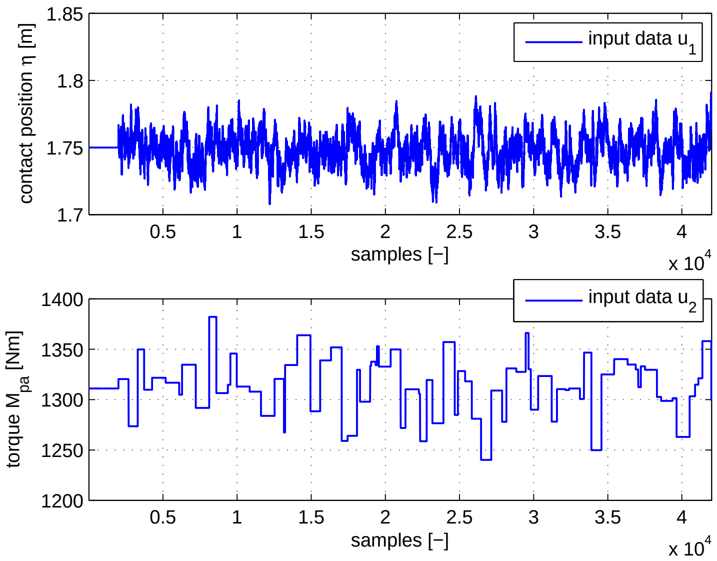


Figure 4.2: **Operation point (OP) input data set** OP65 with sampling time $T_s = 1$ [ms] and length 42 [s]. The contact position η (top) and the pneumatic actuator torque M_{pa} (bottom) are excited around operating point OP65 (compare equation (4.1)) and excited according to Section 4.2.1.

The employed white noise was filtered (postprocessed) using a first order low-pass filter with the corner frequency set to 10 Hz. Therefore the pantograph is only excited in that frequency range, which corresponds to the expected dynamic (relevant for a future implemented control scheme). The amplitude of the white noise is adjusted at about 4 [cm] according to the magnitude of the resulting contact force F_p, which is expected to be around a maximum of 1000 [N] in operation.

- Pneumatic actuator torque M_{pa}: amplitude modulated pseudo-random binary sequence (**APRBS**)

For further information on APRBS consult e.g. [19, page 570, Section 17.7]. The superimposed APRBS noise results in a maximum deviation from the operating point OPX5 = 1310.9 [Nm] of about 6% or $\sim$ 75 [Nm] over the whole simulation time. The employed hold times of the APRBS noise are in the interval $[50, 200]$ $[ms]$. These values again were defined due to expert knowledge (gained from examinations at the pantograph test bench).

4.2.2 Input Data Sets

The following figures (4.2, 4.3, 4.4 and 4.5) show the input sequences of the corresponding two basic types of reference data sets.

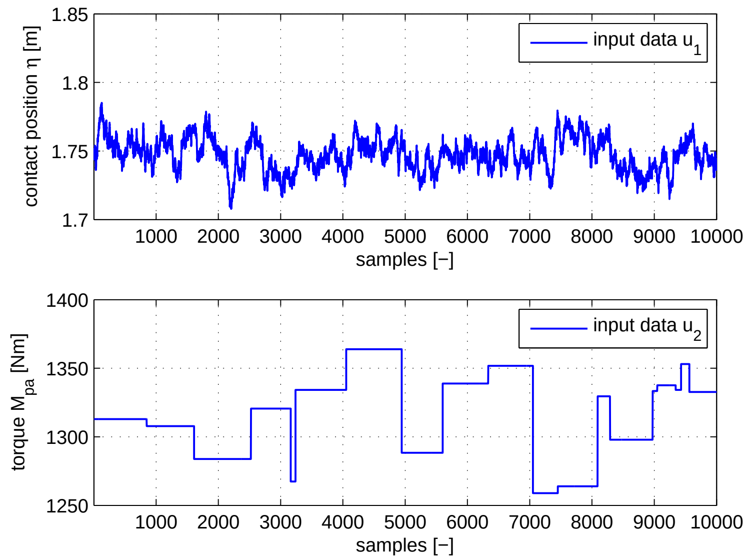


Figure 4.3: Applied training data set with length 10000 samples, extracted from the according operation point input data set OP65 depicted in Figure 4.2 (samples [10001, 20000]).

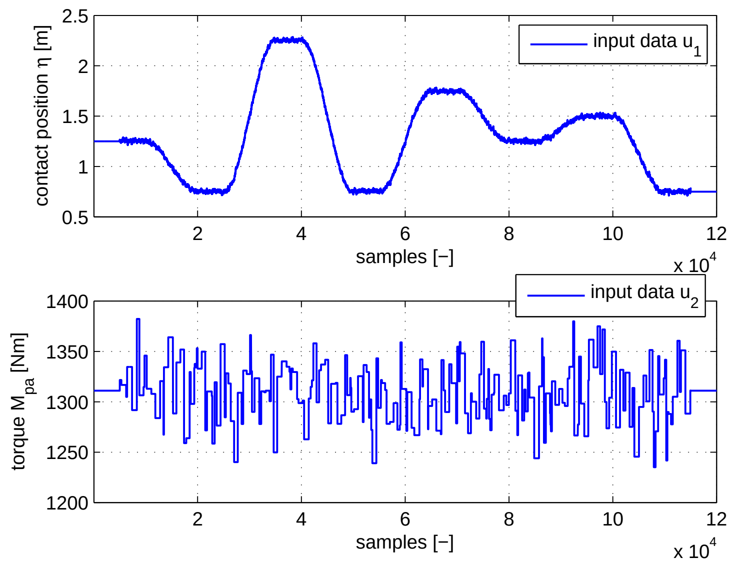


Figure 4.4: **Whole range input data set** with sampling time $T_s = 1$ [ms] and length 120 [s]. The contact position η (top) is driving the pantograph gradually through the whole operating range (excited according to Section 4.2.1), while the pneumatic actuator torque M_{pa} (bottom) is again hold around its operating point (compare equation (4.1), excited according to Section 4.2.1).

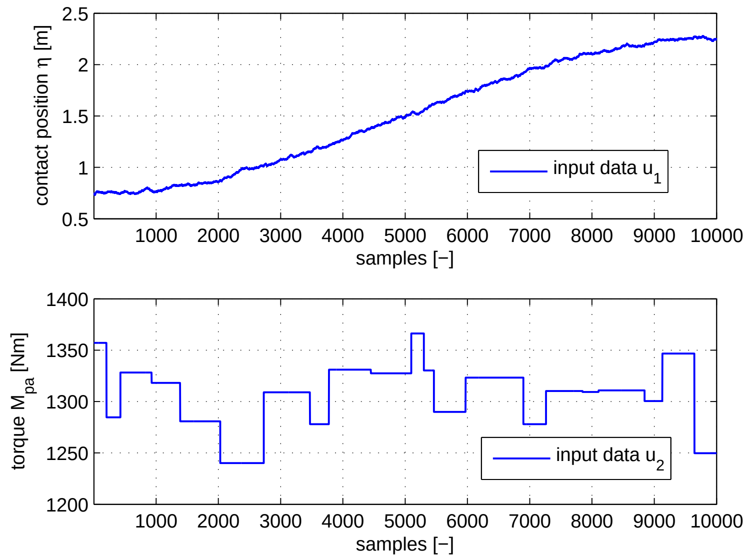


Figure 4.5: Applied training data set with length 10000 samples, extracted from the according whole range input data set depicted in Figure 4.4 (samples [25001, 35000]).

In Figure 4.2 both inputs are excited around a certain operation point (here OP65, compare equation (4.1)) and are superimposed by their specific noise signals (see Section 4.2.1). This type of data sets are referred to as **operation point (OP) data sets**.

Figure 4.4 shows a considerably long input data set, which also drives the pantograph models through the whole operating range. This type of data sets are referred to as **whole range (WR) data sets**. A certain clearance to the operating range limits was maintained, to avoid driving the white-box model into an area of the operating space where the computation becomes infeasible (evaluation of the white-box model relations no longer possible due to the pantograph geometry). Sudden, unsteady changes in the contact position input signal like steps are ruled out, because due to expert knowledge this is not the expected behaviour of a real-world contact line mounting (even if special events like a tunnel entrance or exit are considered).

These input data sets were utilized as inputs to the pantograph LLMNs to identify local linear models state-space matrices together with the corresponding output data sets. The operation point (OP) data sets (Figure 4.2 and 4.3) where utilized to identify the state-space matrices of single LLMs, while the whole range (WR) data sets (Figure 4.4 and 4.5) where used to identify the parameters of several LLMs of the pantograph LLMN (surrogate). For the pantograph LLMN (n4sid) the WR data sets only where applied for the purpose of validating the model, since all the LLMs of that model where identified form OP data sets.

As mentioned above, the Figures 4.3 and 4.5 depict the applied training data sets for the pantograph LLMN (surrogate) that were applied in the following identification runs, furthermore simply referred to as **training data set**. For the validation of the received model the whole signal length was simulated (separate validation data set with different noise excitation). The pantograph LLMN (n4sid) however was trained on the full length operation (OP) data sets, because with that approach more data samples deliver better identification results.

4.3 LOLIMOT algorithm with NMSS systems

Application of the existing ***local linear model tree (LOLIMOT) algorithm*** (compare [19, page 365, Section 13.3.1] and Section 3.2.4), using the input/output data of the pantograph white-box model (developed in [1], see Section 2.1).

The initial approach was carried out using an existing tool, which incorporated the LOLIMOT algorithm, mainly to determine a potential success of applying a local linear neuro-fuzzy type of network to the nonlinear pantograph modeling problem. As stated in Section 3.2.4, the results were discarded due to the discussed short comings of this approach in regard to the modeling goals specified in Section 1.2.1. These short comings however were not due to performance deficiency but mainly due to stability concerns. At this early stage the feasibility of the model was the main interest, therefore no superordinated pantograph model was designed. For the I/O setting of the presented results see Figure 3.8 (angle of the lower arm φ_1 compare Figure 2.2). Finally Figure 4.6 shows an achieved simulation result utilizing the LOLIMOT algorithm for the sake of completeness, where the mentioned stability issues are apparent, although all NMSS systems are stable (see Figure 4.7).

4.3.1 Performance LOLIMOT Algorithm with NMSS systems

The FIT and the computational efficiency of the LOLIMOT based pantograph model approach are presented in this section. Therefore the altered LOLIMOT algorithm was applied on the training data set according to Figure 4.4 (whole range (WR) data set). The following value is the resulting FIT of the angle of the lower bar φ_1 according to Figure 4.6

$$FIT_{LOLIMOT,\varphi_1} = 88.2502[\%]. \tag{4.2}$$

This value is quite good in comparison to other approaches, especially considering that the soaring of the examined signal, which occurs during the blending of the non-minimal state-space (NMSS) systems, is included in this result. This finding

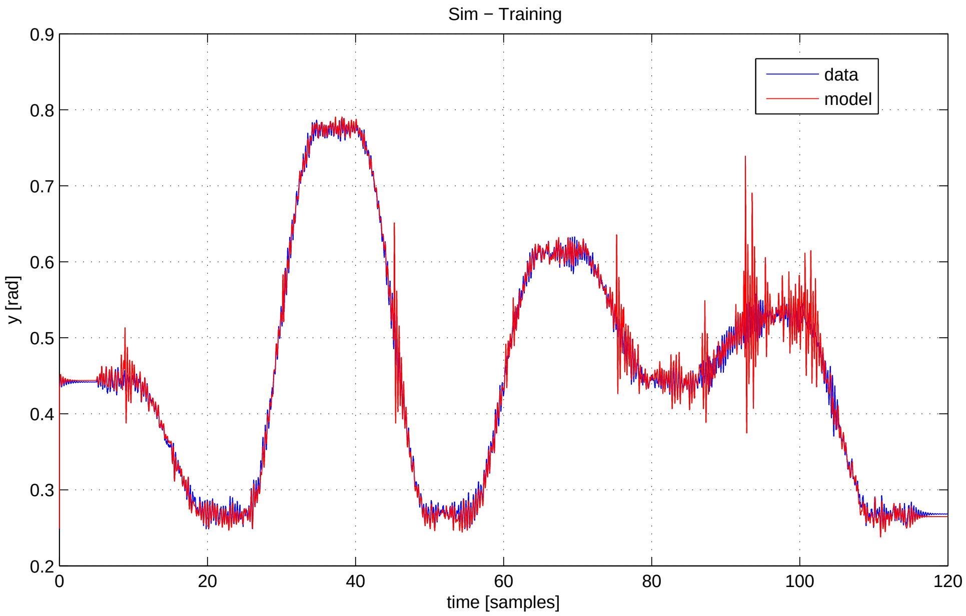


Figure 4.6: Simulation result of an identified pantograph model by utilization of the LOLIMOT approach consisting of 4 local linear models, where the instability in areas of blending of the, form the identified ARX models post-constructed, NMSS systems can be detected, see e.g. at 92 [s]. The output y is representing the angle of the lower bar φ_1 in [rad].

LLM#3	rise time [s] η	rise time [s] M	REAL	IMAG	natural frequencies ω n	damping factors ζ
	0.0215958	0.0499475	0.996982	0.0189813	19.2474	0.14764
stable	settling time [s] η	settling time [s] M	0.996982	−0.0189813	19.2474	0.14764
	1.2986	1.3756	0.987507	0.0810701	82.4285	0.111765
			0.987507	−0.0810701	82.4285	0.111765
	overshoot [%] η	overshoot [%] M	−0.486953	0.435217	2449.57	0.173919
	65.5111	66.3067	−0.486953	−0.435217	2449.57	0.173919

LLM#2	rise time [s] η	rise time [s] M	REAL	IMAG	natural frequencies ω n	damping factors ζ
	0.0472522	0.0429041	0.997896	0.0202057	20.3346	0.0935161
stable	settling time [s] η	settling time [s] M	0.997896	−0.0202057	20.3346	0.0935161
	2.07845	2.07054	0.979318	0.0897201	92.877	0.180015
			0.979318	−0.0897201	92.877	0.180015
	overshoot [%] η	overshoot [%] M	−0.620023	0.44884	2529.17	0.105693
	82.2996	81.1415	−0.620023	−0.44884	2529.17	0.105693

LLM#4	rise time [s] η	rise time [s] M	REAL	IMAG	natural frequencies ω n	damping factors ζ
	0.0362188	0.0404915	0.998188	0.0207739	20.8698	0.076533
stable	settling time [s] η	settling time [s] M	0.998188	−0.0207739	20.8698	0.076533
	2.47119	2.45591	0.986215	0.0824344	84.0387	0.123747
			0.986215	−0.0824344	84.0387	0.123747
	overshoot [%] η	overshoot [%] M	−0.515838	0.415633	2497.54	0.164886
	93.2969	85.6942	−0.515838	−0.415633	2497.54	0.164886

LLM#1	rise time [s] η	rise time [s] M	REAL	IMAG	natural frequencies ω n	damping factors ζ
	0.12073	0.0447157	0.998229	0.0196039	19.6996	0.0801802
stable	settling time [s] η	settling time [s] M	0.998229	−0.0196039	19.6996	0.0801802
	1.8229	2.44977	0.962048	0.0593345	71.749	0.512797
			0.962048	−0.0593345	71.749	0.512797
	overshoot [%] η	overshoot [%] M	−0.464731	0.49121	2361.14	0.165703
	26.4339	90.0618	−0.464731	−0.49121	2361.14	0.165703

1

Figure 4.7: Collected information plot containing information of the identified 4 NMSS systems of the LOLIMOT based pantograph model with 4 LLMs, which are apparently all stable. Nevertheless instability (soaring) is detectable during the simulation of this particular model, caused by the parameter blending of the NMSS systems (compare Figure 4.6).

also motivated the examination of more than one signal for a performance classification as it is done from here on out (e.g. position and according velocity).

The simulation time for the examined LOLIMOT based pantograph model is measured as

$$t_{sim,LOLIMOT} = 106,66[s]. \tag{4.3}$$

with the length of the data set given as $t_{ds} = 120$ $[s]$. Therefore the resulting real-time factor (RTF) for the LOLIMOT based model is determined as

$$RTF_{LOLIMOT} = t_{sim,LOLIMOT}/t_{ds} = 0.889 < 1. \tag{4.4}$$

Therefore the LOLIMOT based pantograph model would be a real-time capable model.

three-mass oscillator, parameter vector $\boldsymbol{\theta}_i$ equ. (3.35)	m_H $[5, 50]$	m_M $[5, 50]$	k_L $[0, 10^7]$	c_L $[10^{-3}, 2 * 10^5]$	r_M $[10^{-3}, 0.6]$	k_W $[0, 10^7]$	c_W $[0, 2 * 10^5]$
collector head, additional parameters equ. (3.37)		m_C $[1, 30]$	k_C $[10^{-7}, 10^7]$	c_C $[10^{-3}, 2 * 10^5]$			
affine term $\mathbf{x}_{TMO,0}$	$x_{0,i}$ $[-2000, 2000]$	with $i = 1, 2, \ldots, 6$		additional parameters according to equation (3.36)			
affine term $\mathbf{u}_{TMO,0}$	$u_{0,i}$ $[-2000, 2000]$	with $i = 1, 2$		additional parameters according to equation (3.36)			
affine term $\mathbf{y}_{TMO,0}$	$y_{0,i}$ $[-2000, 2000]$	with $i = 1, 2, \ldots, 6$		additional parameters according to equation (3.36)			

Table 4.2: Parameter constraints of all available parameters for the constraint optimization of the **pantograph LLMN (surrogate)**.

4.4 Pantograph LLMN (surrogate)

Application of a ***local linear model network (LLMN)*** with local linear models (LLMs) consisting of parametrized MIMO discrete-time state-space systems. The state-space systems represent ***mechanical surrogate models*** (three-mass oscillators) whose parameters were optimized using an output-error (OE) optimization method (MATLAB function `fmincon()`) utilizing the cost-function as given in equation (3.39) with Q_{F_p} set to 1 and the parameter constraints of this constrained optimization in `fmincon()` set as described in Table 4.2 (see Section 3.4.5).

For a detailed description of the pantograph LLMN (surrogate) see Section 3.4.

The set of parameter constrains for the constraint optimization (using MATLAB function `fmincon()`) of the pantograph LLMN (surrogate) parameters can be found in Table 4.2. These values were set for all training runs.

The identified parameter vectors $\boldsymbol{\theta}_i$ (compare Section 3.4.4) from the training (using the whole range (WR) data set) of the pantograph LLMN (surrogate) are collected in Table 4.3. The parameter values for the pantograph LLMN (surrogate) consisting of 4 LLMs were acquired by training runs starting with just 1 LLM and taking those identified values as initial parameter vectors $\boldsymbol{\theta}_i$ for $i = 1, 2$ for an training run with the pantograph LLMN (surrogate) consisting of 2 LLMs. Those two vectors then again were used as the initial vectors for the pantograph LLMN (surrogate) employing 4 LLMs, by setting $\boldsymbol{\theta}_{41} = \boldsymbol{\theta}_{42} = \boldsymbol{\theta}_{21}$ and $\boldsymbol{\theta}_{43} = \boldsymbol{\theta}_{44} = \boldsymbol{\theta}_{22}$ respectively, where the first index indicates the number of LLMs applied and the second index refers to the LLM_i, with a lower number i denoting a lower position in the partition space (i.e. positioned lower with respect to the operating height, see e.g. Figure 3.12).

Pantograph LLMN (surrogate) in affine term formulation with 4 LLMs							
LLM_i $i = 1 \ldots 4$	Parameters according to affine term state-space system equations (3.33)-(3.34)						
$\boldsymbol{\theta}_{i,TMO}$ according to equation (3.35)	m_H $[kg]$	m_M $[kg]$	k_L $[\frac{kg}{s^2}]$	c_L $[\frac{kg}{s}]$	r_M $[m]$	k_W $[\frac{kg}{s^2}]$	c_W $[\frac{kg}{s}]$
LLM_4	13.72	27.33	344.24	30.42	0.19	7999.50	2500.42
LLM_3	15.20	31.41	178.05	36.42	0.03	7999.53	2500.38
LLM_2	16.24	36.24	160.51	45.45	0.11	7999.22	2500.72
LLM_1	16.97	44.65	155.64	51.28	0.27	7998.42	2501.77
$\mathbf{x}_{TMO,0,i}$ according to equation (3.30)	$x_{0,1}$ $[m]$	$x_{0,2}$ $[\frac{m}{s}]$	$x_{0,3}$ $[m]$	$x_{0,4}$ $[\frac{m}{s}]$	$x_{0,5}$ $[m]$	$x_{0,6}$ $[\frac{m}{s}]$	
LLM_4	0.0044	2.63	-0.0009	-0.37	-0.053	0.41	
LLM_3	0.0034	2.02	-0.0011	-0.30	-0.035	0.15	
LLM_2	0.0017	1.00	-0.0005	-0.12.	-0.024	-0.69	
LLM_1	0.0003	0.15	-0.0002	0.02	0.006	-0.53	
$\mathbf{u}_{TMO,0,i}$ according to equation (3.31)	$u_{0,1}$ $[m]$	$u_{0,2}$ $[Nm]$					
LLM_4	-0.67	-0.0003					
LLM_3	-0.60	-0.0005					
LLM_2	-0.33	-0.0006					
LLM_1	-0.11	0.0003					
$\mathbf{y}_{TMO,0,i}$ according to equation (3.32)	$y_{0,1}$ $[m]$	$y_{0,2}$ $[\frac{m}{s}]$	$y_{0,3}$ $[m]$	$y_{0,4}$ $[\frac{m}{s}]$	$y_{0,5}$ $[m]$	$y_{0,6}$ $[\frac{m}{s}]$	
LLM_4	0.159	-3.04	-0.096	0.75	0	0	
LLM_3	0.096	-2.38	-0.081	0.92	0	0	
LLM_2	0.036	-1.20	-0.041	0.46	0	0	
LLM_1	-0.009	-0.22	-0.004	0.16	0	0	

Table 4.3: Identified parameters of the **pantograph LLMN (surrogate)** with 4 LLMs for an identification run using the the whole range (WR) data set (see Figure 4.5).

The determination of the initial values was experienced as a time consuming task, having a strong effect on the achievable results. The danger of the optimization to get caught in a local optima was ubiquitous. A brief discussion of this process is carried out here. Choosing an interpretable mechanical surrogate model helped in assuming initial values corresponding to the specifications of the examined real-world pantograph (which was modeled by the white-box model in [1]). E.g. the masses of the surrogate oscillator m_H and m_M were chosen according to the masses of the bars of the real-world pantograph. Similar was the choice of the length of the lever arm r_M in the surrogate model, keeping in mind the maximum forces that can arise in the real-world pantograph were known. This expert knowledge of the underlying model is also apparent in the choice of the parameter constraints in Table 4.2. The determination of initial values for the spring stiffness and damping factors however was not that obvious and required a trail and error approach. Therefore on the one hand the pantograph LLMN (surrogate) as a linear model (just 1LLM) utilizing the operation point (OP) data sets was examined for several operating points. On the other hand, the extended pantograph LLMN (surrogate), where the parameters of the collector head are set free for identification (compare parameter vector given by equation (3.37)), was examined, which makes the model more flexible and subsequently helps the optimization to converge.

Remark 4.4.1 (Table 4.3)*.* It can be seen from Table 4.3, that the affine term parameters $y_{0,5_i}$ and $y_{0,6,i}$, interpretable as offset correction terms for the position δ_M and velocity $\dot{\delta}_M$ of the lowest mass m_M of the three-mass oscillator (see Figure 3.13), are not altered from their initial zero values. This may be due to the fact, that this mass of the three-mass oscillator has no reference in the white-box model, and therefore the optimization using `fmincon()` is not altering these parameters. Therefore it could be further examined if these parameters could be set to zero, and not as free parameters.

Figure 4.8 illustrates the movement of the parameters of the pantograph LLMN (surrogate) with 4LLMs as given in Table 4.3, where the continuous movement of some parameters (e.g. masses m_H and m_M) from one LLM to another (compare Figure 3.12) indicate a successfully converged optimization.

4.4.1 Simulation Results, Pantograph LLMN (surrogate)

This section presents the achieved simulation results of the pantograph LLMN (surrogate) as plots of the LLMN output signals (compare equation 2.3) in comparison with the white-box model reference data.

Operation point (OP) data set training, pantograph LLMN (surrogate) with 1LLM

First the identification results of a single LLM of the pantograph LLMN (surrogate) will be presented utilizing the operation point (OP) training data sets as exemplarily

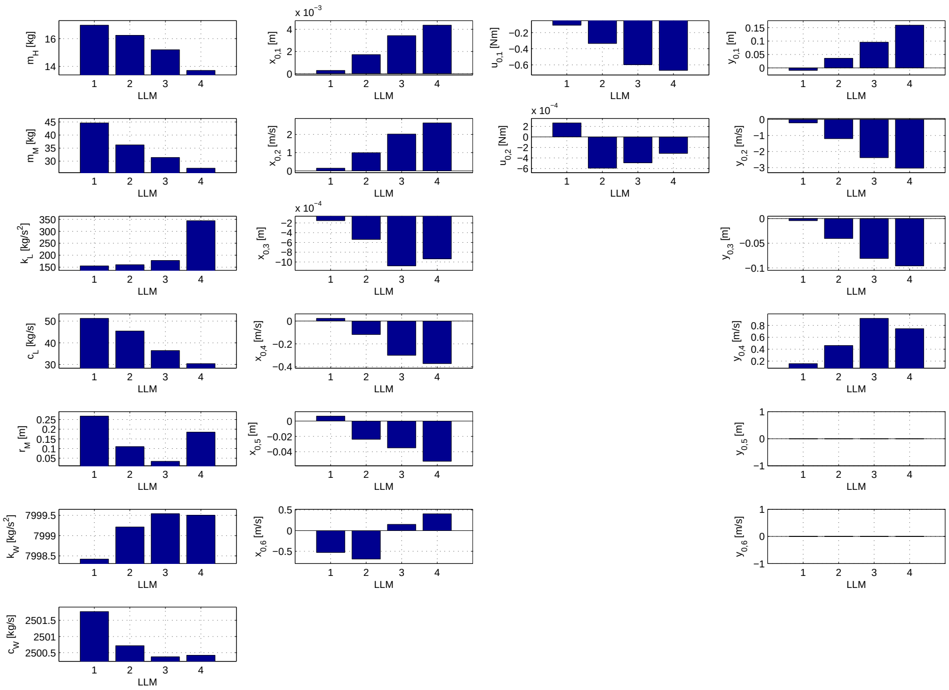


Figure 4.8: Illustration for the change of the parameter vector theta values for the **pantograph LLMN (surrogate)** consisting of 4LLMs. The x-axis represents the number of the LLM_i with $i = 1$ ... 4 corresponding to Figure 3.12, while the y-axis represents the identified parameter value (compare with Table 4.3).

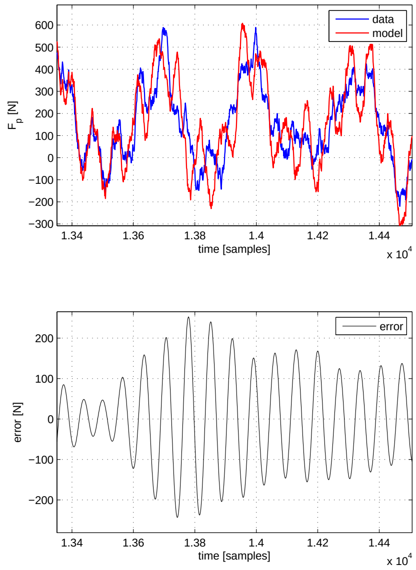


Figure 4.9: **Pantograph LLMN (surrogate), 1LLM, OP65**
Part of the result shown in Figure 4.10 for the contact force F_p (upper figure) showing the maximum absolute error (lower figure), achieved with the pantograph LLMN (surrogate) consisting of 1LLM.

depicted in Figure 4.3. A part of the validation run of the pantograph LLMN (surrogate) trained by that OP data set (OP65) can be seen in Figure 4.10.

Due to the importance of the contact force F_p to this thesis (compare Section 1.2), Figure 4.9 shows the part for the previously exhibited simulation result, where the error of the mapping of the contact force F_p reaches its maximum. The absolute error (where error $= F_{p,data} - F_{p,model}$) is additionally plotted in the lower part of the figure. The maximum relative error is of about 26 [%].

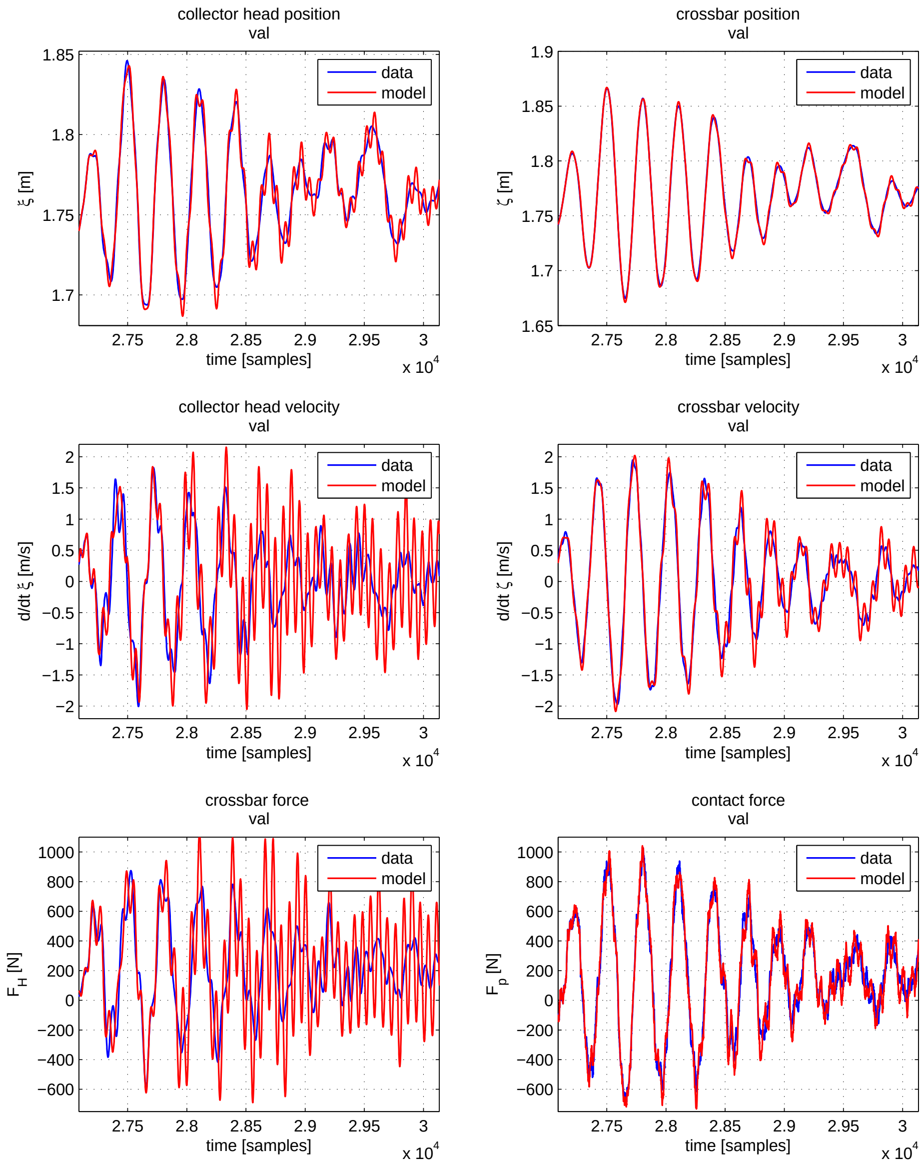


Figure 4.10: **Pantograph LLMN (surrogate), 1LLM, OP65**
Part of the result of the validation run of a pantograph LLMN (surrogate) with 1LLM for all LLMN output signals (compare equation (2.3)). This LLM consists of a state-space matrix trained from the input data set depicted in Figure 4.3 using the OE optimization of the parameter vector. The blue signals 'data' denotes the reference data generated by the white-box model, while the red signals 'model' denotes the output of the pantograph LLMN (surrogate).

Whole range (WR) data set training, pantograph LLMN (surrogate) with 4LLM

Figure 4.11 shows the result of a simulation run with a pantograph LLMN (surrogate), with 4 equidistantly positioned LLMs and affine term extended state-space system, for the validation input data set corresponding to a training (identification procedure) with the input data set depicted in Figure 4.5 (whole range (WR) data set). The performance of that particular simulation run is presented in Section 4.6. Figure 4.12 shows a zoomed out part of the same result, where the deviations from the reference data can be seen more clearly. Especially for the velocity signals ($\dot{\xi}$ and $\dot{\zeta}$) the mismatch in the higher oscillations can be detected plainly (compare with the mismatch of the modes according to f_M and f_H in Figures 4.16 and 4.17).

Figure 4.13 shows again the worst performing part of the previously exhibited simulation result of the contact force F_p together with its absolute error over the sampling time. This plot reveals that the slight deviations in the position signals ξ and ζ visible in Figure 4.12 can lead to strong deviations of the contact force. This happens due to the fact of the large magnitude of the spring stiffness k_E (10^5, given by the white-box model, known quantity) of the collector head in equation (3.14), which can blow up a small error of the position mapping. The maximum relative error is of about 30 [%] for the pantograph LLMN (surrogate).

Figure 4.14 shows again a zoomed in part of the simulation result for the contact force F_p and its absolute error, achieved with the pantograph LLMN (surrogate), where the parameters of the collector head are set free for identification. At the same position of the simulation run, the absolute error is about a third of the maximum expected value (compare with Figure 4.13). The corresponding maximum relative error of this model configuration therefore is of about 10 [%].

The last Figure 4.15 of this section shows the absolute error of the contact force F_p in connection with the MSF values (activation of the pantograph LLMN's (surrogate) local linear models) for the whole range (WR) input sequence. In this plot it can be seen, that the peaks of the contact force F_p does not stem from the blending of the LLMs (parameter blending), but from the phase shift in the mapping of the position signals (compare with Figure 4.13). The peaks obviously lie in sections where primarily one LLM is active (above 97 [%]).

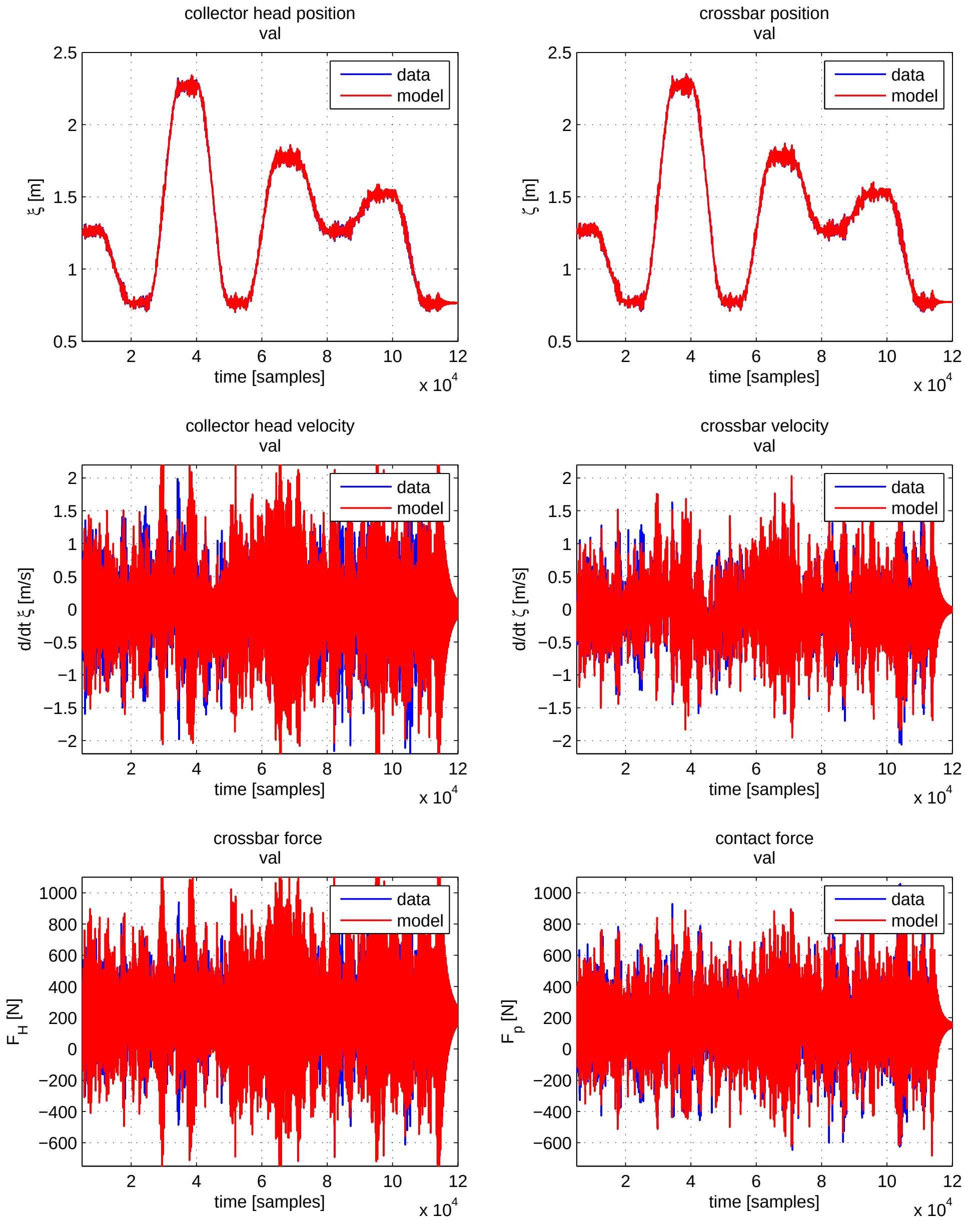


Figure 4.11: **Pantograph LLMN (surrogate), 4LLM, WR, PB** Result of the validation run of a pantograph LLMN (surrogate) with 4LLMs for a training with the input data set depicted in Figure 4.5 for all LLMN output signals (compare equation (2.3)). 'data' denotes the reference data from the white-box model, while 'model' denotes the output of the pantograph LLMN (surrogate).

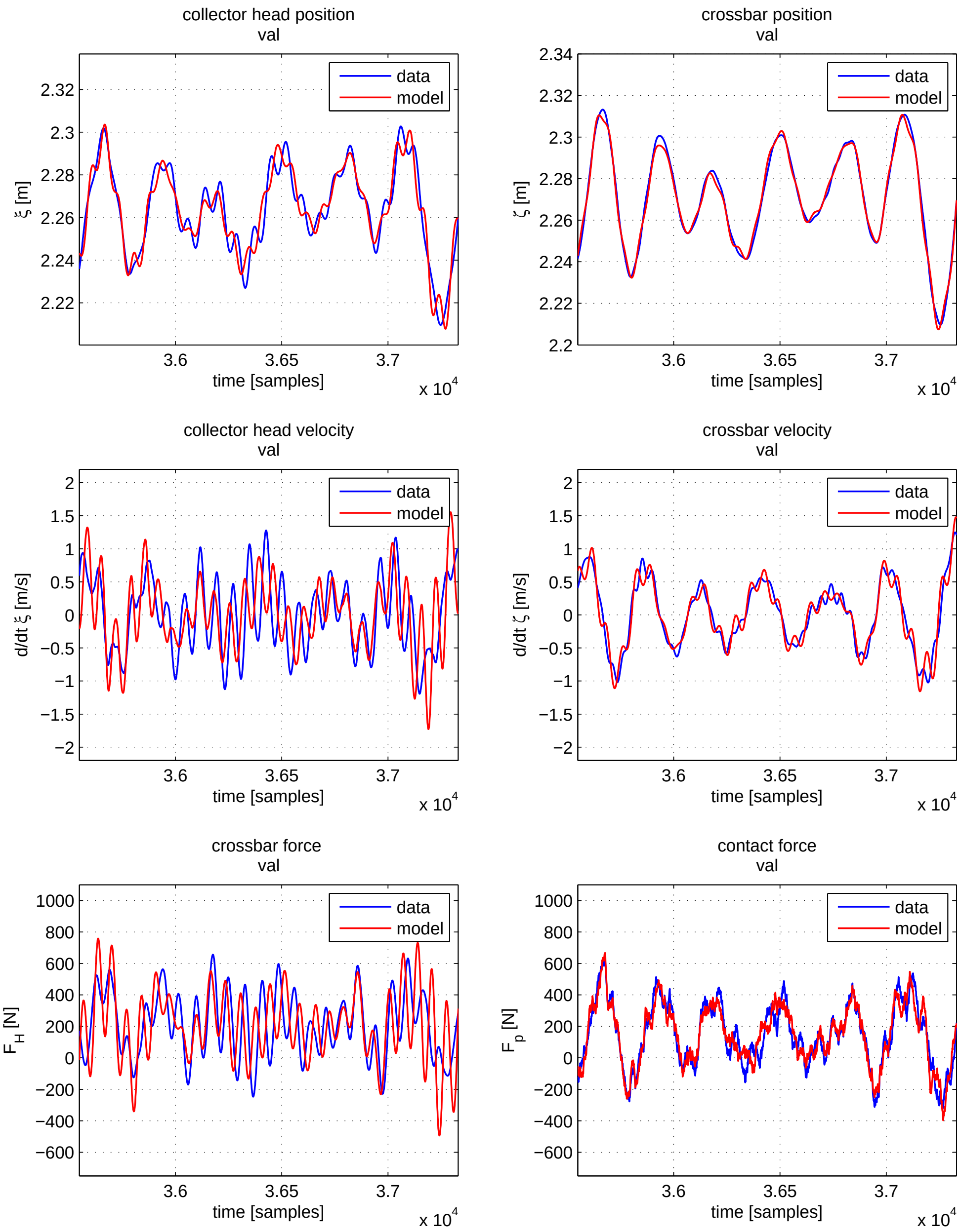


Figure 4.12: **Pantograph LLMN (surrogate), 4LLM, WR, PB**
Zoomed out part of the result shown in Figure 4.11 revealing the deviations of the pantograph LLMN (surrogate) outputs from the reference data (WBM).

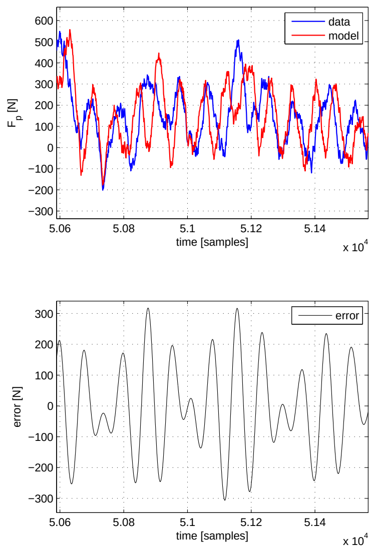


Figure 4.13: **Pantograph LLMN (surrogate), 4LLM, WR, PB**
Part of the result shown in Figure 4.11 for the contact force F_p (upper figure) showing a large absolute error (lower figure), achieved with the pantograph LLMN (surrogate).

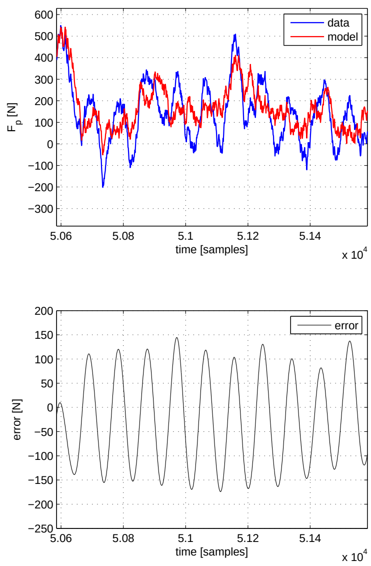


Figure 4.14: **Pantograph LLMN (surrogate), 4LLM, WR, PB, additional parameters**
Part of the simulation result achieved with the pantograph LLMN (surrogate), where the parameters of the collector head are set free for identification, for the contact force F_p showing a smaller absolute error in comparison to Figure 4.13.

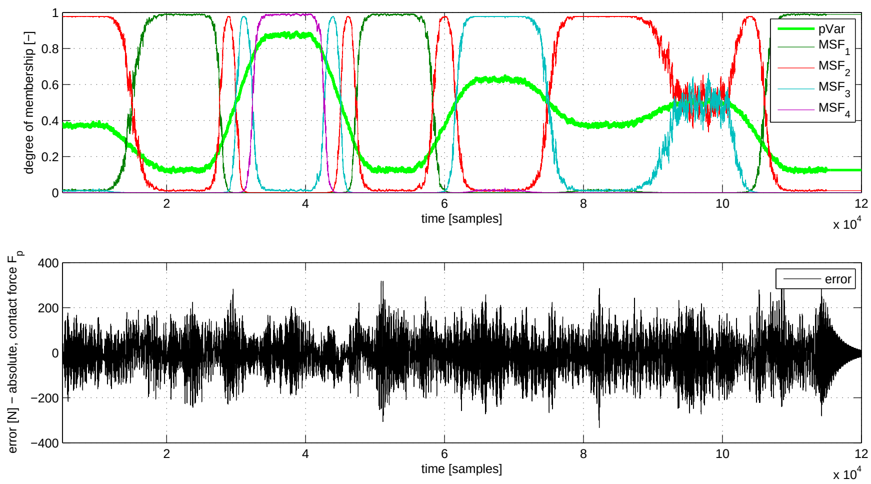


Figure 4.15: **Pantograph LLMN (surrogate), 4LLM, WR, PB**
Here the whole data set for the validation of the pantograph LLMN (surrogate) is shown, revealing peaks of the absolute error of the contact force F_p (lower figure). The maximum error occurs in sections where just a single LLM is active, as can be seen from the plot of the activation of the different LLMs (MSFs, upper figure). These peaks therefore stem from the erroneous mapping of the position signals (compare with Figure 4.13).

4.4.2 Frequency Analysis, Pantograph LLMN (surrogate)

This section treats a frequency analysis of the pantograph LLMN (surrogate), realized by a discrete Fourier transformation (DFT) of the estimated output signals of the pantograph model. These transformed signals are compared with the white-box model reference data for several **operation point input data sets**. The selected signals are the pan-head velocity $\dot{\xi}$ (see Figure 4.16), the crossbar velocity $\dot{\zeta}$ (see Figure 4.17) and the contact force F_p (see Figure 4.18). The results of the white-box model are colored blue and cyan, while the results of the pantograph LLMN (surrogate) are shown in red and magenta, respectively for two different noise excitations in each of the four selected operating points: OP25, OP45, OP65 and OP85 (compare equation 4.1 and Figure 3.12).

These plots indicate, that the pantograph LLMN (surrogate) is **not** able to map the second resonant frequency f_M (compare Section 2.1.3 and Table 2.1) according to the white-box model reference data. This particular frequency is strongly represented in the velocity signals ($\dot{\xi}$ see Figure 4.16 and $\dot{\zeta}$ see Figure 4.17) and weakly in the contact force signal F_p (see Figure 4.18). This resonant frequency f_M at about 12.2 [Hz] is originating from the collector head suspension (and therefore also strongly represented in the crossbar force signal F_H). By examining the pantograph LLMN (surrogate), where the parameters of the collector head are set free for identification

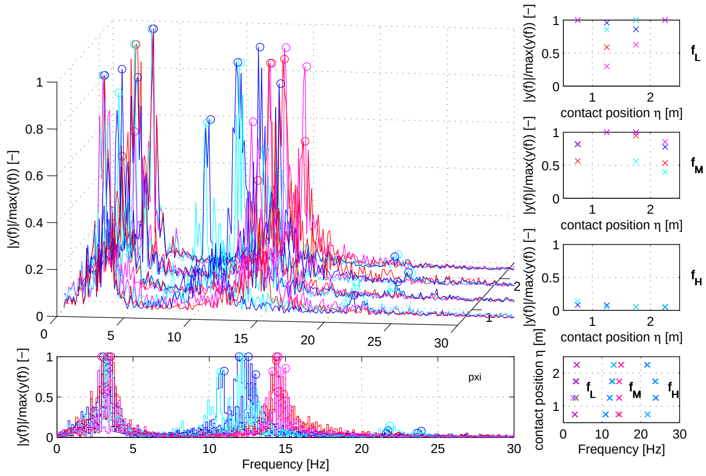


Figure 4.16: DFT of the **pan-head velocity** $\dot{\xi}$ for several identification runs with operation point (OP) input data sets spread over the operating range. Blue and cyan represent white-box model reference DFT data for two different excitation signals in each operating point, while red and magenta represent the according DFT data of the **pantograph LLMN (surrogate)**. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

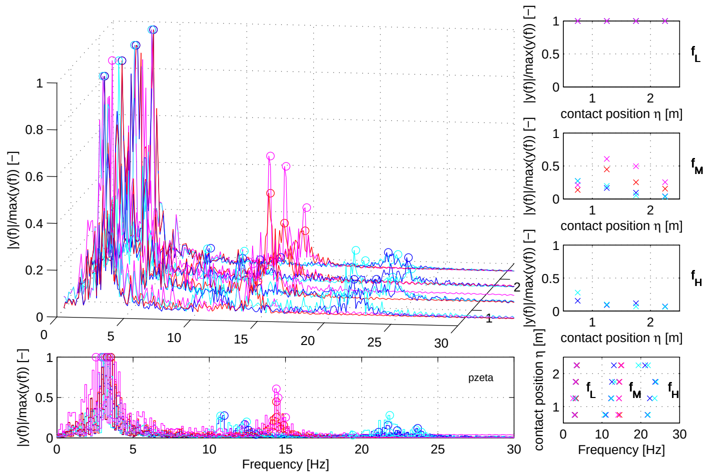


Figure 4.17: DFT of the **crossbar velocity** $\dot{\zeta}$ for several identification runs with operation point (OP) input data sets spread over the operating range. Blue and cyan represent white-box model reference DFT data for two different excitation signals in each operating point, while red and magenta represent the according DFT data of the **pantograph LLMN (surrogate)**. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

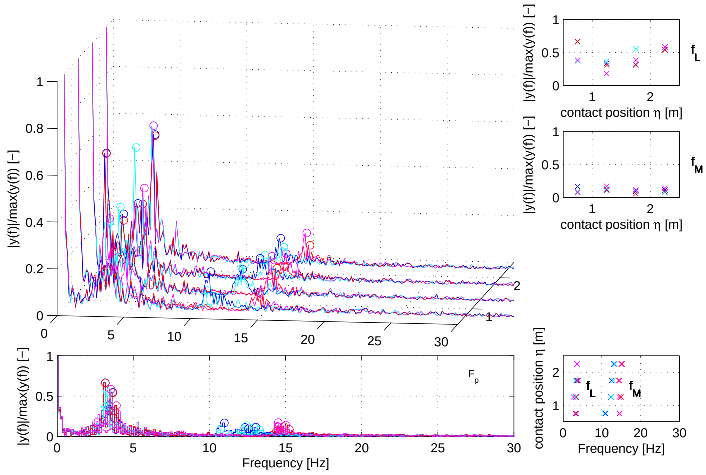


Figure 4.18: DFT of the **contact force** F_p for several simulation runs with operation point (OP) input data sets spread over the operating range. Blue and cyan represent white-box model reference DFT data for two different excitation signals in each operating point, while red and magenta represent the with according DFT data of the **pantograph LLMN (surrogate)**. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

(compare Section 3.4.4), the mode at f_M is successfully mappable. However the reference data (received by the white-box model) was generated by using the exact same values for m_C, k_C and c_C as they were set for the pantograph LLMN (surrogate). Therefore it has to be assumed that the better mapping of the resonant frequency f_M with the collector head parameters set as free parameters is only due to the enhanced flexibility (10 parameters instead on 7) of the pantograph LLMN (surrogate). By taking a look at the state-space equations of the three-mass oscillator in Section 3.4, it can be stated that the free parameters of the surrogate model have **no direct** influence on the collector head position ξ nor velocity $\dot{\xi}$. These signals can only be altered through the **coupling of the states over time**. For a further discussion of that issue see Section 5.1.1.

Furthermore the DFT plots reveal, that the highest resonant frequency f_H of the white-box model at about 22.4 [Hz] is not at all mapped by the pantograph LLMN (surrogate). This result will be examined further by taking a look at the positioning of the Eigenvalues of the state-space systems of the different LLMs, which will be discussed in the following section.

Summarizing this section, it can be stated that the presented results give an alternative view on the phase shift of the mapped signals, that was discussed in the previous section, originating form an inability to map the modes of the underlying white-box

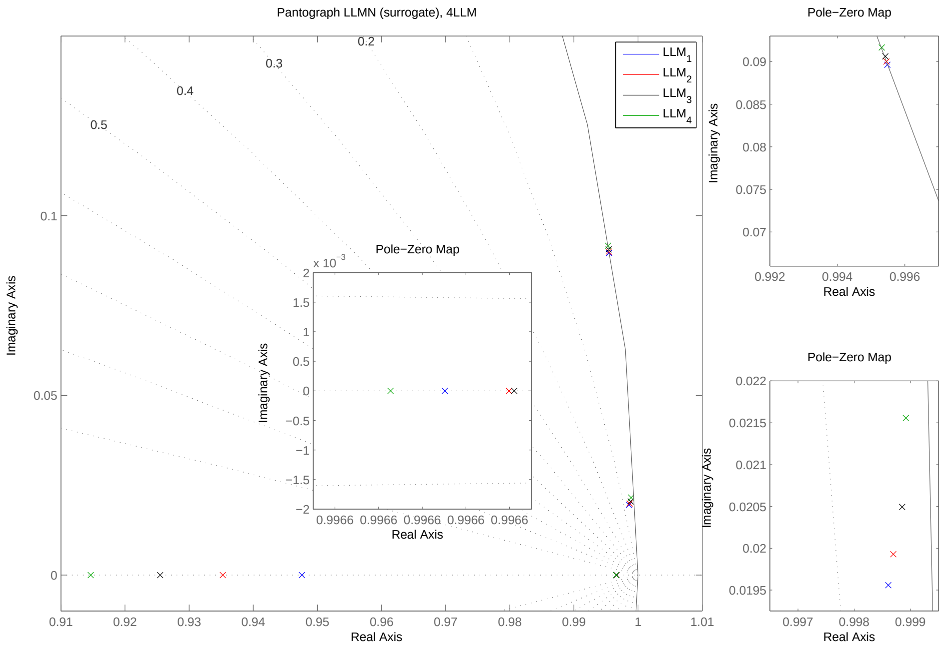


Figure 4.19: Pole positions of the parametrized state-space matrices of the **pantograph LLMN (surrogate)** with **4 LLMs**, trained by the whole range input data set (training data set as shown in Figure 4.5). A continuous movement of the poles between LLM_1 (blue), LLM_2 (red), LLM_3 (black) and LLM_4 (green) can be detected.

model correctly.

4.4.3 Stability Analysis, Pantograph LLMN (surrogate)

Figure 4.19 shows the pole positioning of the identified state-space systems of the pantograph LLMN (surrogate) with 4 LLMs in the $\mathcal{Z}$-plane. It can be stated, that all state-space systems are **stable**, as all their poles lie inside the unit circle. Furthermore it can be seen, that there are two conjugated complex pole pairs and two real poles present in every of the four state-space systems, according to their system orders ($n = 6$). Currently there exists no stability proof for LLMN in state-space configuration where the parameter blending method is applied. Therefore no guarantee for stability in regions between the operation points in which the LLMs are centered can be given (transients). For further discussion on the topic see Section 3.6.3 and Section 5.1.2.

4.5 Pantograph LLMN (n4sid)

Application of a ***local linear model network (LLMN)*** with local linear models (LLM) consisting of MIMO discrete-time state-space systems including a pre- and postprocessing of the input and output signals. The state-space systems are received by a ***subspace identification method*** (utilization of the Matlab functions `n4sid()`, `reduce()` and `canon()`, see Section 3.5) applied on input/output reference data of the pantograph white-box model. The considered signals for the subspace identification are both inputs (contact position η and pneumatic actuator torque M_{pa}) as well as four output signals (positions ξ and ζ and velocities $\dot{\xi}$ and $\dot{\zeta}$).

Remark 4.5.1. The pantograph LLMN (n4sid) requires the use of a low-pass filter and a look-up table for simulation, which is applied instead of the affine term extension as described for the pantograph LLMN (surrogate) to correct the static offsets in the signals (see e.g. Figure 3.19 for illustration). The implementation of this preprocessing of the input signals (low-pass filtering) and postprocessing of the output signals (look-up table) in the pantograph LLMN (n4sid) is necessary due to the fact, that the prior identified state-space matrices (using the Matlab function `n4sid()`, see Section 3.5) only can cope with zero-mean signals. Unless otherwise stated, the presented results of the pantograph LLMN (n4sid) were achieved utilizing the output blending method (see Section 3.6.1).

Remark 4.5.2. The corner frequency of the implemented low-pass filter is set for all presented results to $f_C = 0.25$ [Hz]. The corner frequency of the low-pass filter can be seen as an additional tuning parameter for the pantograph LLMN (n4sid). In case that f_C is set to a low value, e.g. 0.1 [Hz] (corresponding to 10 [s] delay), the activation of a LLM allocated for the current operating height gets delayed if the pantograph travels through the operating range. Consequently, due to the faulty blending procedure during transitions, the accuracy during simulation is worsened. In case that f_C us set to a higher value, e.g. 1 [Hz] (corresponding to 1 [s] delay), a certain amount of noise is captured by the low-pass filtered signal. Subsequently the input signal of the LLMs (which is given as the true unfiltered input subtracted by the low-pass filtered signal) becomes distorted, leading to worse accuracy in the simulation results.

The illustration of a case study in Figure 4.20 is showing the FIT values for the signals $\xi, \dot{\xi}, \zeta, \dot{\zeta}, F_H$ and F_p, simulated for several N4SID identified systems (various system orders from $n = 6$ up to $n = 50$) as well as for subsequently reduced systems (system order $n = 6$). In each case a training and validation data set using preprocessed (zero mean) operating point data sets (see Section 4.2) was evaluated, where the state-space matrices were identified first on the training data set (T) and validated on the validation data set (V). Additionally these tasks were carried out vice versa, hence the denotation OP25TV and OP25VT. The color code indicates good or bad

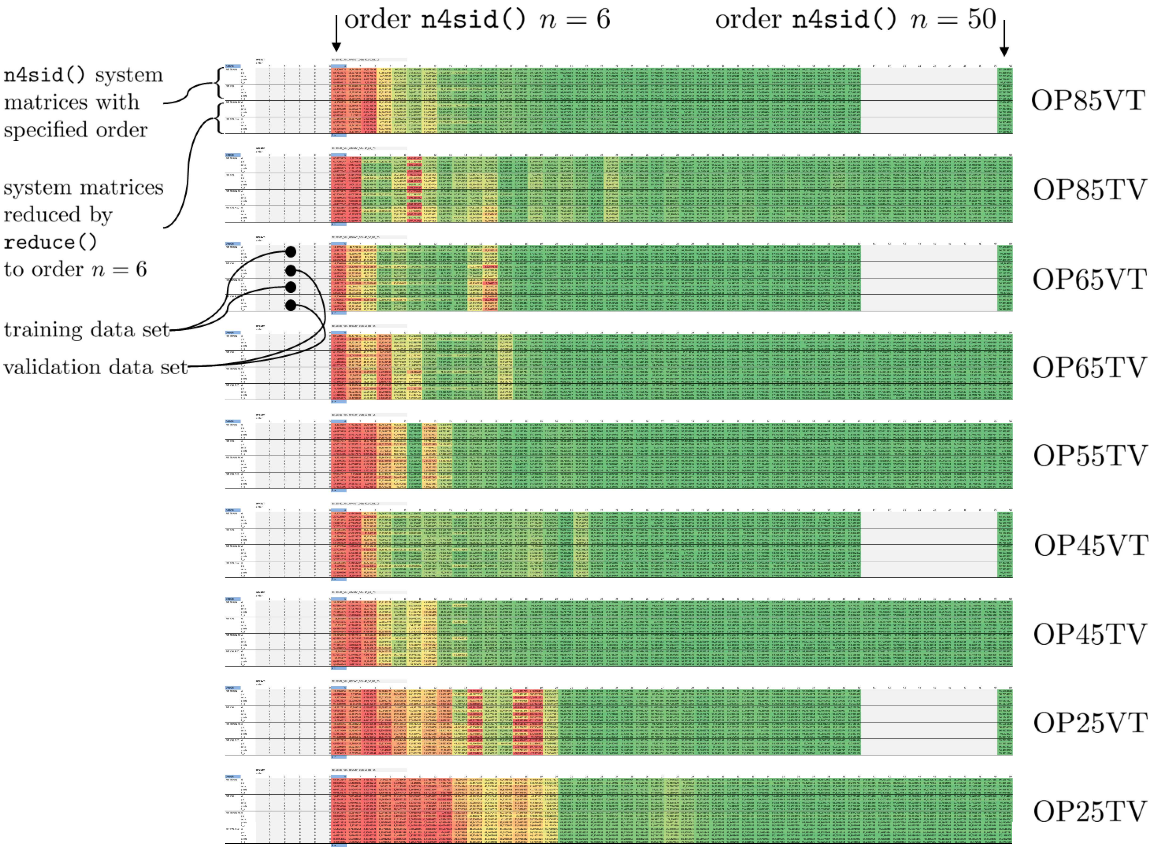


Figure 4.20: Illustration of a case study is showing the FIT values for the signals $\xi, \dot{\xi}, \zeta, \dot{\zeta}, F_H$ and F_p. Color code: red = 0 [%], yellow = 50 [%] and green = 100 [%] of FIT, while the areas in grey were not examined. The results were obtained with the **pantograph LLMN (n4sid)** consisting of **1 LLM** with varying system orders.

FIT with the legend: red = 0 [%], yellow = 50 [%] and green = 100 [%] of FIT. It can be seen that the order reduction is not compromising the performance (loss of about 1 [%] in FIT) and that the identification performance result is dependent of the operating height (this effect is originating either from the nonlinear geometry of the pantograph or the given estimated white-box models parameters determined in [1], see Section 2.1).

Remark 4.5.3 (Figure 4.20). During the application of the Matlab function `n4sid()` it was observed, that the initial value of the identified data set can have a strong impact on the resulting FIT of the simulation, using the identified matrices. This effect can be recognized in Figure 4.20, where e.g. for the OP45TV data set, the FIT drops (yellow coloring) at system order $n = 17$, although the performance was already good (dark green) for the identification with one order less. Therefore it can be stated that some combinations of initial values and system orders are not delivering the

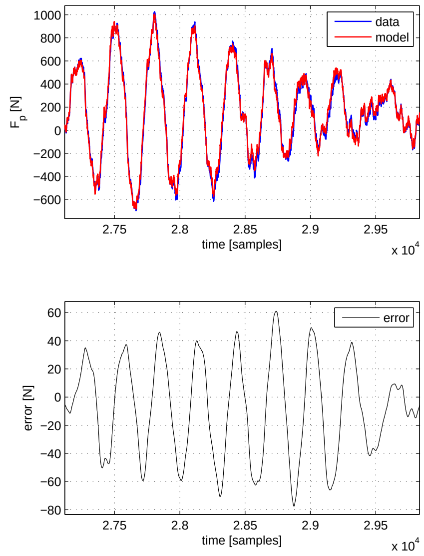


Figure 4.21: **Pantograph LLMN (n4sid), 1LLM, OP65**
Part of the result shown in Figure 4.22 for the contact force F_p (upper figure) showing a comparably small absolute error (lower figure), achieved with the pantograph LLMN (n4sid) consisting of 1LLM.

result that corresponds to the potential of this method. It is suggested, that always multiple identification runs are performed with different parts of the considered data set before any conclusions are drawn.

4.5.1 Simulation Results, Pantograph LLMN (n4sid)

This section presents the achieved simulation results of the pantograph LLMN (n4sid) as plots of the LLMN output signals (compare equation 2.3) in comparison with the white-box model reference data.

Operation point (OP) data set training, pantograph LLMN (n4sid) with 1LLM

The Figure 4.22 shows a part of the mapping of the output signals for an validation run using the `n4sid()` subspace ID method on an **operation point (OP) data set** as it can be seen in Figure 4.2 (OP65). This simulation result was achieved by the pantograph LLMN (n4sid) as describe above. The absolute error for the contact force F_p depicted in Figure 4.21 is considerably smaller than the one obtained with the pantograph LLMN (surrogate).

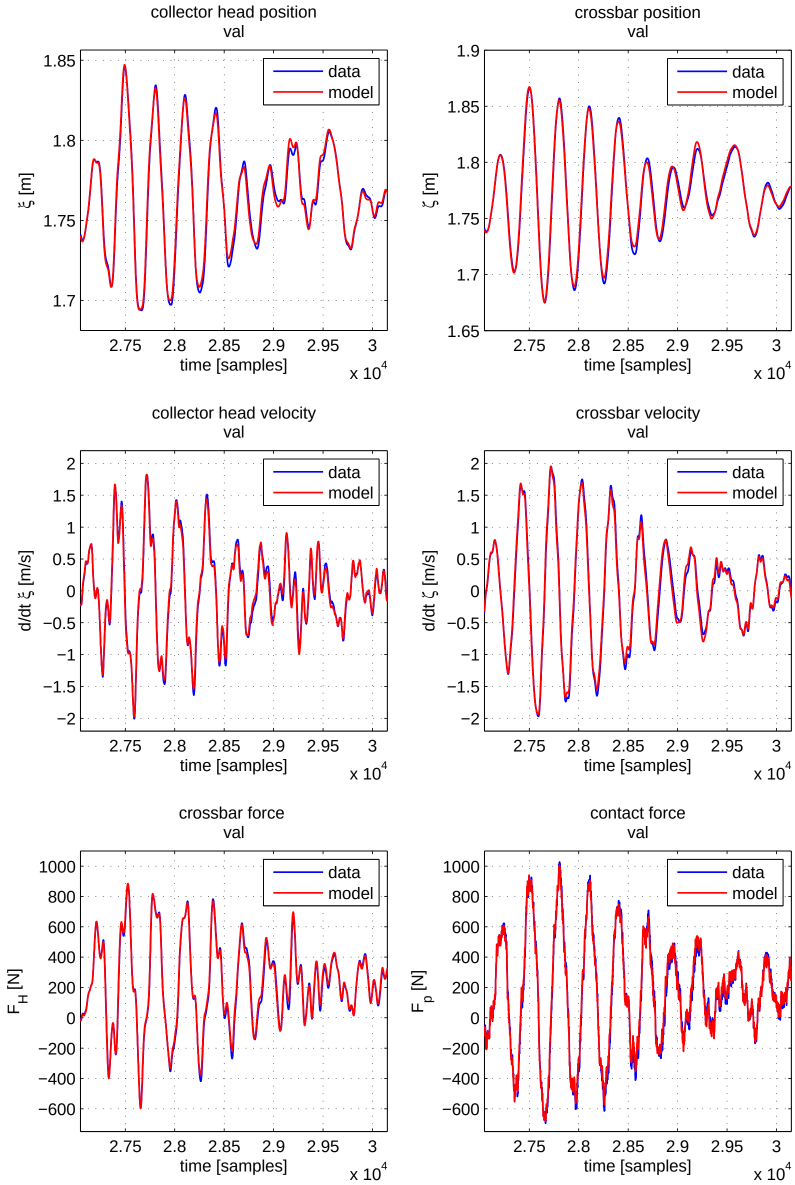


Figure 4.22: **Pantograph LLMN (n4sid), 1LLM, OP65**
Part of the result of the validation run of a pantograph LLMN (n4sid) with 1LLM for all LLMN output signals (compare equation (2.3)). This LLM consists of a state-space matrix identified from the reference data set depicted in Figure 4.2 using the N4SID method (CVA). The blue signals 'data' denotes the reference data generated by the white-box model, while the red signals 'model' denotes the output of the pantograph LLMN (n4sid).

Whole range (WR) data set training, pantograph LLMN (surrogate) with 4LLM

The results of the simulation using the **whole range (WR) data set** (Figure 4.4) are depicted in Figure 4.23. Figure 4.24 shows a close up of a certain part of that same result. The pantograph LLMN (n4sid) consisting of 4LLMs was utilized to receive these results. The state-space matrices inside the LLMs were identified prior using the N4SID method (CVA) on 4 equidistantly positioned operating points, which were OP25, OP45, OP65 and OP85 as shown in Figure 3.12.

The next plot in Figure 4.25 shows a part of the same result just for the contact force F_p and its absolute deviation form the reference signal (absolute error). This is again the part of the simulation result where the pantograph model is performing worst in regard to the mapping of the contact force F_p. The corresponding maximum relative error of this model configuration therefore is of about 22.5 [%].

When taking a look at the whole simulation result in Figure 4.26, where additionally the activation of the different local linear models (MSFs, degree of membership) is depicted, it becomes obvious that the peaks of the absolute error (ringing, soaring up) stem from the blending procedure (output blending in this case). This phenomena was not detectable for the parameter blended pantograph LLMN (surrogate), compare Figure 4.15.

Remark 4.5.4 (Figure 4.26). In comparison to the same plot of the pantograph LLMN (surrogate) (see Figure 4.15), the peaks of the contact force error signal can be clearly detected during the (output) blending of two local linear models.

Remark 4.5.5 (Figure 4.26). As mentioned earlier, the preprocessing of the input signals (low pass filtering) introduces a phase shift to the partition variable which subsequently distorts the blending procedure. The offset error of the contact force error, where the error signal is not oscillating around the zero-axis, stem most likely from the distorted (delayed) blending procedure. E.g. at position 45 [s] (45000 [samples]) this phenomena is clearly visible.

This subsection is completed by presenting a result of pantograph LLMN (n4sid) validation run, where the **parameter blending method** was applied. To achieve the results shown in Figure 4.28, an additional processing step for the state-space matrices was carried out. These matrices are received by the Matlab function `n4sid()` in canonical form and very high order, and a subsequent order reduction using `reduce()` and a transformation back to the lost canonical form as described before. The Matlab function `canon()` however does not consider the arrangement of the Eigenvectors inside the $\mathbf{A}_T$ matrix nor the setting of sings in the $\mathbf{B}_T$ and $\mathbf{C}_T$ matrices. Therefore additionally all the state-space matrices ($\mathbf{A}_T$, $\mathbf{B}_T$ and $\mathbf{C}_T$) where examined regarding the positioning of the modes inside the matrices. By yet another application of a transformation matrix, which only changes signs in the $\mathbf{B}_T$ and $\mathbf{C}_T$ matrices, the state-space systems of the different LLMs were made more compatible.

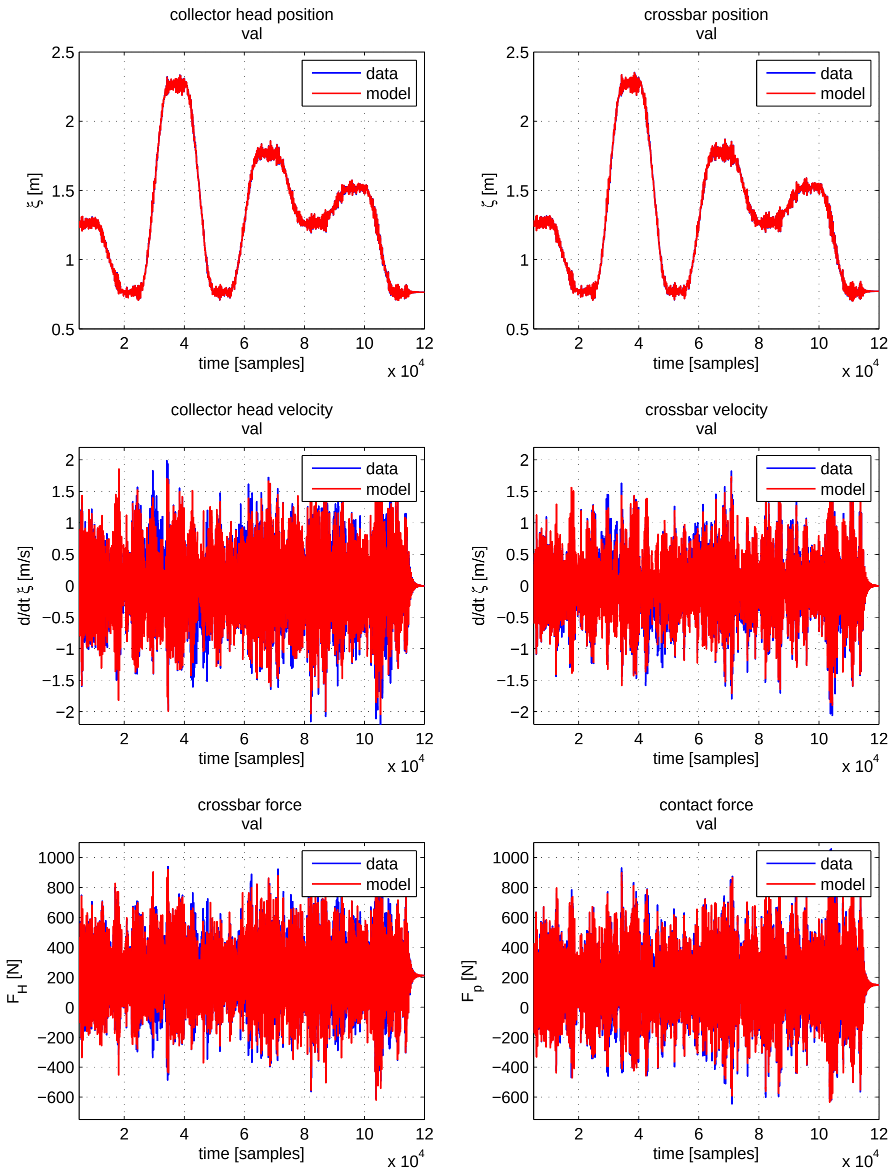


Figure 4.23: **Pantograph LLMN (n4sid), 4LLM, WR, OB**
Simulation result (validation) of the pantograph LLMN (n4sid) with 4 LLMs for the whole range (WR) data set (compare 4.4) for all LLMN output signals (compare equation (2.3)). The blue signals 'data' denotes the reference data generated by the white-box model, while the red signals 'model' denotes the output of the pantograph LLMN (n4sid).

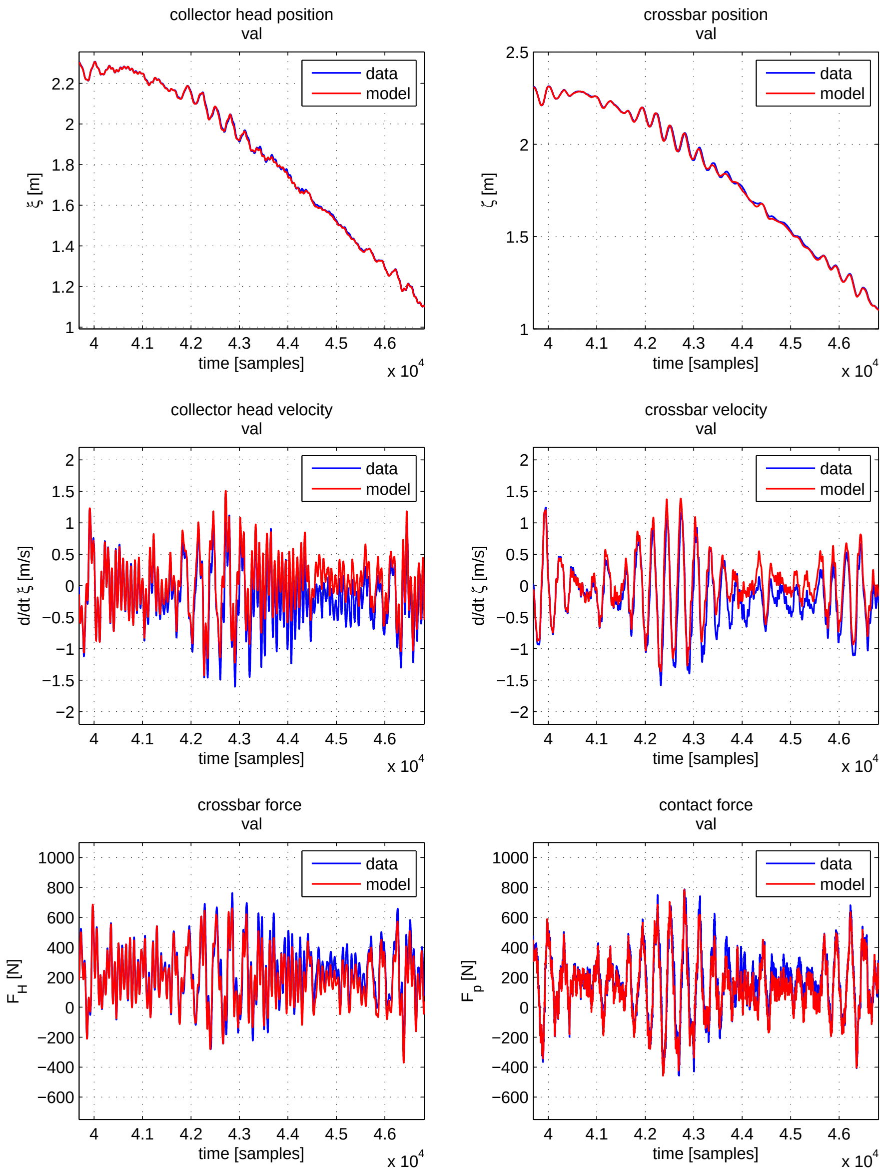


Figure 4.24: **Pantograph LLMN (n4sid), 4LLM, WR, OB**
Part of the result shown in Figure 4.23. This plot reveals that the signal is mapped right if it is located around an operating point, but becomes biased as soon as the blending between two LLMs occurs. This occurs due to the fact of the low-pass filtering of the partition variable, which therefore carries a slight phase shift.

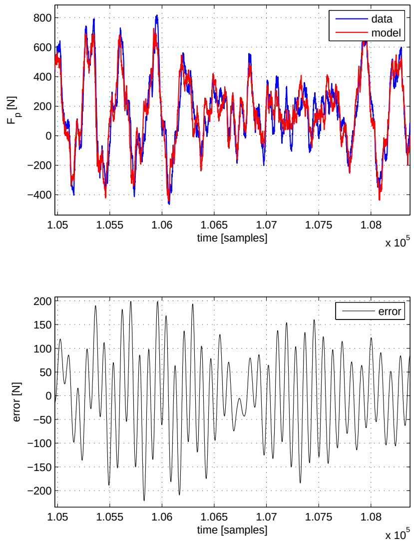


Figure 4.25: **Pantograph LLMN (n4sid), 4LLM, WR, OB**
Part of the result shown in Figure 4.23 for the contact force F_p (upper figure) and the absolute error (lower figure), achieved with the pantograph LLMN (n4sid) consisting of 4LLMs.

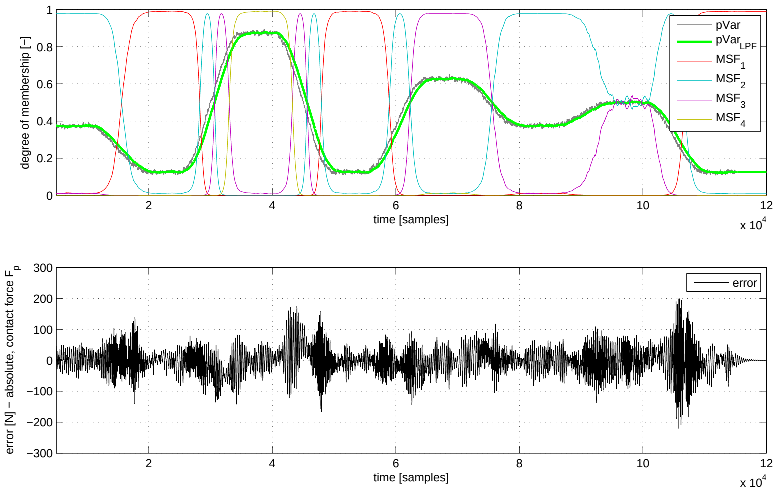


Figure 4.26: **Pantograph LLMN (n4sid), 4LLM, WR, OB**
Simulation with validation whole range data set revealing peaks of the absolute error of the contact force F_p (lower figure). A strong correlation between the interpolation of the local models and peaks of the contact force error is at hand, as it can be seen from the plot of the activation of the different LLMs (MSFs, upper figure).

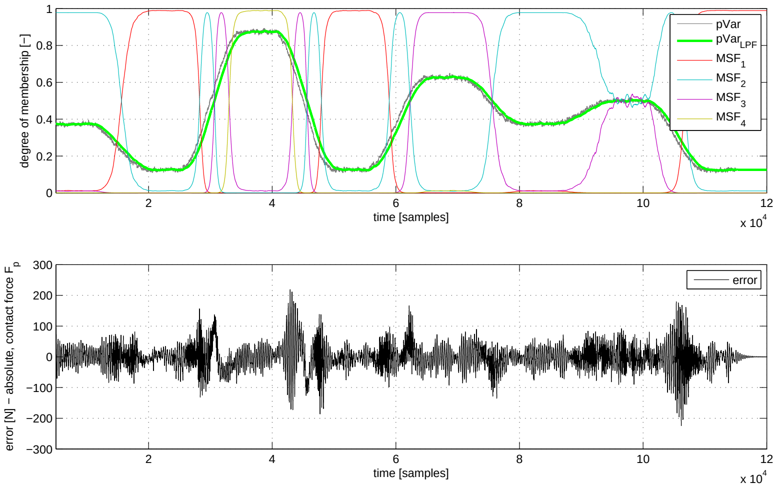


Figure 4.27: **Pantograph LLMN (n4sid), 4LLM, WR, PB**
Simulation with validation whole range data set revealing peaks of the absolute error of the contact force F_p (lower figure) with a plot of the activation of the different LLMs (MSFs, upper figure). The offset in the error signal stem from the preprocessing procedure (LPF).

This additional step however hardly can be implemented in an automated fashion (was realized manually for the presented LLMN (n4sid)). The result shown in Figure 4.28 should demonstrate the best achieved performance during this thesis using the parameter blending method of neuro-fuzzy networks in connection with a subspace identification method.

Additionally in Figure 4.27 again the absolute error of the contact force F_p for the whole simulation result together with the activation of the different local linear models (MSFs, degree of membership) is depicted.

Remark 4.5.6 (Figure 4.27). In comparison with the pantograph LLMN (n4sid) where the output blending method was implemented (see Figure 4.26), a different result is obtained with the same pantograph LLMN (n4sid) in parameter blending configuration. The absolute error seems to be smaller in magnitude of the ringing (soaring up) which stems from the blending procedure, but is significantly larger in cases where several LLMs are passed through in a considerably short period. E.g. the absolute error of the contact force $F_p$is larger at position 31 [s] (31000 [samples]) compared to the result shown in Figure 4.26. Also the offset error, most likely stemming from the preprocessing of the partition variable, displays a different behaviour, e.g. at position 31 [s] (31000 [samples]) there is a positive offset compared with the negative offset visible in Figure 4.26.

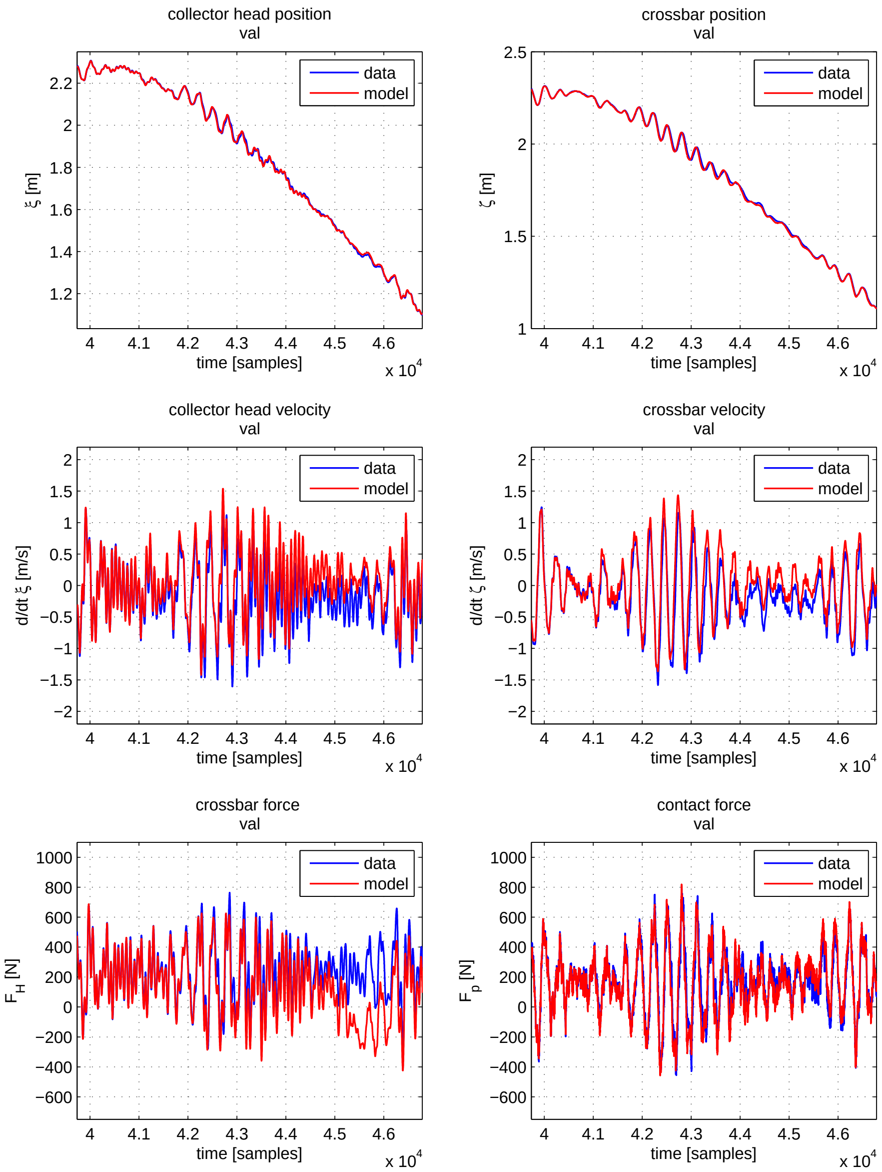


Figure 4.28: **Pantograph LLMN (n4sid), 4LLM, WR, PB**
Simulation of the LLMN model (n4sid) with 4 LLMs using the whole range data set. This results were achieved using the parameter blending method (see Section 3.6.2) performing an additional transformation of the by a subspace method identified state-space matrices inside the LLMs.

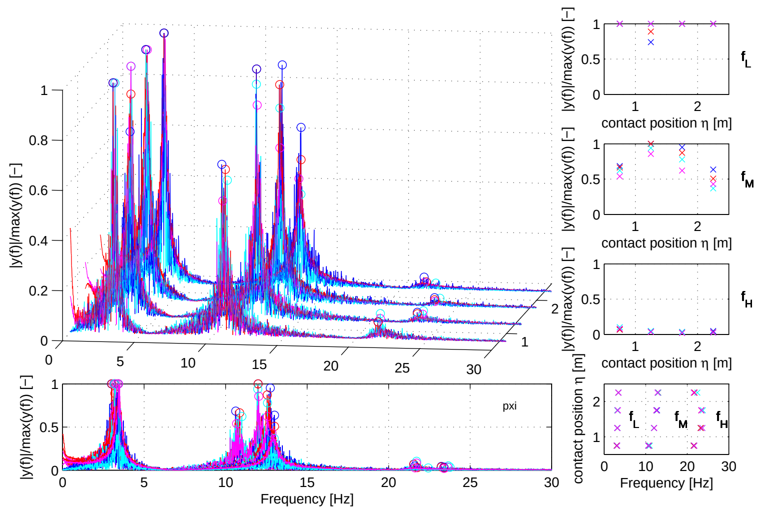


Figure 4.29: DFT of the **pan-head velocity** $\dot{\xi}$ for several simulation runs with operation point input data sets over the operating range. Blue and cyan represents white-box model reference DFT data for two different excitation signals in each operating point, while red and magenta represent the with according DFT data of the **pantograph LLMN (n4sid)**. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

4.5.2 Frequency Analysis, Pantograph LLMN (n4sid)

This section treats the frequency analysis, realized by a discrete Fourier transformation (DFT), of the pantograph LLMN (n4sid) in comparison to white-box model reference data for **operation point (OP) input data sets**. The selected signals are the pan-head velocity $\dot{\xi}$ (see Figure 4.29), the crossbar velocity $\dot{\zeta}$ (see Figure 4.30) and the contact force F_p (see Figure 4.31). The results of the white-box model reference data are colored blue and cyan, while the results of the pantograph LLMN (n4sid) are shown in red and magenta, respectively for two different noise excitations in each of the four selected operating points: OP25, OP45, OP65 and OP85 (compare equation 4.1 and Figure 3.12).

These plots indicate, that the pantograph LLMN (n4sid) is, in contrast to the pantograph LLMN (surrogate), able to map all the resonant frequencies that are detected in the white-box model reference data. This result also leads to considerable better results concerning the FIT values, as can be seen in Section 4.6. Again the positioning of the Eigenvalues of the implemented state-space matrices has an effect on the mapping of the modes, which will be treated in the following section.

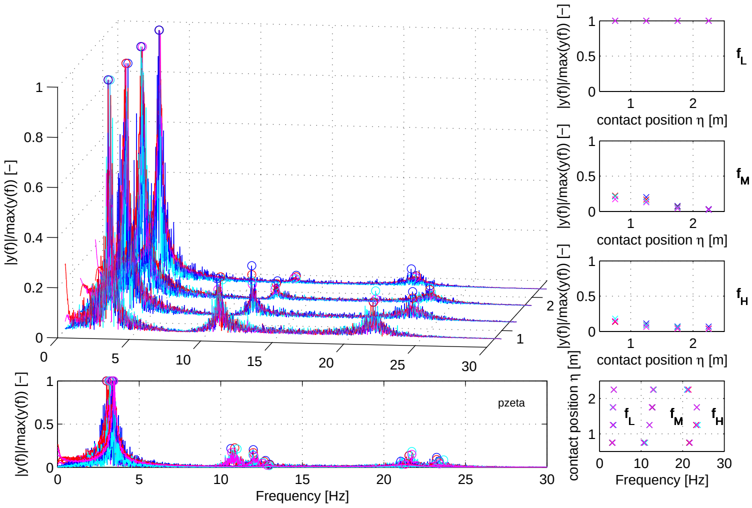


Figure 4.30: DFT of the **crossbar velocity** $\dot{\zeta}$ for several simulation runs with operation point input data sets over the operating range. Blue and cyan represents white-box model reference DFT data for two different excitation signals in each operating point, while red and magenta represent the with according DFT data of the **pantograph LLMN (n4sid)**. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

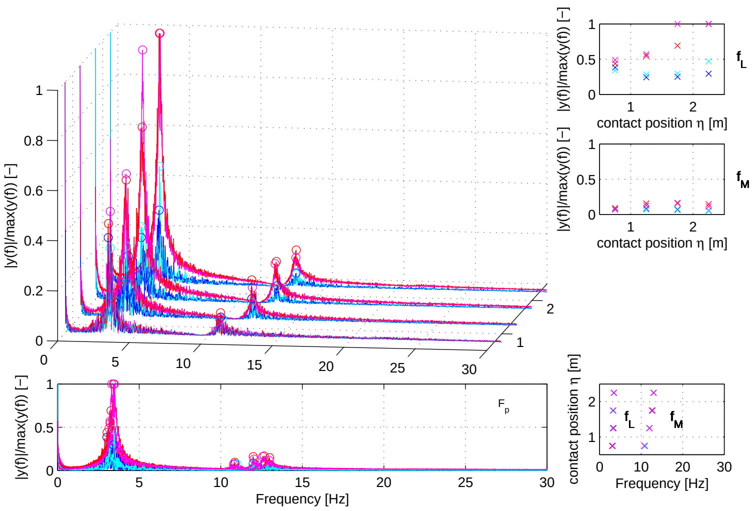


Figure 4.31: DFT of the **contact force** F_p for several simulation runs with operation point input data sets over the operating range. Blue and cyan represents white-box model reference DFT data for two different excitation signals in each operating point, while red and magenta represent the with according DFT data of the **pantograph LLMN (n4sid)**. The axis $|y(f)|/max(y(f))$ represents the normalized power spectrum of the examined signal.

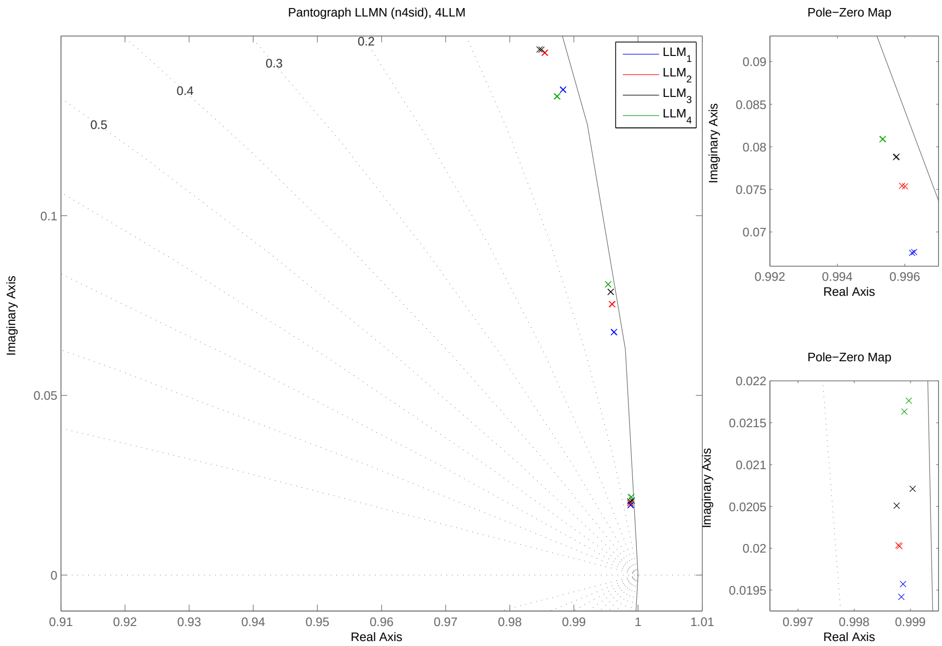


Figure 4.32: Positioning of Eigenvalues of the discrete-time system matrix **A** of the N4SID based LLM state-space systems in the $\mathcal{Z}$-domain for several LLMS (operating points). Again for this approach, a continuous movement of the poles over the operating range can be observed, even for different realizations (noise sequences). These are the pole positions of a **pantograph LLMN (n4sid)** with **4LLMs**.

4.5.3 Stability Analysis, Pantograph LLMN (n4sid)

Figure 4.32 shows the positioning of the poles on the $\mathcal{Z}$-plane for the pantograph LLMN (n4sid). The poles depicted are those of state-space system matrices identified from several operation point (OP) data sets. Again it is mentioned here, that two different noise sequences were used to generate different input signals for each of the four operating points OP25, OP45, OP65 and OP85. All the received state-space systems are stable, but as mentioned before there exists currently no stability proof for LLMN in state-space configuration where the output blending method is applied. Therefore no guarantee for stability in regions between the operation points in which the LLMs are centered can be given (transients). For further discussion on the topic see Section 3.6.3 and Section 5.1.2.

Computational Efficiency								
Data sets			Pantograph WBM		Pantograph LLMN (**surrogate**)		Pantograph LLMN (**n4sid**)	
	t_{ds} $[s]$	t_{ds} $[s]$ [1]	t_{sim} $[s]$ [2]	RTF $[-]$	t_{sim} $[s]$	RTF $[-]$	t_{sim} $[s]$	RTF $[-]$
OP65, T:	42	10	11429.49	272.13	1.35	0.135	27.96	0.666
OP65, V:	42		10807.28	257.32	5.52	0.132	27.85	0.663
WR, T:	120	10	30830.91	256.92	1.85	0.185	$83,27$	0.694
WR, V:	120		28811.58	240.97	20.56	0.171	83.45	0.695

Table 4.4: Comparison of results regarding the computational efficiency (real-time capability) of the two developed pantograph LLMNs and the reference data generating white-box model, where RTF stands for real-time factor with RTF $= t_{ds}/t_{sim}$. T indicates the training, V the validation data set.

4.6 Computational Efficiency and FIT

In this section the computational efficiency and the performance (FIT) of the two developed models is compared according to the specifications in Section 1.2.1. Table 4.4 summarizes the results in regard to the computational efficiency of the to pantograph models for different input data sets. Table 4.5 summarizes the FIT of the two pantograph models for different input data sets according to Section 4.2.2.

Remark 4.6.1 (Table 4.4)*.* As it can be seen in Table 4.4 the pantograph LLMN (surrogate) and the pantograph LLMN (n4sid) are both **real-time capable models** (RTF$<$ 1), both considerably faster than the white-box model.

Remark 4.6.2 (Table 4.4)*.* The values presented for the whole range (WR) data sets were obtained using the pantograph LLMNs with 4 LLMs. In case of the pantograph LLMN (n4sid) the results for the configuration employing the output blending method are depicted.

Remark 4.6.3 (Table 4.4)*.* The real-time factors (RTFs) of the pantograph LLMN (n4sid) are considerably worse than those of the pantograph LLMN (surrogate). This is first due to the different implemented blending methods: using the parameter blending method only requires the simulation of a single blended state-space system (compare Figure 3.19), while the utilization of the output blending method demands

[1] Lengths of training data sets for the pantograph LLMN (surrogate) only (compare Section 4.2.2).
[2] These values correspond to: OP65T $\approx$ 3.2 $[h]$, OP65V $\approx$ 3.0 $[h]$, WRT $\approx$ 8.6 $[h]$) and WRV $\approx$ 8.0 $[h]$.

the simulation of as many state-space systems as there are local linear models (LLMs) present in the local linear model network (LLMN) (compare Figure 3.18). The second more decisive reason for the slower pantograph LLMN (n4sid) is definetly the implementation of the low-pass filter and a look-up table (pre- and postprocessing of I/O signals). A linear interpolation of the values inside the look-up table is carried out in every time-step, at the price of a high computational effort.

Remark 4.6.4 (Table 4.4). The simulation using matrices derived by a linearization of the relations from the white-box model in a certain operating point delivers similar results in regard to the computational efficiency as they are presented for the pantograph LLMN (surrogate). A global linear model derived in such way can be considered as a real-time capable model.

Remark 4.6.5 (Table 4.5). Due to magnitude of the spring stiffness k_E (modeled contact to overhead line, taken from the white-box model [1]), small errors in the position signals are resulting in a big error of the contact force signal F_p, as can be seen by taking a look at equation (3.14).

Remark 4.6.6 (Table 4.5). It has to be mentioned, that for computing these FIT values only the last 75 [%] of the data points of the according signals where considered, to rule out any influence on the result by some transient phenomena due to unfavourable initial values.

Remark 4.6.7 (Table 4.5). In general these results are acceptable in regard to the contact force F_p, considering that the pantograph was moved through the whole operating range under considerable excitation and shows no signs of instability.

Remark 4.6.8 (Table 4.5). It can be observed from Table 4.5 that the FIT values of the simulations using the whole range (WR) input data set considering the position signals (collector head position ξ and crossbar position ζ) are considerably better than the FIT values of the simulations operation point (OP) input data set. It has to be mentioned, that due to the computation of the FIT values, which represent a comparison of the mapped signal with the mean of the reference signal, the results for the OP and WR data set based simulations are not directly comparable in regard to the position signals. It has to be recognized that the OP data sets are realized at a constant operating height only exited by noise, therefore the deviations resulting from phase shifts diminish the computed FIT. In case of the WR data set, the movement of the pantograph over several meters has much more impact on the FIT value than the small deviations resulting from noise excitation (about 10 [cm]).

FIT								
Model configuration →	Pantograph LLMN (**surrogate**) affine term extension				Pantograph LLMN (**n4sid**) LPF and LUT			
	1 LLM		4 LLMs		1 LLM		4 LLMs	
	-		PB		-		OB	PB
	OP data set		WR data set		OP data set		WR data set	
Signal ↓	T in [%]	V in [%]	T in [%]	V in [%]	T in [%]	V in [%]	V in [%]	V in [%]
Panhead position ξ	59.62	67.57	98.67	98.28	92.48	91.52	99.12	99.15
Panhead velocity $\dot{\xi}$	-43.56	-30.40	30.37	-14.81	90.64	90.22	61.08	64.57
Crossbar position ζ	89.56	91.14	99.54	99.16	92.19	91.31	99.07	99.02
Crossbar velocity $\dot{\zeta}$	50.43	60.48	83.67	61.74	91.27	90.57	76.63	77.91
Crossbar force F_H	-60.44	-47.93	29.26	-28.83	91.01	90.85	68.54	49.46
Contact force F_p	54.69	**64.86**	77.56	**63.33**	91.41	**90.81**	**81.33**	**81.78**

Table 4.5: Comparison of results regarding the FIT of the two developed pantograph LLMNs in their respective configurations, with OP for operation point data set and WR for whole range data set according to Section 4.2.2. 'T' stands for Training data set (training data sets according to Figures 4.3 and 4.5 for (surrogate)) and 'V' stands for Validation data set. For the meaning of the denoted data signals (positions, velocities and forces) see Figure 2.2.

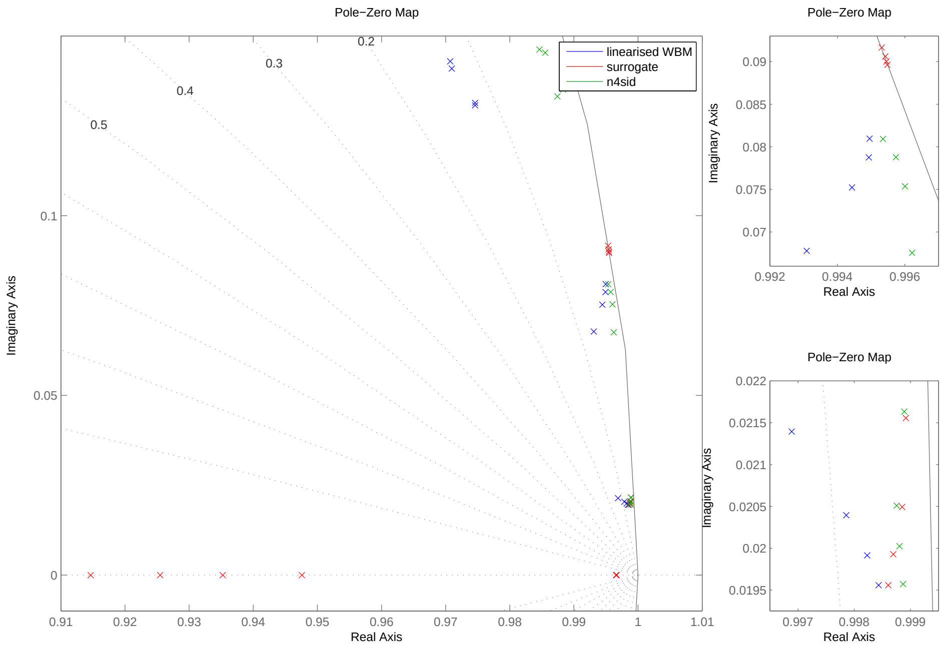


Figure 4.33: Pole positioning of the state-space systems of the linear local models of the pantograph LLMNs (surrogate and n4sid) in comparison with the poles of the - in the according operation points - linearized white-box model.

4.7 Comparison of the Pole-Positions

At the end of this chapter Figure 4.33 shows the pole positions in the $\mathcal{Z}$-plane of the pantograph LLMN (surrogate), the pantograph LLMN (n4sid) and - in comparison - the pole positions of the linearized white-box model for the respective operating points (OP25, OP45, OP65 and OP85) according to the positioning of the local linear models of the pantograph LLMNs (see e.g. Figure 3.12).

Remark 4.7.1 (Figure 4.33). In Figure 4.33 it becomes obvious, that the pantograph LLMN (n4sid) identified matrices are representing the examined white-box model better with regard to the oscillation capabilities. The pantograph LLMN (surrogate) identified two real poles instead of an additional conjugated complex pole pair. This inability to map the modes of the reference data correctly could also be seen in the presented DFT plots.

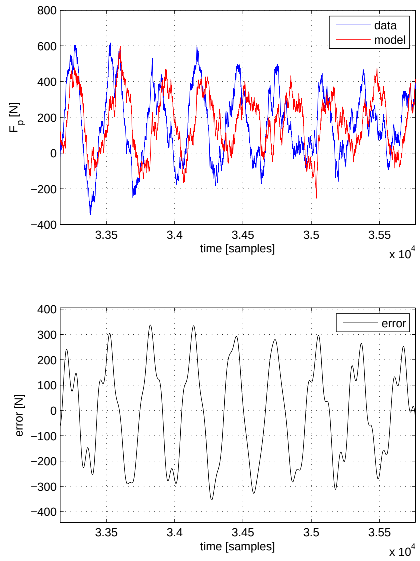


Figure 4.34: **Linearized white-box pantograph model in operating point OP45, global linear model, WR**
Part of the simulation result showing the contact force F_p (upper figure) and its absolute error (lower figure), achieved with the linearized white-box pantograph model.

4.8 Comparision of the Examined Pantograph Models

First this section delivers some results of a global linearized model, based on a state-space system obtained form the linearization of the pantograph white-box model at a certain operating point (here operation point OP45, see equation (4.1)). In Figure 4.34 the mapping of the contact force F_p of the global linear pantograph model is depicted together with its absolute error. The corresponding maximum relative error of this model configuration therefore is of about 35 [%].

Remark 4.8.1. To achieve the presented results utilizing the linearized white-box pantograph model, the resulting output signals were postprocessed to eliminated the appearing offset error. This offset error is due to the fact, that the linearization of the white-box pantograph model only delivers the state-space system matrices $\mathbf{A}_{lin,OP45}$, $\mathbf{B}_{lin,OP45}$, $\mathbf{C}_{lin,OP45}$ and $\mathbf{D}_{lin,OP45}$, and therefore this model is not able to cope with movements of the signals out of the linearized operating point (no affine terms). The postprocessing is realized as a simple subtraction of the offset from the reference data which is computed after the simulation is finished (for an online adaptation, the low-pass filter and look-up table approach introduced for the pantograph LLMN (n4sid) would be a possible realization which is not carried out here).

Table 4.6 gives the comparison of the FIT of the contact force F_p for the linearized

Comparison FIT contact force F_p				
Pantograph model configuration	**Linearized** white-box pantograph model	Pantograph LLMN **(surrogate)**	Pantograph LLMN **(n4sid)**	Pantograph LLMN **(n4sid)**
	post processing	affine term extension	LPF and LUT	LPF and LUT
	global linear	4 LLM	4 LLM	4 LLM
	-	PB	OB	PB
Data set	WR, V in [%]	WR, V in [%]	WR, V in [%]	WR, V in [%]
Contact force F_p	**53.06**	**63.33**	**81.33**	**81.78**

Table 4.6: Comparison of results regarding the FIT of the contact force F_p. The results of the global linear model (lineraized white-box pantograph model at operating point OP45) are compared with the two developed pantograph LLMNs in their respective configurations. The results represent the best achievable FIT for the whole range (WR) validation (V) data set (compare Section 4.2.2).

white-box pantograph model and the pantograph LLMNs, for the whole range data set only for the validation runs. These FIT results represent the best achievable performance of these models for the desired application case.
To round off this chapter a boxplot is presented in Figure 4.35, showing the FIT of the contact force F_p of the examined pantograph models for twelve different whole range validation signals (similar to the data set shown in Figure 4.4, with different slew rates, noise excitations and with a length of 70 [sec]). For these results three different realizations of the presented models were simulated, which were received as follows:

- **Linearized white-box model**, global linear model:
  - M11: Linearized around operating point OP45 (previously examined model).
  - M12: Linearized around operating point OP55.
  - M13: Linearized around operating point OP65.
- **LLMN (surrogate)**, 4 LLMs, parameter blending:
  - M21: Identified from the training data set presented in Section 4.2 Figure 4.5 (previously examined model).
  - M22: Identified form a training data set with the same shape as given in Section 4.2 Figure 4.5, but with a different noise excitation for both input signals. The identified parameters of model M21 were used as the starting values for the identification of this model.

  - M23: Identified form a training data set with the same shape as given in Section 4.2 Figure 4.5, but with a different noise excitation for both input signals. The identified parameters of model M21 were used as the starting values for the identification of this model.

- **LLMN (n4sid)**, 4 LLMs, output blending:
  - M31: Identified from four operation point data sets, as exemplary can be seen in Section 4.2 Figure 4.3 (previously examined model).
  - M32: Identified in the same fashion as model M21, but from according data sets with a different noise excitation for both input signals.
  - M33: Identified in the same fashion as model M21, but from according data sets with a different noise excitation for both input signals.

- **LLMN (n4sid)**, 4 LLMs, additional manual transformation of the state-space matrices, parameter blending:
  - M41: Manually adapted model M31 with better compatibility of the state-space matrices (previously examined model).

The motivation of this plot is to give the reader a lead of the general performance that can be expected from the developed pantograph models. Additionally this figure can also serve as an indication to the variation of the underlying identification procedures (parameter estimation, n4sid).

Remark 4.8.2 (Figure 4.35). The results of the global linear models (first three from the left) mainly differ in the variance of the achievable FIT of the contact force F_p. In general this figure shows, that the three pantograph models deliver significant differing results, where the performance of the pantograph LLMNs is apparently better than the global linear alternative, when mapping of the contact force over the whole operating range is of interest.

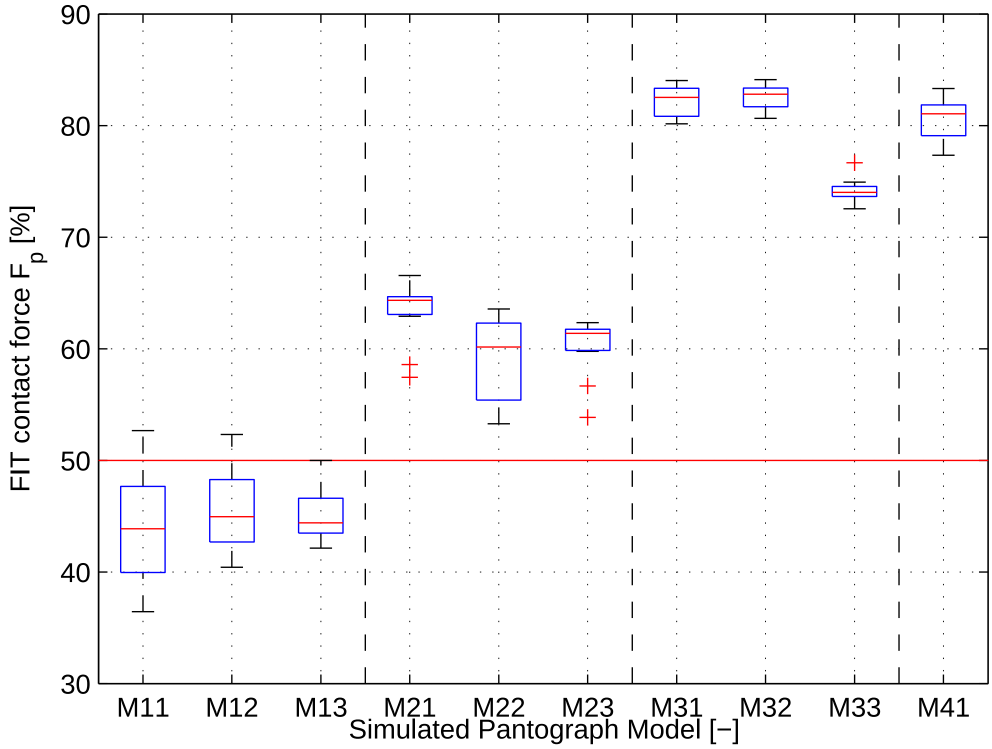


Figure 4.35: Boxplot giving a comparison of all examined pantograph models and the variation due to their underlying identification procedures. It shows the FIT of the contact force for twelve different validation signals in percent on the ordinate and the different models on the abscissa (see Section 4.8).

Chapter 5

Observations and Discussion

This chapter describes some observations made during the development of the pantograph models and discusses the results of this thesis with regard to the goals defined by the specifications in Section 1.2.1. At the end of this chapter, the main findings will be summarized in a conclusion from a methodical and a practical point of view.

5.1 Observations

This section, as announced, describes some phenomena that were encountered during the development of the pantograph models. First the pantograph LLMN's (surrogate) parameter sensitivity will be examined utilizing the Fisher Information Matrix, followed by a discussion concerning the stability and robustness of the developed models. An outlook on further research and suggestions for possible improvements of the models will be given at the end of this section.

5.1.1 Pantograph LLMN (surrogate), Parameter Influence and Fisher Information

To illustrate the influence of the parameter vector on the output of the pantograph LLMN (surrogate), Table 5.1 shows again the allocation of the parameters in the state-space system matrices as defined in the equations (3.22) and (3.23), where the free parameters according to Section 3.4.4 equation (3.35) are highlighted. The additional parameters stemming from the affine term formulation (3.33)-(3.34) are neglected in this illustration, because they are utilized to correct a static offset and do not influence the dynamic of the three-mass oscillator surrogate model.

From that illustration it can be recognized, that the first seven parameters of each parameter vector $\boldsymbol{\theta}_i$ do not have any direct influence on the states that represent positions as well as the output signal collector head velocity $\dot{\xi}$. Therefore no direct influence on that part of the three-mass oscillator, that represents the collector head is given. This is due to the fact that the output matrix $\mathbf{C}$ is defined as an eye matrix and the direct feed-through matrix $\mathbf{D}$ as a zero matrix (i.e. states are outputs, com-

Parameter Allocation for the pantograph LLMN (surrogate)	
$\boldsymbol{\theta} = [$	$m_H \quad m_M \quad k_L \quad c_L \quad r_M \quad k_W \quad c_W\ ]$
$\mathbf{A} =$	$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{(k_E+k_C)}{m_C} & -\frac{c_C}{m_C} & \frac{k_C}{m_C} & \frac{c_C}{m_C} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{k_C}{m_H} & \frac{c_C}{m_H} & -\frac{(k_C+k_L)}{m_H} & -\frac{(c_C+c_L)}{m_H} & \frac{k_L}{m_H} & \frac{c_L}{m_H} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{k_L}{m_M} & \frac{c_L}{m_M} & -\frac{(k_L+k_W)}{m_M} & -\frac{(c_L+c_W)}{m_M} \end{bmatrix}$
$\mathbf{B} =$	$\begin{bmatrix} 0 & 0 \\ \frac{k_E}{m_C} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{1}{m_M r_M} \end{bmatrix}$

Table 5.1: Allocation of the first seven free parameters of the parameter vector $\boldsymbol{\theta}$ (compare Section 3.4.4) in the system ($\mathbf{A}$) and input matrix ($\mathbf{B}$) of the pantograph LLMN (surrogate) state-space systems.

Fisher Information Matrix (FIM) for the pantograph LLMN (surrogate)

Outputs	Parameters m_H	m_M	k_L	c_L	r_M	k_W	c_W
ξ	–	–	–	–	–	–	–
$\dot{\xi}$	–	–	–	–	–	–	–
ζ	–	–	–	–	–	–	–
$\dot{\zeta}$	14	–	10	12	–	–	–
δ_M	–	–	–	–	–	–	–
$\dot{\delta}_M$	–	2	−2	0	4	−1	1
	$x_{0,1}$	$x_{0,2}$	$x_{0,3}$	$x_{0,4}$	$x_{0,5}$	$x_{0,6}$	
ξ	7	0	1	−6	−11	−12	
$\dot{\xi}$	7	6	7	0	−5	−6	
ζ	0	−7	7	0	−4	−6	
$\dot{\zeta}$	6	0	6	6	2	1	
δ_M	−18	−25	−17	−18	−5	−11	
$\dot{\delta}_M$	−11	−17	−10	−11	−6	−4	
	$u_{0,1}$	$u_{0,2}$					
ξ	0	−20					
$\dot{\xi}$	6	−14					
ζ	−8	−14					
$\dot{\zeta}$	−1	−7					
δ_M	−26	−19					
$\dot{\delta}_M$	−18	−12					
	$y_{0,1}$	$y_{0,2}$	$y_{0,3}$	$y_{0,4}$	$y_{0,5}$	$y_{0,6}$	
ξ	6	–	–	–	–	–	
$\dot{\xi}$	–	5	–	–	–	–	
ζ	–	–	6	–	–	–	
$\dot{\zeta}$	–	–	–	6	–	–	
δ_M	–	–	–	–	−5	–	
$\dot{\delta}_M$	–	–	–	–	–	−5	

Table 5.2: **Exponents** of the entries of the FIM for all parameters of the pantograph LLMN (surrogate) with 4LLMs (compare Section 4.4), where '–' indicates no parameter sensitivity on the corresponding output.

pare Section 3.4). By looking at the entries of the Fisher information matrix given in Table 4.5, this phenomenon becomes more obvious.

The Fisher Information Matrix (FIM), as defined in [9], delivers a measure or insight of how much of an impact a certain parameter has on a certain output. The results of the FIM as they are presented in Table 5.2 are based on the partial derivative of the model output with respect to the according model parameter, without taking the dependence of the state vector on the past values into account (coupling over time).

However the parameter sensitivity $\mathbf{\Psi}(k)$ is computed for the identified pantograph LLMN (surrogate) with 4 LLMs (compare Section 4.4) for each output and parameter separately (MIMO system) in each time step and subsequently the FIM for a certain simulation (whole range WR data set in this case) is obtained as

$$\boldsymbol{\mathcal{I}} = \frac{1}{\sigma^2} \sum_{k=1}^{N} \mathbf{\Psi}^T(k)\mathbf{\Psi}(k), \tag{5.1}$$

with the variance σ of the according output. The parameter sensitivity vector $\mathbf{\Psi}(k)$ is given as the partial derivation of the parameter dependent terms when equation (3.100) is inserted into equation (3.101) (taking the parameter blending into account) as

$$\begin{aligned} \mathbf{\Psi}(k) = \frac{\partial \hat{\mathbf{y}}(k)}{\partial \boldsymbol{\theta}} = \bar{\mathbf{C}}(k) \left(\sum_{i=1}^{M} \frac{\partial \mathbf{A}_i}{\partial \boldsymbol{\theta}_i}\ \Phi_i(\eta(k)) \right) \ (\mathbf{x}(k-1) - \bar{\mathbf{x}}_0(k)) \\ - \bar{\mathbf{C}}(k)\bar{\mathbf{A}}(k) \left(\sum_{i=1}^{M} \frac{\partial \mathbf{x}_{0,i}}{\partial \boldsymbol{\theta}_i}\ \Phi_i(\eta(k)) \right) \\ + \bar{\mathbf{C}}(k) \left(\sum_{i=1}^{M} \frac{\partial \mathbf{B}_i}{\partial \boldsymbol{\theta}_i}\ \Phi_i(\eta(k)) \right) (\mathbf{u}(k-1) - \bar{\mathbf{u}}_0(k)) \\ - \bar{\mathbf{C}}(k)\bar{\mathbf{B}}(k) \left(\sum_{i=1}^{M} \frac{\partial \mathbf{u}_{0,i}}{\partial \boldsymbol{\theta}_i}\ \Phi_i(\eta(k)) \right) \\ - \bar{\mathbf{C}}(k) \left(\sum_{i=1}^{M} \frac{\partial \mathbf{x}_{0,i}}{\partial \boldsymbol{\theta}_i}\ \Phi_i(\eta(k)) \right) + \left(\sum_{i=1}^{M} \frac{\partial \mathbf{y}_{0,i}}{\partial \boldsymbol{\theta}_i}\ \Phi_i(\eta(k)) \right) \end{aligned} \tag{5.2}$$

which gets evaluated at each time step (solely partial derivative considered).

Remark 5.1.1 (Equation 5.2). The derivation of the output matrix $\mathbf{C}_i$ is a zero matrix for every parameter, while the direct feed-through matrix $\mathbf{D}$ is defined as a zero matrix.

Remark 5.1.2 (Equation 5.2). The matrices and vectors $\bar{\mathbf{A}}(k)$, $\bar{\mathbf{B}}(k)$, $\bar{\mathbf{C}}(k)$, $\bar{\mathbf{x}}_0(k)$ and $\bar{\mathbf{u}}_0(k)$ represent in terms of better readability the blended matrices and vectors evaluated at each time step, e.g. $\bar{\mathbf{A}}(k) = \left(\sum_{i=1}^{M} \mathbf{A}_i(\boldsymbol{\theta}_i)\ \Phi_i(\eta(k)) \right)$, where k is the current time step, M the number of LLMs and the contact position η the one-dimensional partition variable. Compare with denotation in Section 3.6.2 equations (3.100)-(3.101).

Remark 5.1.3. For further research and an evaluation of the coupling of the states over the past values can be realized by an Fisher information matrix in output error (OE) configuration (see e.g. [10, page 254, Section 3.2]). There the FIM $\boldsymbol{\mathcal{I}}_{OE}$ has to be computed recursively by taking the parameter sensitivity vector's total derivation of the model output with respect to the parameter vector into account. This evaluation is not carried out in this thesis. The computation of the parameter sensitivity is then given as

$$\Psi_{OE}(k) = \frac{d\hat{y}(k)}{d\boldsymbol{\theta}}. \tag{5.3}$$

However the received values of the FIM in Table 5.2 show a strong influence of some parameters on the crossbar velocity $\dot{\zeta}$ and just a weak influence on the velocity $\dot{\delta}_M$ of the third mass m_M of the oscillator, which is not represented in the white-box model. From the FIM values concerning the additional affine term parameters, it can be recognized that they mainly influence the upper two masses m_C and m_H, which describing the movement of the collector head and the crossbar and also have very weak influence on the lowest mass m_M.

5.1.2 Stability Considerations

A short discussion regarding the stability of the developed pantograph models is carried out here. As mentioned earlier (see Section 3.6.3), there are currently no stability tests available for the combination of the applied methods (LLMN in state-space configuration). It is mentioned here, that on the one hand throughout the development of the two presented models no issues regarding instability during the blending of the LLMs was witnessed, when implementing the one-dimensional input space partitioning. This behaviour stems seemingly from the consistent structure of the state-space systems inside the LLMs, where physically interpretable parameters of similar magnitude were blended. Additionally all the local state-space systems inside the LLMs were modeled stable. Therefore some kind of stability preserving behaviour can be attested to this particular model.
On the other hand, the goal of applying the parameter blending method to the N4SID based pantograph LLMN could not be reached with a high level of satisfaction, as still signs of soaring can be detected in Figure 4.28. Although all possibilities of making the identified matrices as compatible and similar in their structure as possible, by exploiting the canonical representation of the state-space system (modal form).

5.1.3 Discussion on Model Simplifications

The white box model is designed in such a way, that the input contact position η at the top of the pantograph acts like a solid wall, realizing a forced oscillation of the pantograph. Furthermore the connection between the pantograph and the overhead line is as described in Section 2.1 modeled by a spring element, therefore enables the model to map positive and negative values of the contact force F_p. This is not corresponding to the behaviour of a real-world pantograph, where the pan-head would lift off the contact line instead of realizing a negative contact force. This case is not considered in the white-box model nor in the pantograph LLMNs, because in this approaches solely the dynamic of the pantograph was of interest. At points where the pantograph lifts off the contact line, the models do not map the considered signals correctly.

The white box model and subsequently the developed pantograph LLMN models do not consider any friction effects, which are apparent in early pantograph test bench measurement data (hysteresis effects). It is expected that the mapping of real-world pantograph data will suffer in accuracy from that model simplification.

5.1.4 Subspace Identification using N4SID

The results of the case study presented in Section 4.5 Figure 4.20 show, that the Matlab function `n4sid()`, with the estimation of the initial value and the disturbance model switched off (see Section 3.5.3), has great difficulties identifying suitable state-space system matrices $\mathbf{A}_T$, $\mathbf{B}_T$, $\mathbf{C}_T$ and $\mathbf{D}_T$ if the system order is chosen as $n = 6$. This is an interesting result, considering that the system dimension of the underlying system is known to be of that order (see Section 2.1). It shows furthermore that if the subspace identified systems get reduced by the Matlab function `reduce()` to that order, the achievable simulation results are nearly as good (loss of about 1 [%] of FIT) as the ones of an arbitrary higher order system. Therefore it can be observed, that in the chosen configuration (initial values and disturbance model switched off), the N4SID requires a significant higher order system to identify the matrices of the underlying system (up to a similarity transformation, see Section 3.5) accurately. Furthermore the case study in Figure 4.20 indicates that this phenomenon depends on the operating height of the pantograph. E.g. the identifications in the lower region (OP25, compare (4.1)) require a significant larger system dimension than the ones carried out with other operation point data sets. For example the identification run in OP65 shows good results with the system order $n = 13$ of the N4SID system, while in OP25 the same performance regarding the FIT can be first achieved with an system order of $n = 21$.

Remark 5.1.4. As mentioned earlier (see Remark of Figure 4.20 in Section 4.5) the choice of the initial value, i.e. using another part of the training data set for identification, also alters the performance of the N4SID. So it is mentioned here, that the results presented in this case study for a certain operating point were all realized with exactly the same part of the utilized data sets, i.e. the case study does not show the effects of a variation of the initial values, but only of the system order.

5.1.5 Potential Improvements, Further Research

The local linear model network was chosen for its strengths in interpretation capabilities and improving the developed models with regard to accuracy could be achieved by altering the structure of the LLMN. Furthermore another network structure could be chosen (e.g. a RBF network), but these efforts may however diminish physical interpretability of the model which is counterproductive if the aim is to gain knowledge about the underlying process.

The positioning and validity region of the MSF parameters *center* and *spread* could be fine tuned according to the mapping of the static values in Figure 2.4, especially at the upper and lower end of the partition space. The optimization of these parameters, which is referred to as *Splitting Ratio Optimization* as discussed in [19, page 376, Section 13.3.4], could yield promising results for simple model structures with few rules, which is the case for the pantograph model. It should be possible to alter the splitting ratio, with adapting widths of the validity functions (danger of normalization side effects), due to expert knowledge by studying the edge regions of the contact position (input) as depicted in Figure 2.4. Furthermore simply more than 4 LLMs could be implemented, i.e. a finer grid partitioning could be carried out.

Special attention could be given to the proportionality factor k_σ (see Section 3.3), which is often used as a tuning parameter and is simply set to $frac13$ in this thesis according to the recommendation in [19, page 365, equ. (13.37)] (rule of thumb). However as discussed in [19, page 374, Section 13.3.3] the *smoothness optimization* (nonlinear optimization of the proportionality factor k_σ) in general does not yield satisfactory results. Summarizing this is due to the observation that when using a global optimization approach for this parameter, it tends to become bigger, while when using a local optimization approach the smoothness parameter is resulting close to zero, comparable to the effects in the optimization of the rule premise and consequent parameters for fuzzy systems (bias/variance tradeoff, model flexibility/interpretability tradeoff).

The local linear neuro-fuzzy model can cope with a arbitrary dimensional partition space, so incorporating another signal (input, fed back output) to the partition space would be realizable. This would create LLMs which are positioned in a multidimensional partition space, where new rule sets that consider additional information could be defined. For this multidimensional partition space there exist e.g. axis-orthogonal or axis-oblique (*hinging hyperplanes* see e.g. [19, page 438, Section 14.8.1]) decomposition algorithms that provide great freedom from a modeling point of view. The obvious disadvantages are on the one hand the increasing impact of the *curse of dimensionality* and on the other hand the vanishing interpretability capabilities (hyperplanes). In case of the nonlinear pantograph with the given specifications (see Section 1.2.1) no vast improvement is expected. This is due to the fact, that the pantograph model is defined in this thesis as a one-dimensional model and all the available signals - positions, velocities and forces - are connected inside the state-

SWOT analysis for the developed pantograph LLMNs	
Strengths	Interpretability capabilities, computational speed (real-time capable models), ability to map the nonlinear behaviour of the pantograph white-box model
Weaknesses	Mapping accuracy (errors in magnitude and phase), variance-error: remaining uncertainty due to data-driven identification (danger of weak local optima)
Opportunities	Further structure optimization possible, real-time-capable pantograph model applicable for the pantograph-catenary co-simulationm in connection with a modern day control scheme, possible online adaptation of the identification to develop a nonlinear *dynamic* model, utilizable for pantograph controller design
Threats	No guarantee for stability of the pantograph LLMN (surrogate) and (n4sid), therefore unexpected loss of stability cannot be precluded

Table 5.3: SWOT analysis of the developed pantograph LLMNs.

space system. Possibly such a model would yield a higher accuracy, but in this thesis the advantages of the simpler model (faster, better interpretability) yield the better deal.

5.2 Discussion

The final section concludes this diploma thesis by summarizing the presented findings for the pantograph local linear model networks. First an evaluation of the developed pantograph LLMNs is carried out utilizing the instrument referred to as SWOT analysis, see Table 5.3, known from enterprise and project analysis. Furthermore a confrontation of the two pantograph LLMNs (surrogate and n4sid), crossing out their specific advantages and disadvantages is given in Table 5.4.

The aim was to identify a real-time capable pantograph model based on a specified structure. This basic pantograph model should then be used as a static model for further implementation in a superordinated problem, a co-simulation or a control problem (controller design), where the interaction and coupling with a overhead line is of primary interest. This thesis provides two different configurations of such pantograph models with differing properties (interpretability, performance) that show promising behaviour over the defined operating range of the nonlinear pantograph (in regard to stability and accuracy). Therefore these models are qualified to be utilized in further tasks (co-simulation and control problems).

An intense analysis of an existing white-box model was carried out in this thesis, on the one hand supplying information of the nonlinear pantograph dynamics, and on the other hand providing a stencil for approaching pantograph modeling in general. This acquired knowledge can be used as basis for a different modeling approach.

Confrontation of the developed pantograph LLMNs		
	Pantograph LLMN (**surrogate**)	Pantograph LLMN (**n4sid**)
Advantages	High amount of expert knowledge can be incorporated (system structure)	Only input-output data is needed (training data set), nearly no prior knowledge required (system structure)
	Physical interpretability of parameters/states of the model	State-space systems are computed directly from input/output data
Disadvantages	Tediously Output-error optimization with possible unidentifiable parameters and danger of ending up in a weak local optima	Identified system matrices can only cope with zero-meaned signals, therefore a look-up table in connection with a low-pass filter has to be incorporated as a quasi affine term, which is bad for the computational speed
		No state consistency given over the LLMs
		No interpretability capabilities of the state-space system matrices nor state vector

Table 5.4: Confrontation of the developed pantograph LLMNs revealing specific advantages and disadvantages of the different approaches.

5.3 Main Statements, Conclusion

The conclusion of this thesis will be formulated in two parts, on the one hand outlining the findings from a **methodical point of view** and on the other hand from an **practical point of view**. Again the fact is stressed here, that this thesis examined an existing white-box model of the nonlinear pantograph and therefore represents a model based study. Drawing conclusions to the behaviour of a real-world pantograph should be done cautiously (e.g. recognizing that due to the simplifications of the white-box model no influence of friction on the mechanism is considered).

The main goal of this thesis was the application of a local linear model network (local linear neuro-fuzzy system) in state-space configuration to the nonlinear pantograph modeling problem, implementing one-dimensional input partitioning in combination with a parameter or output blending method and therefore creating a real-time capable pantograph model. Two types of models were developed, one based on a mechanical surrogate model (three-mass oscillator) and one based on subspace identification methods (N4SID) incorporating different blending methods (parameter respectively output blending). The offset correction was realized either using the affine term state-space formulation (case surrogate) or a low-pass filter in combination with a look-up table respectively (case n4sid).

5.3.1 Conclusion from a methodical point of view (scientist)

Ad pantograph LLMN (surrogate): The incorporation of expert knowledge for the design of the pantograph LLMN (surrogate) based on a mechanical surrogate model (three-mass oscillator) was carried out extensively as envisioned. The interpretation capabilities of this model are seen as its primary strength, whereas the performance with regard to the achievable accuracy is seen as its primary weakness. Especially the inability to identify a suitable parameter set which would give the model the ability to map the Eigenmodes detected in the white-box model data correctly in the applied model configuration has to be lined out as a main draw back. At this point it is not conclusive, if such a parameter set exists for this type of model (i.e. if a global optimum exists), or if the model structure is not appropriate for the problem. It can however be stated, that the optimization is vulnerable of reaching a local optima and sensitive to the initial values of the parameter vector (starting point). Furthermore it can be stated, that on the one hand the collector head model (linear one-mass oscillator) is modeled identical for the white-box model and the pantograph LLMN (surrogate) and on the other hand the three masses of the surrogate model theoretical provide the ability to map three modes. The allocation of the poles of the pantograph LLMN (surrogate) system matrices in comparison with the linearized white-box model underline a different dynamic behaviour.

Ad pantograph LLMN (n4sid): Concerning the applied methodologies for this model it has to be stated, that the endeavor of implementing an open-loop subspace identification method (N4SID) into a local-linear modeling structure still provides questions but also potential. It was shown, that a stable and smooth parameter blending procedure can be realized, if the identified state-space matrices are transformed to modal form, where additional awareness is given to the positioning of the modes and corresponding signs is given. The automation of this task however is not solved by this thesis and will need further attention and research. Furthermore the training or identification process, when the subspace identification is utilized, was realized through somewhat a grid-based approach. Therefore in its current implementation the pantograph LLMN (surrogate) cannot be directly identified for multiple LLMs automatically. Another issue is the experienced seemingly arbitrary performance of subspace identified models due to the lack of the estimation of an initial value (possible in general, but not appropriate for the given modeling problem). However, the achievable mapping accuracy of a nonlinear mechanism utilizing the LLMN based on subspace identification methods is impressive.

5.3.2 Conclusion from a practical point of view (engineer)

The conclusion from a practical point of view is formulated as an evaluation of the developed pantograph models with regard to the **specifications** defined in Section 1.2.1.

Computational efficiency

Both developed pantograph LLMNs (surrogate and n4sid) are distinct real-time capable models as documented in Table 4.4. The pantograph LLMN (n4sid) is due to the applied preprocessing of the input signals and postprocessing of the output signals, which is carried out in every time-step, significantly slower than the pantograph LLMN (surrogate).

Structure

Both developed pantograph LLMNs (surrogate and n4sid) are realized as local linear model networks, consisting of an arbitrary amount of local linear models (LLMs). The validity region for those LLMs is chosen equidistant over the operating height and they are blended via an one-dimensional partitioning variable (contact position η) utilizing either the parameter blending method (pantograph LLMN (surrogate)) or the output blending method (pantograph LLMN (n4sid)), therefore providing strong capabilities for physical interpretation.

Mathematical formulation

Both developed pantograph LLMNs (surrogate and n4sid) contain state-space systems in their respective LLMs with a system order of $n = 6$.

Ad pantograph LLMN (surrogate): In case of the pantograph LLMN (surrogate) these systems are given by the state-space formulation of the equations of motion of a parametrized three-mass oscillator, whose parameters where determined with an output error optimization method utilizing the reference data of the pantograph white-box model. To cope with occurring offset values in the positions, the state-space systems are extended by affine terms. Due to the parameter blending method, uniqueness of the state-vector is given. The blended system, as well as the state-vector are physically interpretable.

Ad pantograph LLMN (n4sid): In case of the pantograph LLMN (n4sid), the state-space systems are reduced state-space systems, originally identified by the application of the subspace identification method N4SID on input/output data of the pantograph white-box model. To cope with occurring offset values in the positions, a preprocessing of the inputs and a postprocessing of the outputs is carried out in every time step. Due to the applied output blending method, no uniqueness of the state-vector is given. The systems and states are not physically interpretable, but similar in their structure due to the manipulations of the identified system matrices.

Performance

For a detailed comparison of the developed models mapping performance and accuracy consult Sections 4.4.1 and 4.5.1, as well as Table 4.5 and especially Figure 4.35.

Ad pantograph LLMN (surrogate): As documented the strength of this pantograph model lies in its physical interpretability, where all system matrices as well as the state vector of the blended state-space system (obtained by the implementation of the parameter blending method) correspond to physical quantities. The inability of identifying the modes (resonant frequencies) of the pantograph white-box model (reference model) has to be seen as the main drawback of this approach implementing the described model structure (surrogate model as a three-mass oscillator). This shortcoming is detectable by comparing the positioning of the poles of the utilized state-space systems as well as the frequency analysis to the output signals to those of the pantograph white-box model. However due to its outstanding computational speed and its satisfactory accuracy in mapping the contact force F_p over the whole operating range, the pantograph LLMN (surrogate) could seen as an alternative to a global linear model.

Ad pantograph LLMN (n4sid): The approach of incorporating a subspace identification method into a local linear model structure can be seen as a success. The pantograph model is identifiable with considerably little effort, provided the required measurements are available. Due to this property, the model could be utilized for the examination of different pantograph geometries. The subspace modeling approach still withholds some potential, on the one hand in the structure of the identified matrices (compatibility) and on the other hand in the training process of the pantograph LLMN (n4sid), i.e. automatic identification of a multiple LLM LLMN or even online adaptation. The presented solution utilizing a low-pass filter and look-up table is costly both in regard to the effort necessary for an identification of the model and to the computational speed of the identified model. Nevertheless the already achieved accuracy over the whole operating range of the pantograph LLMN (n4sid) is impressive.

Bibliography

[1] Guilherme Aschauer. Nonlinear Pantograph. Diploma Thesis, Technical University of Vienna, Wien, 2014.

[2] Jacques Cathala Balan-Mire and Pascale Forte. Device For Adjusting The Bearing Force Of A Pantograph On A Catenary Wire And Process Relating Thereto. Patent, May 1992. US005115405A.

[3] Jesus Benet, Angelines Alberto, Enrique Arias, and Tomas Rojo. A Mathematical Model of the Pantograph-Catenary Dynamic Interaction with Several Contact Wires. *IAENG International Journal of Applied Mathematics*, 37(2), 2007.

[4] Stefano Bruni, Giuseppe Bucca, Andrea Collina, and Alan Facchinetti. Numerical and Hardware-In-the-Loop Tools for the Design of Very High Speed Pantograph-Catenary Systems. *Journal of Computational and Nonlinear Dynamics*, 7, 2012.

[5] Alessandro Chiuso and Giorgio Picci. Consitency analysis of certain closed-loop subspce identification methods. *Automatica*, 41(3):377–391, 2005.

[6] Chun Tung Chou and Michel Verhaegen. Subspace algorithms for the identification of multivariable dynamic errors-in-variables models. *Automaticy*, 33(10):1857–1869, 1997.

[7] Gang Feng. *Analysis and Synthesis of Fuzzy Control Systems, A Model-Based Approach*. Automation and Control Engineering Series. CRC Press, Boca Raton, 2010.

[8] Pascal Gahinet, Arkadii Nemirovskii, Alan Laub, and Mahmoud Chilali. The MATLAB LMI Control Toolbox User's Guide. *The MathWorks Inc., Natick, MA*, Version 1.0, 1995.

[9] Graham Goodwin and Robert Payne. *Dynamic System Identification: Experiment Design and Data Analysis*. Academic Press, 1977.

[10] Christoph Hametner, Markus Stadlbauer, Maxime Deregnaucourt, Stefan Jakubek, and Thomas Winsel. Optimal experiment design based on local model networks and multilayer perceptron networks. *Engineering Applications of Artificial Intelligence*, 26:251–261, 2013.

[11] Kenneth J. Hunt, Roland Haas, and Roderick Murray-Smith. Extending the functional equivalence of radial basis functions networks and fuzzy inference systems. *IEEE Transactions on Neural Networks*, 7(3):776–781, 1996.

[12] Stefan Jakubek and Alexander Schirrer. *Adaptive and Predictive Control*. Institute for Mechanics and Mechatronics, Vienna University of Technology, Wien, 2013.

[13] Magnus Jansson. Subspace identification and ARX modeling. *Preprints of the IFAC Symposium on System Identification (SYSID)*, Oxford: Elsevier Science Ltd, Rotterdam:1625–1630, 2003.

[14] Shahin Hedayati Kia, Fabio Bartolini, Augustin Mpanda-Mabwe, and Roger Ceschi. Pantograph-Catenary Interaction Model Comparison. *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*, 1:1584–1589, 2010.

[15] Bart Kosko. Fuzzy Systems as Universal Approximators. *IEEE Transactions on Computers*, 43(11):1329–1333, 1994.

[16] Wallace Larimore. Canonical Variate Analysis in Identification, Filtering and Adaptive Control. *Proceedings of the 29th IEEE Conference on Decision and Control*, :596–601, 1990.

[17] Aleksandrs Matvejevs and Andrejs Matvejevs. Pantograph-Catenary System Modeling Using MATLAB-Simulink Algorithms. *Schientific Journal of Riga Technical University*, 44:38–44, 2010.

[18] Roderick Murray-Smith and Tor Arne Johansen. *Multiple Model Approaches to Modelling and Control*. Taylor & Francis, New York, 1997.

[19] Oliver Nelles. *Nonlinear System Identification, From Classical Approaches to Neural Networks and Fuzzy Models*. Springer-Verlag, Berlin Heidelberg, 2001.

[20] Oliver Nelles, Fink, and Isermann. Local linear model trees (lolimot) toolbox for nonlinear system identification. *IFAC Symposium on System Idenification (SYSID)*, 0, June 2000.

[21] Oliver Nelles and M Fischer. Local linear model trees (LOLIMOT) for nonlinear system identification of a cooling blast. *European Congress on Intelligent Techniques and Soft Computing (EUFIT)*, 0:1187–1191, September 1996.

[22] Bernd-Markus Pfeiffer and Rolf Isermann. Criteria for Successful Applications of Fuzzy Control. *Engineering Applications of Artificial Intelligence*, 7(3):245–253, 1994.

[23] Bernd-Markus Pfeiffer, Jens Jäkel, Andreas Kroll, Christian Kuhn, Helge-Björn Kuntze, Ulrich Lehmann, Timo Slawinski, and Volker Tews. Erfolgreiche Anwendung von Fuzzy-Logik und Fuzzy Control. *at - Automatisierungstechnink*, 50(10):461–471, 2002.

[24] Bernd-Markus Pfeiffer, Jens Jäkel, Andreas Kroll, Christian Kuhn, Helge-Björn Kuntze, Ulrich Lehmann, Timo Slawinski, and Volker Tews. Erfolgreiche Anwendung von Fuzzy-Logik und Fuzzy Control. *at - Automatisierungstechnink*, 50(11):511–521, 2002.

[25] Joe Qin. An overview of subspace identification. *Elsevier, Computers and Chemical Engineering*, 30:1502–1513, 2006.

[26] Joe Qin and Lennart Ljung. Closed-loop subspace identification with innovation estimation. *Preprints of the IFAC Symposium on System Identification (SYSID)*, Oxford: Elsevier Science Ltd, Rotterdam:887–892, 2003.

[27] Frederico Grases Rauter, Joao Pombo, Jorge Ambrosio, and Manuel Pereira. Multibody Modeling of Pantographs for Pantograph-Catenary Interaction. *IUTAM Symposium on Multiscale Problems in Multibody System Contacts*, 1:205–226, 2007.

[28] Alexander Schirrer. Efficient robust control design and optimization methods for flight control. Doctoral Thesis, Technical University of Vienna, Wien, 2011.

[29] Alexander Schirrer and Stefan Jakubek. Hochdynamische Oberleitungs-Emulation eines Pantographenprüfstandes: Modellierungs- und Regelkonzepte (engl. High-dynamic catenary emulation of a pantograph test rig: modeling and control concepts). Talk at 49th control colloquium in Boppard, Germany, March 2015.

[30] Torsten Söderström and Petre Stoica. *Instrumental Variable Methods for System Identification*. Springer-Verlag, New York, 1983.

[31] Kazuo Tanaka and Michio Sugeno. Stability analysis and design of fuzzy control systems. *Fuzzy Sets and Systems*, 45(2):135–156, 1992.

[32] Kazuo Tanaka and Hua Wang. *Fuzzy Control Sytems Design and Analysis, A Linear Matrix Inequality Approach*. John Wiley& Sons, Inc, New York, 2001.

[33] Peter van Overschee and Bart de Moor. N4SID: Subspace Algorithms for the Identification of Combined Deterministic-Stochastic Systems. *Automatica*, ESAT/SISTA 1992-34-I, 1992.

[34] Peter van Overschee and Bart de Moor. A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12):1853–1864, 1995.

[35] Peter van Overschee and Bart de Moor. *Subspace Identification for linear systems*. Kluwer Academic Publishers, Leuven, 1996.

[36] Michel Verhaegen. Subspace model identification part 3. Analysis of the ordinary output-error state-space model identification algorithm. *International Journal of Control*, 56(3):555–586, 1993.

[37] Michel Verhaegen. Identification of the deterministic part of MIMO state-space models given in innovations form from input-output data. *Automatica*, 30(1):61–74, 1994.

[38] Michel Verhaegen and Patrick Dewilde. Subspace model identification part 1. The output-error state-space model identification class of algorithms. *International Journal of Control*, 56(5):1187–1210, 1992.

[39] Michel Verhaegen and Vincent Verdult. *Filtering and System Identification, A Least Squares Approach*. Camebridge University Press, Camebridge, first edition, 2007.

[40] Simon Walters. Simulation of Fuzzy Control Applied to a Railway Pantograph-Catenary System. *Knowledge-Based and Intelligent information and Engineering Systems*, 6277:322–330, 2010.

[41] Li-Xin Wang and Jerry M. Mendel. Fuzzy Basis Functions, Universal Approximation, and Orthogonal Least-Squares Learning. *IEEE Transactions on Neural Networks*, 3(5):807–814, 1992.

[42] T. X. Wu and M. J. Brennan. Basic Analytical Study of Pantograph-catenary System Dynamics. *Vehicle System Dynamics*, 30:443–456, 1998.

[43] Lin Xie, Serge Shishkin, and Minyue Fu. Piecewise Lyapunov functions for robust stability of linear time-varying systems. *Systems & Control Letters*, 31:165–171, 1997.

[44] Lotfi A. Zadeh. Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. *IEEE Trans. Systems, Man and Cybernetics*, 3:28–44, 1973.