Faculty of Aerospace Engineering

Maximum Likelihood Estimation of Linear Time-Varying Pilot-Vehicle System Parameters

M. Kers

July 22, 2012

TU Delft Delft University of Technology

Challenge the future

Maximum Likelihood Estimation of Linear Time-Varying Pilot-Vehicle System Parameters

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering at Delft University of Technology

M. Kers

July 22, 2012

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

Delft University Of Technology
Department Of
Control and Simulation

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **"Maximum Likelihood Estimation of Linear Time-Varying Pilot-Vehicle System Parameters"** by **M. Kers** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: July 22, 2012

Readers:

prof.dr.ir. M. Mulder

dr.ir. E. van Kampen

dr.ir. M. Mulder

ir. D. M. Pool

Contents

List of Symbols

Greek Symbols

α	Line-Search Parameter	
δ_e	Elevator Deflection	[deg]
ϵ	Prediction Error	[deg]
ζ_{nm}	Neuromuscular Damping Ratio	[deg]
θ	Parameter	
θ_p	Parameter Variance Percentage	[%]
μ	Mean	
σ	Standard Deviation	
τ_m	Motion-Perception Time Delay	[s]
τ_v	Pilot Visual-Perception Time Delay	[s]
ϕ_d	Disturbance Sinusoid Phase Shift	[rad]
ϕ_m	Phase Margin	[deg]
ϕ_t	Target Sinusoid Phase Shift Frequency	[rad]
ω	Frequency	[rad/s]
ω_d	Disturbance Sinusoid Frequency	[rad/s]
ω_m	Measurement Base Frequency	[rad/s]
ω_{nm}	Neuromuscular Frequency	[rad/s]
ω_t	Target Sinusoid Frequency	[rad/s]

Θ	Parameter Vector	

Roman Symbols

a	Matrix Entry	
e	Tracking Error Signal	[deg]
f_d	Disturbance Forcing Function	[deg]
f_t	Target Forcing Function	[deg]
h	Padé Approximation Coefficient	
h^*	Interval Size	
i	i-th Row of a Matrix	
j^*	j-th Row of a Matrix	
k	Number of Discrete Instants	
k_{rk}	Runge-Kutta Predictor	
m	Number of Measurements	
n	Measurement Noise Signal, Pilot Remnant Signal	[deg]
n_p	Pilot Remnant Signal Variance Percentage	[%]
n_t	Target Forcing Function Integer Factor	
s	Laplace Variable	
t	Time	[s]
t_{half}	Sigmoid Halfway Time	[s]
u	Pilot Control Signal	[deg]
w	Gaussian White Noise Signal	[deg]
x	State Signal	
y	System Output Signal	[deg]
A	State Matrix	
A_d	Disturbance Sinusoid Amplitude	[deg]
A_t	Target Sinusoid Amplitude	[deg]
B	Input Matrix	
C	Output Matrix	
D	Feedthrough Matrix	
H_c	Vehicle Dynamics	
H_{nm}	Pilot Neuromuscular Dynamics	

H_n	Pilot Remnant Filter	
H_{ol}	Open-Loop Dynamics	
H_p	Pilot Response	
H_{pade}	Padé Approximant	
H_{pe}	Error Frequency Response Function	
$H_{p\theta}$	Pilot Motion Frequency Response Function	
H_{θ,δ_e}	Controlled Pitch Dynamics	
H_{px}	State Frequency Response Function	
H_{sc}	Semicircular-Canal Dynamics	
K	Gain	
K_c	Vehicle Dynamics Gain	
$K_{\delta_e,u}$	Pitch Stick Gain	
K_m	Motion-Perception Gain	
K_n	Pilot Remnant Gain	
K_v	Pilot Visual-Perception Gain	
L	Likelihood Function	
$M_{\Theta\Theta}$	Fisher Information Matrix	
P_n	Remnant-to-Control-Signal Power Ratio	
Q	Covariance Matrix	
S	Sensitivity Matrix	
T_c	Vehicle Dynamics Time Constant	[s]
T_{lag}	Visual-Lag Time Constant	[s]
T_{lead}	Pilot Visual-Lead Time Constant	[s]
T_m	Measurement Time	[s]
T_n	Pilot Remnant Time Constant	[s]
T_{sc}	Semicircular-Canal Time Constant	[s]
X	Laplace Transform of the Input	

Abbreviations

ARX	Autoregressive Exogenous
BEM	Basis Expansion Modeling
BPDN	Basis Pursuit Denoising
C&S	Control and Simulation
CRLB	Cramér-Rao Lower Bound
DUT	Delft University of Technology
EKF	Extended Kalman Filter
EnKF	Ensemble Kalman Filter
FC	Fourier Coefficients
FIM	Fisher Information Matrix
GA	Genetic Algorithm
GN	Gauss-Newton
KF	Kalman Filter
LASSO	Least Absolute Shrinkage and Selection Operator
LDS	Linear Dynamical System
LMS	Least Mean Squares
LPM	Local Polynomial Modeling
LPV	Linear Parameter-Varying
LSE	Least Squares Estimation
LTI	Linear Time-Invariant
LTV	Linear Time-Varying
MLE	Maximum Likelihood Estimation
MSE	Mean Square Error
NMS	Neuromuscular System

PC	Polynomial Chaos
PDF	Probability Density Function
PE	Persistent Excitation
PSD	Power Spectral Density
RLS	Recursive Least Squares
SRS	SIMONA Research Simulator
TVARX	Time-Varying Autoregressive Exogenous
VAF	Variance Accounted For
VBS	Variable Bandwidth Selection
WF	Wiener Filter
WLS	Weighted Least Squares

Preface

This thesis is the final work submitted to Delft University of Technology (DUT) in part satisfaction of the requirements for the Master of Science Degree in Aerospace Engineering. During my first year as a graduate student at the Control & Simulation Department of our university, several courses fueled my interest for topics related to system identification in human-machine systems. Although I had only touched the surface of system identification theory by the time professor Max Mulder and I started to conceptualize my research topic, my existing interest in system identification made it no surprise that I ultimately chose to continue the Department's research on estimation methods.

For the past year or so, I have had the opportunity to immerse myself in the extremely interesting topics of human control behavior and system identification on an academic level. Constructing models from observed data can be challenging, especially when the system you are modeling contains a nonlinear human operator. From the thrills of reading the wildest system identification techniques during the literature research phase to the most unimaginative instances of waiting for the estimation results to be computed: performing fundamental scientific research in this area has it all. Now that the end is in sight, I can wholeheartedly say that it was an exciting journey with its euphoric ups and depressing downs on the cutting-edge of science and technology. The thesis in front of you is the result of this journey. It consists of a preliminary thesis and additional final thesis chapters, which are concisely summarized in the included scientific paper.

The research, which ultimately led to this thesis, involved numerous exchanges of thoughts with a variety of people. First, and most of all, I would like to thank Daan Pool for his contributions to my research. Not only did he push my boundaries at the right moment, but he also helped me develop an extensive toolkit for doing academic research. Daan, I admire your clear way of thinking and I am thankful that you were always available for questions or discussions. Without your guidance and patience throughout the process it would not have been possible to get to this result. I would also like to express my gratitude to Qiping Chu for sharing his scientific and mathematical knowledge in the field of control

systems engineering. Because of your enthusiasm about the research and the inspiring tales about your life, I often left your office full of energy and with new insights. I would also like to mention René van Paassen and Max Mulder and thank them for their valuable coaching and guidance at crucial moments during the process. Other people who have helped me during my research and deserve to be mentioned are Peter Zaal, Tomas McKelvey, Roland Tóth, and Roeloef Koekoek. I would also like to thank Joris Bisschop, Thijs Franken, Koen Rutten, and Erik-Jan van Kampen for their constructive suggestions on my writings.

I have dedicated this thesis to my father and my mother, Bram and Kusuma. Over the years, their unconditional love and support have helped me to persist through good and bad times. I am grateful for the wisdom and the freedom they have given me to develop myself in various aspects of my life. After my parents, I would like to mention my brother David and my best friends Thijs, Joris, Welmoed, and Maurits, I love you guys! Throughout my student life I have made many more friends who contributed to who I am today. My final thanks, therefore, go out to the friends I made in Delft, in Leiden, on the boards and commissions I had the honor to be part of, and the friends I made in Taiwan, mainland China, and the United States of America.

Finally, I want to say that I am proud of the work that lies in front of you. I imagine that the literature research used in my preliminary thesis can be a good start for anybody who wants to get a hold of the fundamentals of system identification and pilot-vehicle control behavior. The results of the study as documented in the paper and the final thesis chapters fit in the larger framework of the contemporary research that is being done on estimation techniques for pilot-vehicle systems. Please do not hesitate to contact me if I can help clarify any questions you have. Happy reading!

Delft, July 2012, Martin Kers

martinkers@gmail.com

Part I

Paper

Maximum Likelihood Estimation of Linear Time-Varying Pilot-Vehicle System Parameters

M. Kers,* D. M. Pool,† Q. P. Chu,‡ M. M. van Paassen,§ and M. Mulder¶
Delft University of Technology, P.O. Box 5058, 2600 GB Delft, The Netherlands

In this paper a time-domain identification procedure based on maximum likelihood estimation is introduced for the estimation of time-varying pilot-vehicle system parameters. The parameter estimation is performed using simulated data of a control task containing human operator and vehicle dynamics in which the time-varying parameters are modeled with Boltzmann sigmoids. Simulated data is generated for different amounts of human operator remnant. During the estimation, different sets of initial parameter values are used. The proposed method returns accurate results when no human operator remnant is present and when the initial parameter values are chosen close to the values used to set up the simulation. The results, however, indicate that the proposed method is sensitive to the presence of remnant, but even more sensitive to the choice of initial parameter values. Additionally, the results show that the procedure is influenced by local minima. The proposed method is promising, but to deal with the mentioned issues it is necessary to adapt the estimation method in future research.

Nomenclature

A_t	target sinusoid amplitude, deg
e	tracking error signal, deg
f_t	target forcing function, deg
H_c	vehicle dynamics
H_n	human operator remnant shaping filter
H_{ol}	open-loop dynamics
H_{nm}	human operator neuromuscular dynamics
H_p	human operator pilot dynamics
j	imaginary unit
K_c	vehicle dynamics gain
K_n	shaping filter gain
K_v	human operator visual-perception gain
k	number of discrete instants
L	likelihood function
$M_{\Theta\Theta}$	Fisher information matrix
m	number of measurements
n	human operator remnant signal, deg
n_p	remnant variance percentage, %
n_t	target forcing function integer factor
Q	covariance, deg^2
s	Laplace operator
T_c	vehicle dynamics time constant, s
T_l	human operator visual-lead time constant, s
T_m	measurement time, s
T_n	shaping filter time constant, s
t	time, s
t_1	begin time of Boltzmann sigmoid, s
t_2	end time of Boltzmann sigmoid, s
t_{half}	Boltzmann sigmoid halfway time, s
t_i	time step, s
u	human operator control signal, deg
$\hat{u}$	estimated human operator control signal, deg
w	white noise signal, deg
y	system output signal, deg

Symbols

α	line-search parameter
ϵ	prediction error, deg

*MSc Student, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058, 2600 GB Delft, The Netherlands; m.kers@tudelft.nl.

†PhD Student, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058, 2600 GB Delft, The Netherlands; d.m.pool@tudelft.nl. Member AIAA.

‡Associate Professor, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058, 2600 GB Delft, The Netherlands; q.p.chu@tudelft.nl. Member AIAA.

§Associate Professor, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058, 2600 GB Delft, The Netherlands; m.m.vanpaassen@tudelft.nl. Member AIAA.

¶Professor, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058, 2600 GB Delft, The Netherlands; m.mulder@tudelft.nl. Member AIAA.

ζ_{nm}	neuromuscular damping ratio	σ_n	remnant signal standard deviation, deg
Θ	parameter vector	τ_v	human operator visual-perception time delay, s
$\hat{\Theta}$	parameter estimates vector	ϕ_t	target sinusoid phase shift, rad
$\hat{\Theta}_{ML}$	maximum likelihood parameter estimates vector	ω	frequency, rad/s
θ	parameter	ω_c	crossover frequency, rad/s
$\hat{\theta}$	estimated parameter	ω_m	measurement base frequency, rad/s
θ_0	true parameter	ω_{nm}	neuromuscular frequency, rad/s
θ_p	parameter variance percentage, %	ω_t	target sinusoid frequency, rad/s
σ_u	control signal standard deviation, deg		

I. Introduction

Human manual control behavior is a nonlinear, time-varying, and closed-loop process. The time-varying aspect of this behavior is the primary reason that traditional system identification methods are of limited use for contemporary research in comparison to adaptive methods. Proper system identification can help explain the effects of different perceptual modalities on a human's control behavior through the determination of model parameters, such as weighing gains and time delays, from experimental data. This knowledge can for example be used to design advanced manual control systems for cars, aircraft, and spacecraft.

In the past, human time-invariant manual control methods have been successfully employed.[1–3] Although human manual control behavior is time-varying in nature, much of the knowledge about it still comes from time-invariant analysis. In Ref. 1 a foundation is laid for these time-invariant models by treating the pilot-vehicle system as a closed-loop entity. A new method for identification of human control behavior with linear time-invariant models is proposed in Ref. 2. Ref. 3 proposes an output error method for time-invariant models using Maximum Likelihood Estimation (MLE) augmented with a genetic algorithm. These studies show that the accuracy of such time-invariant models is, however, often affected by biases and the control tasks require highly specific demands.

To reduce the disadvantages associated with time-invariant models, more recent research focuses on time-varying estimation methods.[4–6] A time-varying estimation technique using wavelets is proposed in Ref. 4 for estimating time-variant neuromuscular admittance. In Ref. 5 the wavelet method is compared to a sliding time window MLE method for estimating human operator parameters. The results from this study show that wavelets are more sensitive to measurement noise, but that the proposed MLE method cannot detect fast changes in human control behavior. Lastly, in Ref. 6 two identification methods, an output error method and a sliding window Fast Fourier Transform method, are successfully used for estimation of time-varying human operator parameters.

This paper presents the results of a study on parameter estimation of time-varying pilot model parameters using a MLE method in the time-domain. A human operator (i.e., a pilot) controlling a vehicle will be modeled with time-varying parameters to generate simulated data. The control task performed by the human operator starts with single-integrator dynamics after which a transition to double-integrator dynamics takes place. The transition between the two different dynamics is parametrized with so-called Boltzmann sigmoids. The proposed MLE method is used to estimate the time-varying parameters in the time-domain from the simulated data. In comparison with linear time-invariant and wavelet methods, which both require an additional step in the frequency domain for parameter estimation,[3] MLE only uses a single step in the time-domain, which contributes to a significant reduction of bias and variance.[5] The proposed method differs from earlier MLE approaches in the sense that the parameters are estimated as a function of time.

Finally, the MLE method in this paper will be compared with a sliding time window MLE method as considered in Ref. 5 for analyzing and identifying time-varying pilot dynamics. To be able to compare the two methods, the control task in this study is modeled after the control task in Ref. 5. In accordance with this approach, the following hypothesis will be tested in this paper: *The time-varying time-domain identification procedure as proposed in this paper will return more accurate parameter estimates in comparison with the windowed MLE method proposed in Ref. 5.*

This paper is structured as follows. In Sec. II the control task model used to generate simulation data is explained. The theory on MLE, the proposed MLE method, and the sliding time window method of Ref. 5 will be discussed in Sec. III. The results of the parameter estimation will be given in Sec. IV, followed by a discussion of the results in Sec. V. Finally, Sec. VI gives an overview of the conclusions of this study.

II. Simulation Setup

For this study, the choice is made to use simulated data instead of experimental data, because the values that will be estimated can be set for a control task that is used to generate such simulated data. The control task and its corresponding dynamics are described in this section.

II.A. Simulated Control Task

In this paper, the problem of identifying time-varying pilot dynamics is considered for a simulated manual control task similar to the one considered in Ref. 5. In Fig. 1 the closed-loop process for a human operator controlling a vehicle is shown. Here the operator is actively controlling the vehicle dynamics H_c, while following a target signal f_t. This forcing function f_t is used for excitation of the system, which is usually a multisine signal in human operator research.[3,5,7,8] The error signal e between the target forcing function f_t and the system output y can for example be displayed on a screen, so that the operator can see the error, minimize it and ultimately optimize his control behavior. The operator's control signal u consists of the human operator response H_p, which is combined with a remnant n to account for nonlinear control behavior of the human operator. The remnant is generated by passing a zero-mean Gaussian white noise signal w through shaping filter H_n. The time-varying aspect of the simulated control task as considered in this paper represents a human controller who is initially controlling a system with single-integrator dynamics when the controlled element dynamics H_c are then changed to double-integrator dynamics.

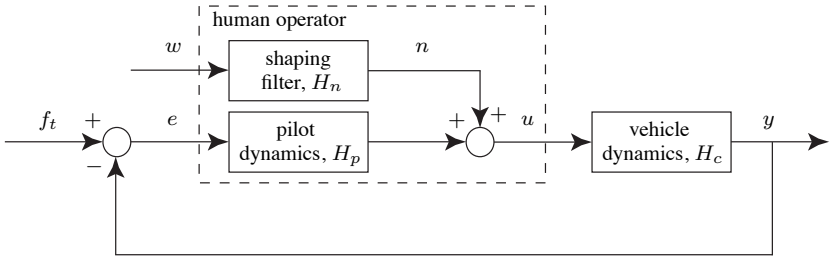


Figure 1. Closed-loop manual control task.

For the vehicle dynamics $H_c(s,t)$ Eq. (1) is used. This equation contains a time-varying gain $K_c(t)$ and a time constant $T_c(t)$.

$$H_c(s,t) = \frac{K_c(t)}{T_c(t)s^2 + s} \tag{1}$$

Below a break frequency of $1/T_c$, the vehicle dynamics act as a single-integrator system ($K_c(t)/s$). Above the break frequency, the vehicle dynamics emulate a double-integrator system ($(K_c(t)/T_c(t))/s^2$). The corresponding human operator response $H_p(s,t)$, which is representative for the change in pilot dynamics that would be required in this case, is defined by Eq. (2) and consists of a time-varying visual-perception gain $K_v(t)$, a time-varying visual-lead time constant $T_l(t)$, a visual-perception time delay constant τ_v and neuromuscular dynamics $H_{nm}(s)$.

$$H_p(s,t) = K_v(t)\,(T_l(t)s + 1)\,e^{-s\tau_v} H_{nm}(s) \tag{2}$$

The neuromuscular dynamics of the human operator are defined in Eq. (3) with neuromuscular frequency ω_{nm} and neuromuscular damping ζ_{nm}.

$$H_{nm}(s) = \frac{\omega_{nm}^2}{s^2 + 2\zeta_{nm}\omega_{nm}s + \omega_{nm}^2} \tag{3}$$

The low-pass shaping filter $H_n(s,t)$ for the zero-mean Gaussian white noise signal w is defined in Eq. (4). In this equation $K_n(t)$ is a time-varying gain and T_n a time constant. The resulting human operator remnant n is added to the output signal of H_p. The shaping filter is used to make sure that the human operator remnant signal variance percentage $n_p = (\sigma_n^2/\sigma_u^2)\cdot 100\%$ for the single-integrator and double-integrator dynamics remain representative of the noise levels typically found in manual control.

$$H_n(s,t) = \frac{K_n(t)}{T_n s + 1} \tag{4}$$

II.B. Model parametrization

It is important for the parameter estimation to be aware that there are different ways of parametrizing the linear time-varying parameters of the model. In this section three different ways of parametrization are described: linear functions, polynomials, and Boltzmann sigmoids. As described in Sec. II.A, the human operator changes his control behavior from single-integrator dynamics to double-integrator dynamics. The time-varying parameters define the transition between these two forms of behavior. Examples of the three forms of parametrization are shown in Fig. 2. These figures all start with constant single-integrator dynamics, which end at t_1. Between t_1 and t_2 the transition with the respective linear, polynomial, or sigmoid function takes place. At t_2 the transition is completed and constant double-integrator dynamics are reached.

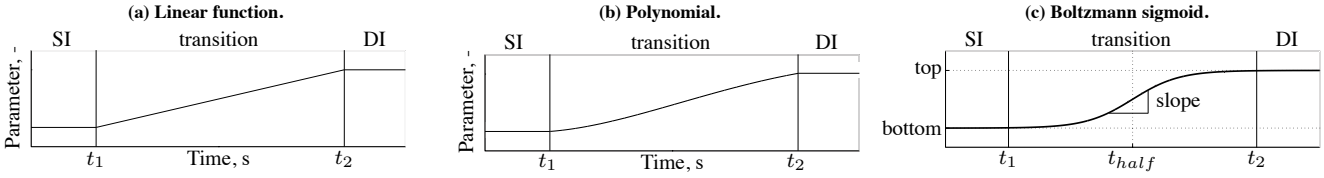


Figure 2. Three different functions are shown in this figure that can be used for parametrization of the linear time-varying parameters. The figures start with constant single-integrator dynamics (SI). After t_1 the transition with the specific function takes place. At t_2 the constant double-integrator dynamics (DI) are reached.

To show how the parametrization of a linear time-varying parameter takes place, gain $K_v(t)$ is used as an example. For a linear function in accordance with Fig. 2a, $K_v(t)$ is parametrized according to Eq. (5). The parameters K_{v_0} and K_{v_1} are constants that define the slope and y-intercept of the linear function, respectively. Note that a linear function is in fact a first-order polynomial. In this case, the total of single-integrator dynamics, double-integrator dynamics, and the linear transition forms a ramp signal.

$$K_v(t) = K_{v_0} + K_{v_1} t \tag{5}$$

The polynomial function plotted in Fig. 2b is an example of a fourth-order polynomial, which is the highest order polynomial used for this research. Second-order and third-order functions are also evaluated in this study to see what the effects of the different orders are on the parameter estimation. A fourth-order polynomial parametrization of $K_v(t)$ is given in Eq. (6) with constants K_{v_0}, K_{v_1}, K_{v_2}, K_{v_3}, and K_{v_4}, which determine the polynomial's shape.

$$K_v(t) = K_{v_0} + K_{v_1} t + K_{v_2} t^2 + K_{v_3} t^3 + K_{v_4} t^4 \tag{6}$$

Finally, the linear time-varying parameters are parametrized with a Boltzmann sigmoid as in Fig. 2c. In Eq. (7) the definition of the Boltzmann sigmoid is stated, together with its parametrization. The choice is made to parametrize $top - bottom$ as one parameter $K^*_{v_0}$ in order to avoid problems with parameter estimation when the $bottom$-value surpasses the top-value. K_{v_1} defines the bottom of the sigmoid, whereas K_{v_2} sets the halfway time t_{half} of the sigmoid, and K_{v_3} fixes the sigmoid's slope.

$$K_v(t) = bottom + \frac{top - bottom}{1 + \exp\left(\frac{t_{half} - t}{slope}\right)} = K_{v_1} + \frac{K^*_{v_0}}{1 + \exp\left(\frac{K_{v_2} - t}{K_{v3}}\right)} \tag{7}$$

For the control task in this paper, parametrization with the linear function and the polynomials turns out to be disadvantageous to parameter estimation. The main advantage of a linear function is that it only has two parameters, which ensures that the estimation problem stays relatively simple. The human operator, however, operates as a nonlinear system. Higher-order polynomials provide an opportunity to model nonlinear behavior, but for each additional order, the constants used to shape the polynomial rapidly decrease to very small values. When the visual-perception gain is modeled as a fourth-order polynomial, for example, the polynomial contains three constants with values between 10^{-4} and 10^{-8}. The size of and the differences between these small constants give rise to mathematical problems during parameter estimation. Therefore, in this study the Boltzmann sigmoid is selected for parametrization of the linear time-varying parameters. In addition, the Boltzmann sigmoid provides a gradual transition between two types

of dynamics. This is exactly the type of time-varying dynamics that is required for this study. It is, however, likely that there will be more appropriate or alternative functions to model this kind of time-varying behavior.

II.C. Simulated Control Task Dynamics

A single simulation run is set to last 81.92 s. This is based on 2^{13} samples and a sampling rate of 100 Hz as to retain an acceptable frequency resolution. The value of 81.92 s equals one measurement time T_m, a value related to the target forcing function which will be explained in Sec. II.D. Parameter estimation will only be performed on the second simulation run in order to avoid transient behavior. For the control task in Fig. 1 the human operator gain $K_v(t)$, the human operator lead time constant $T_l(t)$, the shaping filter gain $K_n(t)$, the vehicle dynamics gain $K_c(t)$, and the vehicle dynamics time constant $T_c(t)$ are modeled as Boltzmann sigmoids. Note that the parameters are based on the parametrization in Ref. 5. The sigmoids are plotted in Fig. 3.

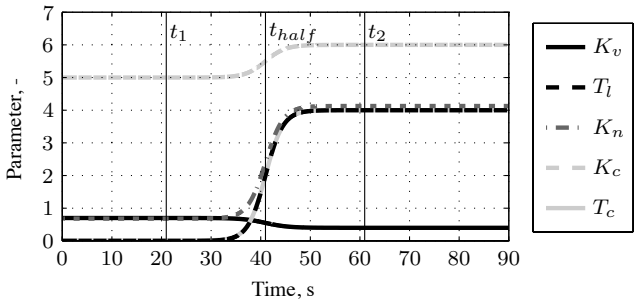


Figure 3. Time-varying parameters modeled as Boltzmann sigmoids in accordance with Equation Eq. (7).

Table 1. The parameter values for the control task. The parameters in the left column are the parameters that will be estimated.

Pilot Dynamics Parameters			Other Parameters		
Parameter		Value	Parameter		Value
K_{v_0}	sigmoid top gain constant	0.70 -	K_{c_0}	sigmoid top gain constant	6.00 -
K_{v_1}	sigmoid bottom gain constant	0.40 -	K_{c_1}	sigmoid bottom gain constant	5.00 -
K_{v_2}	sigmoid t_{half} gain constant	40.96 s	K_{c_2}	sigmoid t_{half} gain constant	40.96 s
K_{v_3}	sigmoid slope gain constant	-2.30 -	K_{c_3}	sigmoid slope gain constant	2.02 -
T_{l_0}	sigmoid top time constant	4.00 -	T_{c_0}	sigmoid top time constant	4.00 -
T_{l_1}	sigmoid bottom time constant	0.00 -	T_{c_1}	sigmoid bottom time constant	0.01 -
T_{l_2}	sigmoid t_{half} time constant	40.96 s	T_{c_2}	sigmoid t_{half} time constant	40.96 s
T_{l_3}	sigmoid slope time constant	1.77 -	T_{c_3}	sigmoid slope time constant	1.77 -
τ_v	time delay constant	0.20 s	K_{n_0}	sigmoid top gain constant	4.13 -
ω_{nm}	neuromuscular frequency	10.00 rad/s^2	K_{n_1}	sigmoid bottom gain constant	0.68 -
ζ_{nm}	neuromuscular damping	0.20 -	K_{n_2}	sigmoid t_{half} gain constant	40.96 s
			K_{n_3}	sigmoid slope gain constant	1.79 -
			T_n	shaping filter time constant	0.20 s

The parameters θ are held constant at the single-integrator dynamics values until a t_1 of 20.96 s, which is the instance when the sigmoids start. The sigmoids are halfway at a t_{half} of 40.96 s–exactly in the middle of the simulation run– and end 20 s later at a t_2 of 60.96 s. After this the double-integrator dynamics values are reached and the parameters are constant again. The corresponding parameter values of the sigmoids are given in Table 1 together with the time-invariant values of the time delay constant τ_v, the neuromuscular frequency ω_{nm}, the neuromuscular damping ratio ζ_{nm}, and the shaping filter time constant T_n. The parameters that will be estimated are stated in the left column of this table. Note that the values of the shaping filter gain $K_n(t)$ are chosen such that the remnant variance percentage of both the single-integrator and double-integrator dynamics parts has a value of 10%. The values of the human operator lead time constant $T_l(t)$ and the vehicle time constant $T_c(t)$ are almost equal, so that near-perfect human operator compensation is assumed for the double-integrator vehicle dynamics at higher frequencies.[5] For perfect compensation

T_{c_1} should also have a value of zero, but the state-space equations used to simulate the vehicle dynamics contain a division by T_{c_1}. It can thus not have a value of zero. This is, however, not an issue, as the resulting controlled element dynamics still approximate a pure single-integrator system over the entire frequency range. Finally, it should be noted that when the sigmoids are modeled, there will always exist a small gap at t_1 and t_2 where the sigmoid connects to the constant dynamics. To make sure that this gap has no significant contribution, the difference between the constant dynamics and the sigmoid at these instances has a value that is in the order of 10^{-5} or lower.

II.D. Target Forcing Function

A target forcing function f_t is used to excite the system in Fig. 1. This target forcing function is taken from Ref. 3. The forcing function, a multisine, is based on sine waves with different frequencies as in Eq. (8). The sine wave frequencies are defined as integer multiples of the experimental measurement time base frequency, $\omega_m = 2\pi/T_m$ with $T_m = 81.92$ s (based on 2^{13} samples and a simulation sampling frequency of 100 Hz). Table 2 shows the values for integer factors n_t, target sinusoid frequency ω_t, target sinusoid amplitude A_t, and target sinusoid phase shift ϕ_t. This multisine is plotted in Fig. 4a.

$$f_t(t) = \sum_{k=1}^{10} A_t(k) sin(\omega_t(k)t + \phi_t(k)) \tag{8}$$

Table 2. Properties of multisine target forcing function f_t.

k, -	n_t, -	ω_t, rad/s	A_t, deg	ϕ_t, rad
1	6	0.460	0.698	1.288
2	13	0.997	0.488	6.089
3	27	2.071	0.220	5.507
4	41	3.145	0.119	1.734
5	53	4.065	0.080	2.019
6	73	5.599	0.049	0.441
7	103	7.900	0.031	5.175
8	139	10.661	0.023	3.415
9	194	14.880	0.018	1.066
10	229	17.564	0.016	3.479

Table 3. System I & II parameter values.

Parameter	System I	System II
K_v	0.70 -	0.40 -
T_l	0.00 s	4.00 s
K_c	5.00 -	6.00 -
T_c	0.01 s	4.00 s
K_n	0.68 -	4.13 -
T_n	0.20 s	0.20 s
τ_v	0.20 s	0.20 s
ω_{nm}	10.00 rad/s^2	10.00 rad/s^2
ζ_{nm}	0.20 -	0.20 -

II.E. Simulated Time Traces

Before parameter estimation is performed, the simulated data is validated by an analysis of the system's time traces. In Fig. 4 the time traces of the system are plotted. From Fig. 4a it becomes clear that the system output y accurately follows target forcing function f_t. In the period of time where single-integrator dynamics are controlled, the vehicle closely follows the f_t while rapidly adapting to shoot-overs. During the transition from single-integrator to double-integrator dynamics it is harder to follow f_t especially after t_{half} at 40.96 s. For the double-integrator dynamics the vehicle follows f_t quite accurately again, but the human operator adapts slower to the changes in f_t as is expected for this more difficult, controlled element. The same behavior can be deduced from the plot of tracking error e in Fig. 4b. In Fig. 4c control signal u is plotted together with the remnant n. The human operator requires less effort for the single-integrator dynamics in comparison with the double-integrator dynamics. This change in the control behavior from single-integrator to double-integrator dynamics is clearly visible in this figure between t_1 and t_2. The figure also shows that the human operator remnant n is smaller for the single-integrator dynamics compared with the remnant for the double-integrator dynamics. This is caused by the shaping filter gain K_n, which keeps the remnant signal variance percentage n_p at 10%.

To better understand the dynamics of the system, the system is split-up in two parts in accordance with Table 3: System I (single-integrator dynamics) and System II (double-integrator dynamics) corresponding to the data before t_1 and after t_2, respectively. The data in Table 3 can also be deducted from Table 1, but is explicitly shown again to show the difference between System I and II, and moreover, to indicate that only constant values are used instead of sigmoids. In Fig. 5 the Power Spectral Density (PSD) of f_t, y, e, and u is plotted for System I and System II. It can be observed that system output y and tracking error e have more power at higher frequencies for the single-integrator

(a) Target forcing function f_t plotted with the system output signal y.

(b) Tracking error signal e.

(c) Control signal u plotted with the remnant signal n.

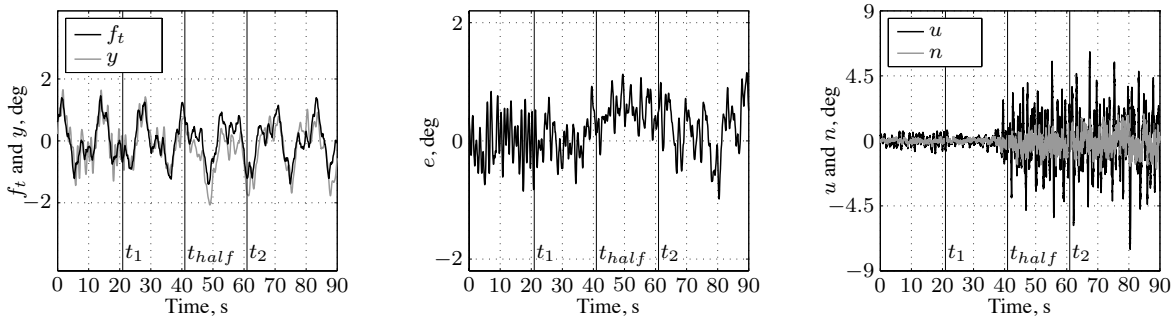


Figure 4. Time traces of the simulated signals. The Boltzmann sigmoid begins at t_1 (20.96 s) and ends at t_2 (60.96 s).

(a) Sys. I PSD of f_t and y.

(b) Sys. I PSD of e.

(c) Sys. I PSD of u.

(d) Sys. II PSD of f_t and y.

(e) Sys. II PSD of e.

(f) Sys. II PSD of u.

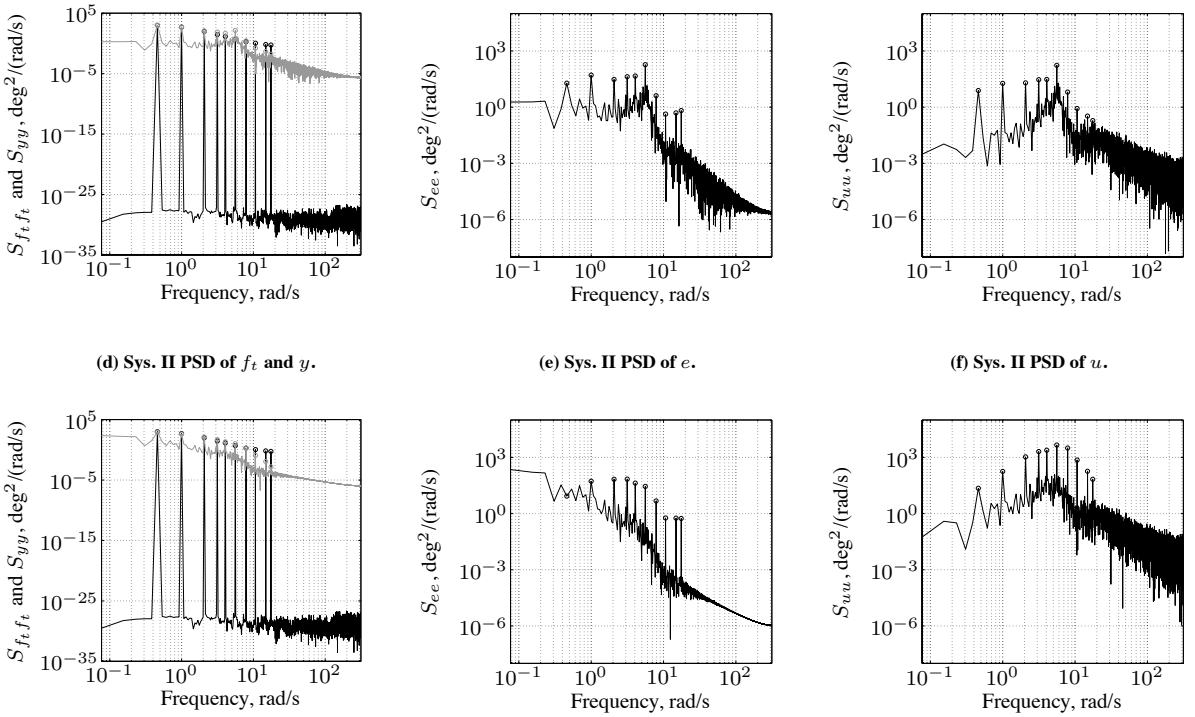


Figure 5. Power Spectral Densities of target forcing function f_t (black), output signal y (gray), error signal e, and control signal u. The frequencies ω_t, which belong to target forcing function f_t, are highlighted by the black and gray circles at the peaks of the spectra.

dynamics, which is expected after the analysis of the time traces. The power for f_t is equal for System I and II over the entire frequency range, as there is no difference for f_t between the two systems. The power for u is comparable for System I and II over the entire frequency range. Finally, in Fig. 6 the analytical frequency responses are plotted for $H_p(j\omega)$, $H_c(j\omega)$, and $H_{ol}(j\omega)$. The difference between the single-integrator dynamics and double-integrator dynamics is clearly visible from these figures. The open-loop gain crossover frequency ω_c for System I has a value of 4.14 rad/s, for System II ω_c has a value of 2.55 rad/s. The differences in these crossover frequencies explain the peaks that can be observed in the PSD plots of System I in Fig. 5.

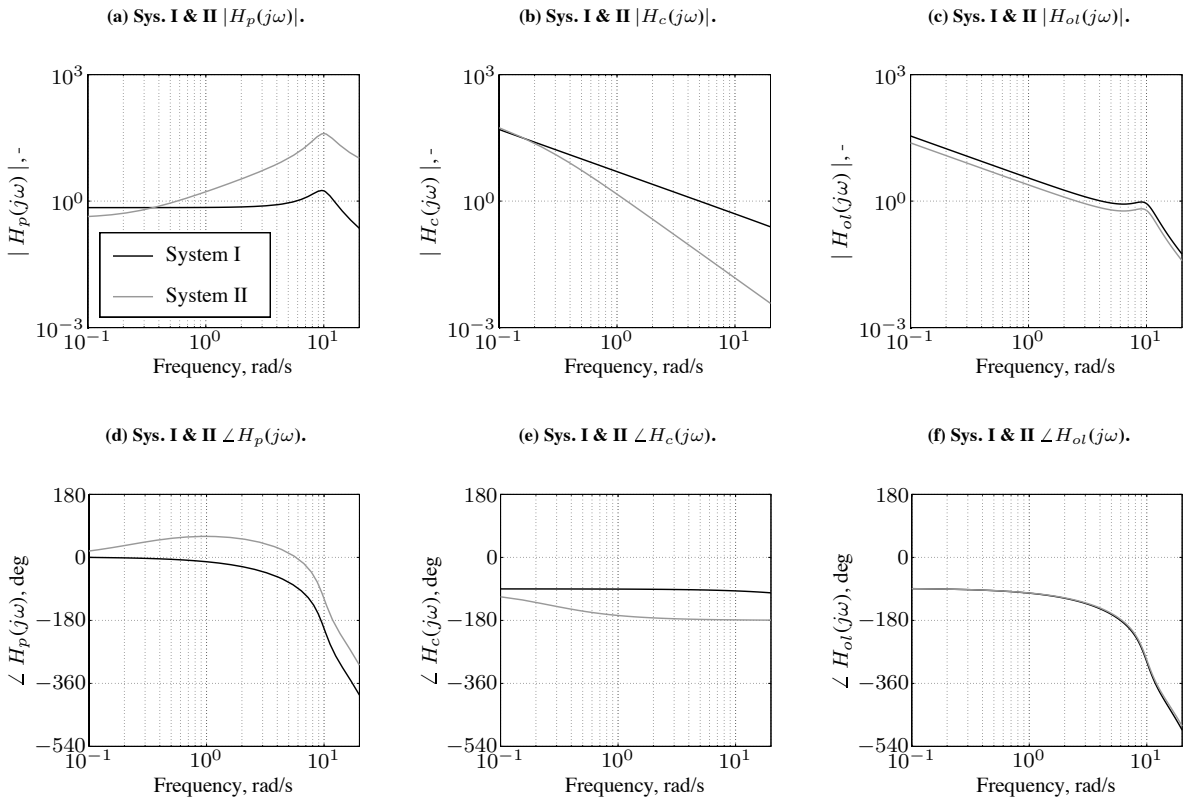


Figure 6. Frequency responses of $H_p(j\omega)$, $H_c(j\omega)$, and $H_{ol}(j\omega)$ for System I and System II as defined by Table 3.

III. Time-Varying Time-Domain Identification Procedure

Maximum Likelihood Estimation (MLE) is a statistical parameter estimation method introduced by Ronald Aylmer Fisher in 1912.[9] In the past, MLE methods for the estimation of human operator model parameters have been studied[3,5] and applied.[10] In this section, the MLE time-domain identification procedure augmented with a Gauss-Newton (GN) algorithm–a gradient-based optimization method–will be explained. In addition to the MLE method proposed in this study, a sliding time window MLE method[5] will be introduced to which the proposed method is compared.

III.A. Maximum Likelihood Estimation Theory

MLE maximizes a likelihood function L to find a vector of estimates $\hat{\Theta}$ of parameter vector Θ. All unknown model parameters that are intended to be estimated, should be part of this parameter vector. Next, the estimate $\hat{\Theta}$ of parameter vector Θ can be found by maximizing the likelihood function $L(\Theta)$ in Eq. (9), which is defined as the joint conditional Probability Density Function (PDF) of the prediction error $\epsilon(k)$ for m measurements of a control signal $u(k)$.

$$L(\Theta) = f(\epsilon(1), \epsilon(2), \ldots, \epsilon(k), \ldots, \epsilon(m) \mid \Theta) \tag{9}$$

In Eq. (9) the prediction error $\epsilon(k)$ (i.e., the residual) is the difference between the measured control signal $u(k)$ and the estimated control signal $\hat{u}(k)$ at discrete time instants k. Assuming the remnant n is an additive zero-mean Gaussian white noise signal, the conditional PDF for one measurement of prediction error $\epsilon(k)$ is given in Eq. (10).

$$f(\epsilon(k) \mid \Theta) = \frac{1}{\sqrt{2\pi\sigma_n^2}} e^{-\frac{\epsilon^2(k)}{2\sigma_n^2}} \tag{10}$$

In this equation σ_n is the standard deviation of the remnant signal n. To maximize the conditional PDF, the negative natural logarithm is minimized, which provides a more straightforward optimization problem. When a global mini-

mum of the negative log-likelihood is found, the resulting parameter vector $\hat{\Theta}_{ML}$ is the maximum likelihood estimate, which is stated in Eq. (11).

$$\hat{\Theta}_{ML} = \arg\min_{\Theta} \left[-\ln L(\Theta)\right] = \arg\min_{\Theta} \left[\frac{m}{2}\ln\sigma_n^2 + \frac{1}{2\sigma_n^2}\sum_{k=1}^{m}\epsilon^2(k)\right] \tag{11}$$

Eq. (11) summarizes the maximum likelihood parameter estimation problem. Depending on the application (e.g. model or data) Eq. (11) can represent a linear or a highly nonlinear optimization problem.

The GN algorithm is used to optimize the parameter estimation by finding $\hat{\Theta}_{ML}$. The GN algorithm is one of the most used algorithms in combination with MLE and further optimizes the negative log-likelihood of the likelihood function L.[11] It is a gradient-based method and Eq. (12) is its iterative parameter update equation. Note that the combination of MLE with the nonlinear GN method is an output error method, which requires the system to be written in state-space form. For this purpose the transfer functions in Sec. II.A are rewritten in their controller canonical form.

$$\hat{\Theta}(k+1) = \hat{\Theta}(k) - \alpha(k) M_{\Theta\Theta}^{-1}(\hat{\Theta}(k)) \frac{\partial L(\hat{\Theta}(k))}{\partial \Theta} \tag{12}$$

The line-search parameter α in this equation has a value $0 \leq \alpha \leq 1$, and is updated before each iteration to ensure optimal minimization of likelihood function L along the vector defined by $M_{\Theta\Theta}^{-1}(\hat{\Theta}(k))\frac{\partial L(\hat{\Theta}(k))}{\partial \Theta}$. The nonlinear estimation problem under consideration in this paper yields a high number of local minima.[12] The GN algorithm is very sensitive to these local minima in the sense that when it converges to a certain local minimum, finding the global minimum is impossible. The incorporation of a line-search in the GN method is advocated in Ref. 13 to increase GN algorithm stability and the chance of convergence to the global minimum. The line-search used for this purpose always starts at $\alpha_0 = 0$ and $\alpha_1 = 1$. Now, with $\alpha_i = \{1 \ldots k\}$, when

$$L(\alpha_0) > L(\alpha_i) \quad \longrightarrow \quad L = L(\alpha_i)$$

and, when

$$L(\alpha_0) < L(\alpha_i) \quad \& \quad L(\alpha_0) > L\left(\frac{\alpha_0 + \alpha_i}{2}\right) \quad \longrightarrow \quad L = L\left(\frac{\alpha_0 + \alpha_i}{2}\right).$$

However, when

$$L(\alpha_0) < L(\alpha_i) \quad \& \quad L(\alpha_0) < L\left(\frac{\alpha_0 + \alpha_i}{2}\right) \quad \longrightarrow \quad \alpha_{i+1} = \frac{\alpha_0 + \alpha_i}{2}$$

the line-search algorithm starts from the beginning but now with $\alpha_i = \alpha_{i+1}$. The line-search vector stops and selects the lowest likelihood value when the ratio $\left(L(\alpha_0) - L(\alpha_i)\right)/L(\alpha_i)$ drops below a value of 10^{-3}. The stopping criterion of the GN algorithm is defined by $\left(L(\hat{\Theta}(k)) - L(\hat{\Theta}(k+1))\right)/L(\hat{\Theta}(k+1))$ and is fixed at a value of 10^{-6}. In Eq. (12) the Fisher information matrix $M_{\Theta\Theta}$ is defined by Eq. (13) and should be symmetrical and positive definite to be inverted for Eq. (12).

$$M_{\Theta\Theta} = \frac{1}{\sigma_n^2}\sum_{k=1}^{m}\left(\frac{\partial \epsilon(k)}{\partial \Theta}\right)^2 \tag{13}$$

To calculate the $\partial\epsilon(k)/\partial\Theta$-term of the Fisher information matrix, it is necessary to determine the so-called sensitivity matrices. The sensitivity matrices show how the coefficients in the state-space matrices change when the value of a certain parameter also changes. The sensitivity of the entries of the state-space matrices $A(t)$, $B(t)$, $C(t)$ and $D(t)$ to changes in each parameter in Θ are calculated and used to determine both $M_{\Theta\Theta}$ and the likelihood gradient $\partial L(\hat{\Theta}(k))/\partial\Theta$. The necessity of writing the system in state-space form is hampered by the presence of time delay τ_v, which occurs as an exponential term in Eq. (2). The exponential time delay term is therefore included as a fifth-order Padé approximation, as is done in Ref. 3.

III.B. Comparing Two Time-Varying Maximum Likelihood Estimation Methods

In this paper a time-domain identification procedure based on maximum likelihood estimation is proposed for the estimation of time-varying pilot vehicle system parameters. To see how well this time-varying MLE method compares to other methods, it will be compared to the sliding time window MLE method used in Ref. 5. In this section both

methods will be briefly introduced. It is possible to compare the methods relative to each other with the use of the Variance Accounted For (VAF). The VAF is used to compare the estimated control signal $\hat{u}$ of one of the methods to the corresponding control signal u. If the estimated control signal equals the simulated control signal, the VAF will be 100%. Note that the line-search algorithm as proposed in Sec. III.A is used for both methods.

III.B.1. Method 1: Time-Varying Model Identification

For the time-varying model identification procedure the entire data set of a single simulation run of 81.92 s will be used. As discussed in Sec. II.C, for the control task in this paper, data from the second simulation run is used to avoid transient behavior in the first simulation run. The parameters related to the linear human operator response, stated in the left column of Table 1, will be estimated. Consequently, the parameter vector contains eleven parameters as stated in Eq. (14).

$$\Theta = \begin{bmatrix} K^*_{v_0} & K_{v_1} & K_{v_2} & K_{v_3} & T^*_{l_0} & T_{l_1} & T_{l_2} & T_{l_3} & \tau_v & \omega_{nm} & \zeta_{nm} \end{bmatrix}^\top \tag{14}$$

Before the identification procedure is initiated, initial values have to be given to the parameters, which are used for the first parameter estimation iteration. Although a randomization of these initial values is usually applied, the choice is made to offset each parameter with an equal percentage from the true values that were used to create the simulated data. This is done in order to make it easier to compare the results. The true values are labeled as 100%, so when *all* parameters are subjected to a positive offset of 10% this equals a parameter variance θ_p of 110%. Note that T_{l_1} has a value of zero. To be able to give T_{l_1} an offset, it is set to 0.01 for the instances that it is given an offset by θ_p. Next to giving all parameters an equal offset of θ_p, another experiment will be done in which all parameters, except for one, are held at the true values. This will be done to check if varying the initial value of each single parameter has a significant effect on the parameter estimation.

III.B.2. Method 2: Sliding Time Window Identification

In Ref. 5 the sliding time window MLE method is described as follows:

To estimate time-varying human operator model parameters in the current study, the MLE optimization is performed at every time step t_i using a sliding time window of length Δt. Choosing a Δt that is too small will decrease the accuracy of estimated parameters related to low-frequency dynamics. A Δt that is too large will reduce the method's ability to detect small variations in human operator model parameters. In the current study, the length of the time window is chosen to be 20 s.

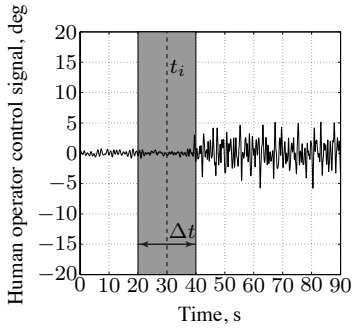


Figure 7. Sliding time window MLE method.

The sliding time window MLE method in Ref. 5 is performed every 2 s, and because of the width of the window it is initiated at $t = 12.0$ s and stops at $t = 80.0$ s. To compare the sliding time window MLE method with the proposed method in this paper, the sliding time window method will be performed according to the definition in Ref. 5, but on the simulated data generated by control task defined in Sec. II. In Fig. 7 a representation of the sliding time window MLE method is given as defined by Ref. 5. Because of the sliding time window, this method only observes a certain part of the data and the parameters are assumed to have constant values for the data in Δt. The parameter vector in Eq. (15), therefore, only consists of five parameters.

$$\Theta = [K_v \;\; T_l \;\; \tau_v \;\; \omega_{nm} \;\; \zeta_{nm}]^\top \tag{15}$$

In Ref. 5 the initial parameter set was constructed from the simulated pilot model parameters at each time step. Because only $K_v(t)$ and $T_l(t)$ are considered in this parameter estimation problem and because the proposed time-varying model MLE method only uses one initial parameter which is set at the beginning instead of at each time step, it is chosen to set the initial values of the sliding time window method to the average of the top and $bottom$ values of these sigmoids multiplied by the same parameter variance θ_p that is chosen for the proposed MLE method.

IV. Results

With the simulated data from the control task in Sec. II and the MLE methods in Sec. III, parameter estimation is performed. The results of this estimation will be discussed in this section. Simulations were performed for three different noise realizations with different parameter variance percentages θ_p and different remnant variance percentage n_p. Following these results, the outcome of further investigation on the influence of the remnant n and the initial parameter values will be discussed and the results of an evaluation on the effect of local minima will be examined.

IV.A. Parameter Estimation Results

The results of parameter estimation are plotted in Fig. 8. The plots show the results obtained by setting the initial parameter values to a parameter variance θ_p of 110%. For the estimation, simulated data is used, which is generated by the control task in Sec. II without a remnant (n_p = 0%). The four estimated sigmoid parameters for both $K_v(t)$ and $T_l(t)$ were used to plot the sigmoids in Fig. 8a and Fig. 8b. For the proposed MLE method (indicated by MLE_1) the sigmoidal shape is clearly visible. Without any human operator remnant signal, the parameter estimation with the sigmoids is perfect. The sliding time window MLE method (indicated by MLE_2) also produces estimates which come

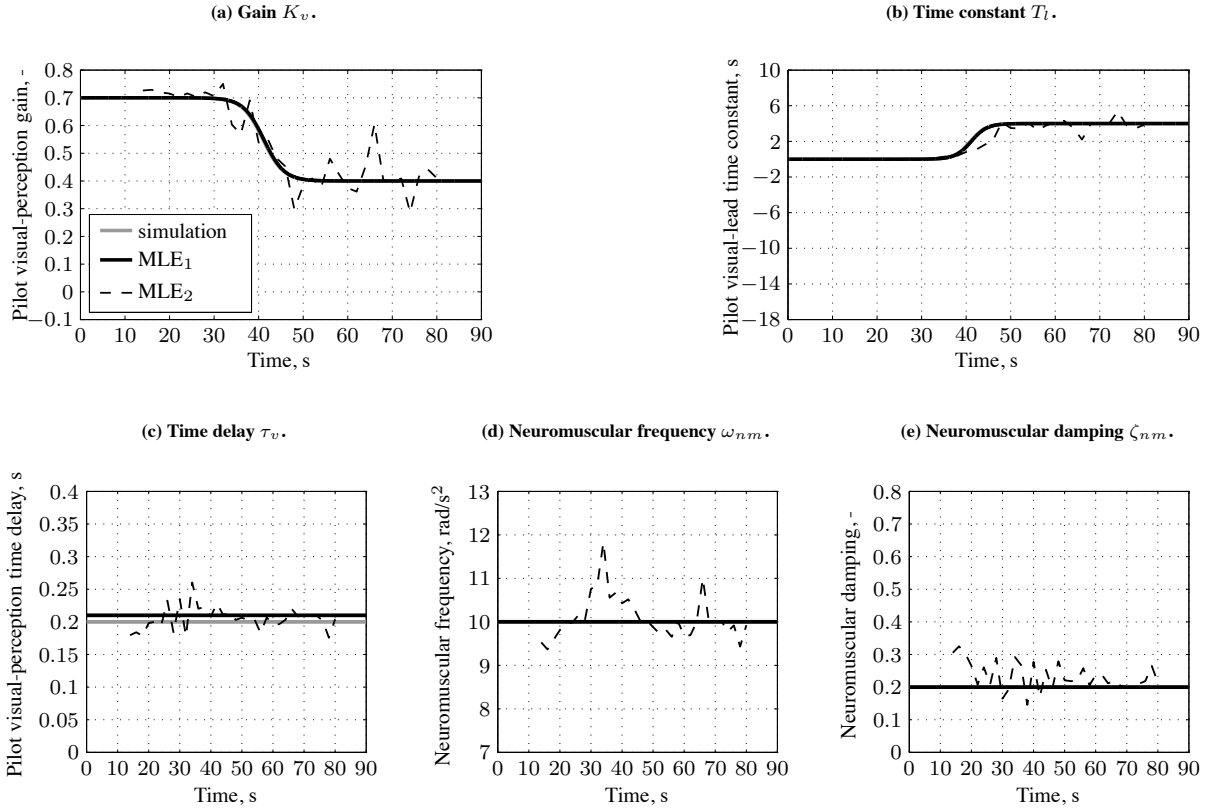


Figure 8. Parameter estimation results for the proposed MLE method (MLE_1) and the sliding time window MLE method (MLE_2). For the estimation, the initial parameter values were all subjected to a θ_p of 110%.

close to the values provided by the simulated data, but it shows a lot of peaks and fluctuation around these values. An evaluation of MLE_1 and MLE_2 for the estimation of the fixed, time-invariant parameters in Fig. 8c, Fig. 8d, and Fig. 8e shows similar results. With a VAF_1 of 100% and a VAF_2 of 82.3% the proposed method is significantly better than the sliding time window method. Note that VAF_1 is only calculated for the measurement time that the sliding time window method acts (from 12 s to 80 s). VAF_2 is calculated by averaging the total sum of the VAFs for each time window estimation instance.

When the remnant variance percentage n_p is increased, however, a clear difference with the system without remnant can be seen. To show this, $K_v(t)$ and $T_l(t)$ are plotted in Fig. 9 for estimation performed on simulated data with remnant variance percentages of 0%, 5%, and 10%. The figures show that both MLE methods return worse estimation results when the remnant contribution is increased. For the results with an n_p of 5% and 10%, the top-value of $K_v(t)$ and the $bottom$-value of $T_l(t)$ seem to be estimated correctly, but their respective $bottom$- and top-values do not follow the simulated sigmoids. The sliding time window MLE method seems to have similar problems when estimating the changing behavior (between 20.96 s and 60.96 s), but rapidly stabilizes and starts fluctuating around the simulated values after 60.96 s. The plots of the fixed values τ_v, ω_{nm}, and ζ_{nm} are not shown here, but indicate good results for MLE_1. Despite the fact that MLE_2 shows a lot of peaks and fluctuation around the fixed values, the estimation for the fixed values is also adequate. With a VAF_1 of 95.5% and a VAF_2 of 76.6% for a n_p of 5%, the proposed method still produces better results. For a n_p of 10% this is also the case, with a value of 90.8% for VAF_1 and a value of 71.0% for VAF_2. In conclusion, the addition of a remnant to the system has adverse effects on the parameter estimation. There is a clear difference in performance between the two methods and MLE_1 consistently performs better. It is notable that with an increase in human operator remnant, the ΔVAF between the two methods also increases.

The results obtained for parameter estimation up until now are based on simulation data in which the time-varying parameters are modeled as Boltzmann sigmoids. It is important to understand that for the parameter estimation in this case, the time-varying parameters are estimated as Boltzmann sigmoids. The time-varying dynamics that can be modeled with a sigmoid are expected to be representative for a transition in human control behavior between two

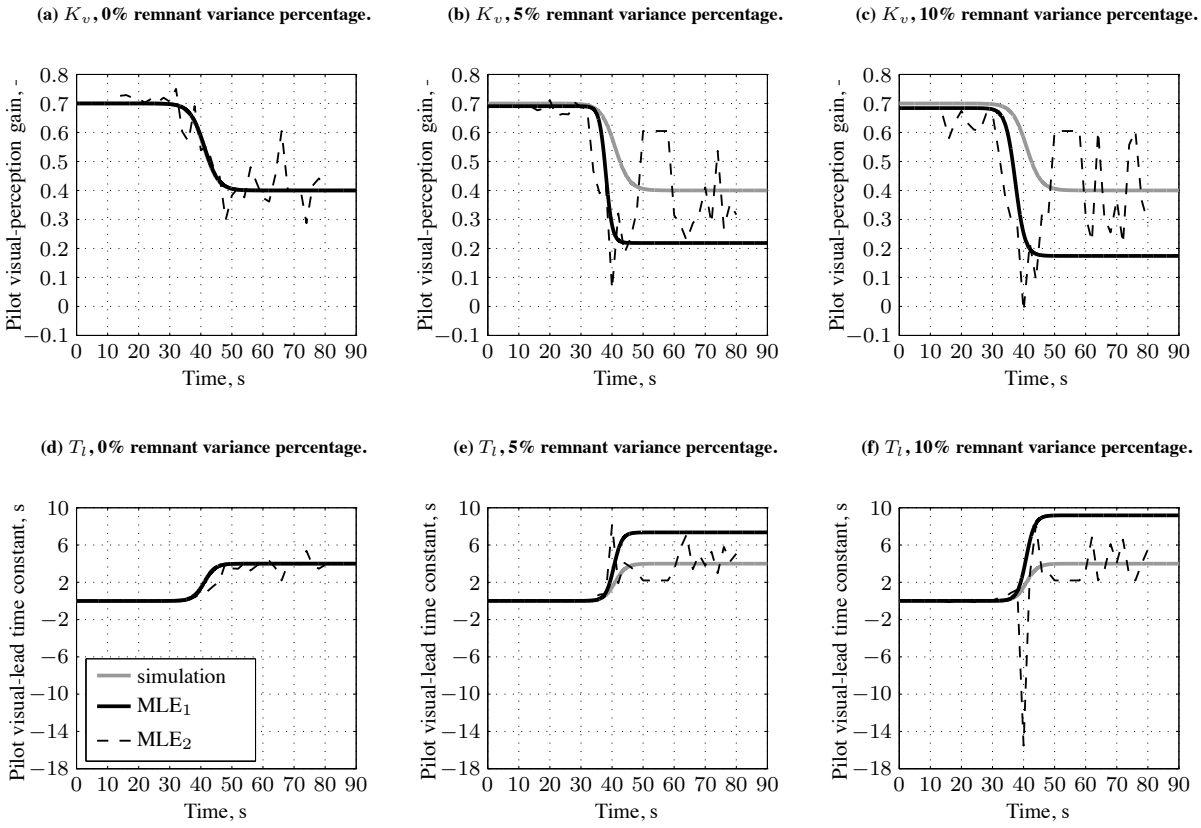


Figure 9. Parameter estimation results for sigmoids K_v and T_l under the influence of different amounts of remnant variance percentage n_p. For the estimation, the initial parameter values were all subjected to a θ_p of 110%.

dynamic systems. However, it is interesting to see in what way the proposed estimation method with sigmoids will react on simulation data in which the time-varying parameters are not modeled as a Boltzmann sigmoid. To check this, the Boltzmann sigmoid in the simulation data is replaced with a linear transition as depicted by Fig. 2a and estimated with the proposed estimation method. The parameter estimation results of both the proposed method MLE_1 and the sliding time window method MLE_2 are shown in Fig. 10 for K_v and T_l with remnant variance percentages n_p of 0%, 5% and 10%. The results are comparable to those of Fig. 9, except for the results of K_v with an n_p of 0% in which the sigmoid seems to be influenced by biases. It is also worth mentioning that the sliding time window method MLE_2 seems to perform better than MLE_1 for T_l. In general, however, the proposed method still performs better in terms of the VAF. For an n_p of 0%, VAF_1 equals 96.4% and VAF_2 equals 84.8%. For an n_p of 5%, VAF_1 has a value of 95.5% and VAF_2 has a value of 79.8%. And, consequently, for an n_p of 10%, VAF_1 amounts to 91.6% and VAF_2 to 75.4%. The big differences between the VAFs are not directly visible from Fig. 10, but are more evident from the estimation of the fixed values τ_v, ζ_{nm}, and ω_{nm} where MLE_2 performs drastically worse for this simulation data set.

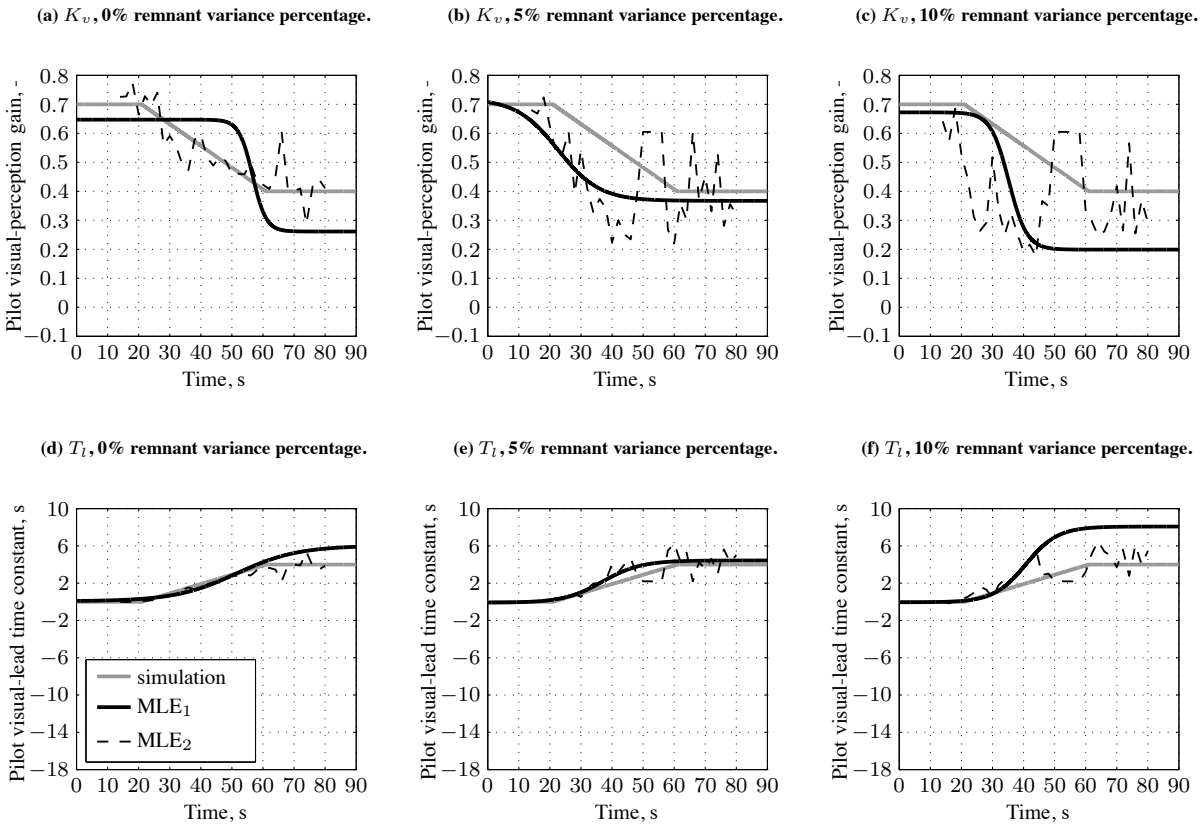


Figure 10. Parameter estimation results for ramp signals K_v and T_l under the influence of different amounts of remnant variance percentage n_p. For the estimation, the initial parameter values were all subjected to a θ_p of 110%.

IV.B. Influence of Remnant and Initial Parameter Values

In Sec. IV.A it became clear that there are two important factors that have an influence on the accuracy of the parameter estimation results: the amount of remnant signal in the simulated data (the remnant variance percentage n_p) and the offset given to the initial parameter values for parameter estimation (the parameter variance percentage θ_p). Although a larger remnant has a detrimental effect on the results, the parameter variance percentage θ_p showed to have a small effect on parameter estimation. To verify this, and to understand how the remnant variance percentage and the parameter variance percentage influence the proposed MLE method, twelve sets of simulation data for three different white noise realizations are generated. Table 4 gives an overview of the characteristics of these twelve simulation data sets. For the complete control task as defined in Sec. II, six sets are generated: three closed-loop system sets and three open-loop system sets. The difference in these sets of three is the way in which white noise w is added to the system:

Table 4. Overview of the characteristics of twelve packages of simulation data. Closed-loop is abbreviated to 'CL' and open-loop to 'OL'.

Complete System		System I*	System II*
CL filtered noise	OL filtered noise	CL filtered noise	CL filtered noise
CL varying white noise	OL varying white noise	CL varying white noise	CL varying white noise
CL constant white noise	OL constant white noise	CL constant white noise	CL constant white noise

Table 5. System I* & II* parameter values.

Parameter	System I*	System II*
K_{v_0}	0.70 -	0.70 -
K_{v_1}	0.40 -	0.40 -
K_{v_2}	122.88 s	122.88 s
K_{v_3}	-2.30 -	-2.30 -
T_l	0.00 s	4.00 s
K_c	5.00 -	6.00 -
T_c	0.01 s	4.00 s
T_n	0.20 s	0.20 s
τ_v	0.20 s	0.20 s
ω_{nm}	10.00 rad/s^2	10.00 rad/s^2
ζ_{nm}	0.20 -	0.20 -

as filtered noise through the shaping filter (i.e., the remnant), as a zero-mean Gaussian white noise signal multiplied by a time-varying sigmoid K_n to keep the remnant variance percentage of the single-integrator dynamics equal to that of the double-integrator dynamics, or as a time-invariant zero-mean Gaussian white noise signal. Two other sets of three are generated to see if MLE is influenced by the number of parameters, and to see if the single-integrator (System I*) and double-integrator (System II*) dynamics behave different from the complete system. Compared to Sec. IV.A the number of parameters is reduced as T_l is now set to a fixed value. The system parameters for System I* and System II* are given in Table 5. Note that System I* and System II* differ from System I and System II, as for the former systems, $K_v(t)$ is modeled as a sigmoid. Furthermore, the values of $K_n(t)$ are not given in the table, as the values are dependent on the kind of noise and the remnant variance percentage n_p for each set in Table 4.

The twelve simulation data sets are evaluated by estimating parameters for a θ_p between 70% and 130%, and for a n_p ranging from 0% to 20%. For each parameter a three-dimensional plot is made, where θ_p and n_p are set out against the percentual difference $\Delta\theta$ of estimated parameter $\hat{\theta}$ in comparison with the true parameter value θ_0. In Fig. 11 the parameter estimation results for the lead time constant parameter T_{l_0} are shown.

When Fig. 11a is compared with Fig. 11b and Fig. 11c, a clear difference can be seen between the filtered noise closed-loop system and the white noise systems: while the white noise systems show U-shaped plots in the direction of the n_p-axis, the filtered noise system shows a disordered plot. Filtering the noise thus has an adverse effect on parameter estimation. From the results in Sec. IV.A it was expected that a higher n_p would have a significant influence on the estimation result. This is true, as for all plots (except for Fig. 11g) a gradual increase in $\Delta\theta$ can be seen with an increase in n_p (with some exceptions at low values of θ_p). Unexpected, however, are the large differences that are introduced to the system by θ_p. In general, it seems that a θ_p between 90% and 110% gives the best results. Below and above these values, the percentual difference in parameter value $\Delta\theta$ rapidly increases towards very large values. When the open-loop system plots in Fig. 11d, Fig. 11e, and Fig. 11f are compared to the closed-loop system plots, the open-loop systems show better results, especially for a θ_p above 120%. The results for System I* in Fig. 11g, Fig. 11h, and Fig. 11i illustrate that $\Delta\theta$ becomes very large for the single-integrator dynamics. Because parameter estimation returned so-called 'not a number' (NaN) values–in this instance unrepresentably large values–in the case of constant white noise, the plot in Fig. 11i is not complete. At the location where NaNs were found, empty spots are visible in this figure. This indicates that the constant white noise band has a contribution that is too big for the single-integrator dynamics. When the double-integrator dynamics are evaluated, Fig. 11j shows that $\Delta\theta$ grows rapidly when θ_p is not 100%. This is even more clear from Fig. 11l. For System II*, $\Delta\theta$ rapidly increases with an increasing n_p. The varying white noise results for System II* are unexpected: the estimation of the single-integrator dynamics

in Fig. 11h is much harder than the estimation of the double-integrator dynamics in Fig. 11k. In addition, the plots of all parameters for closed-loop System II* with varying white noise are very similar to Fig. 11k, whereas normally some differences can be seen between the parameters. This indicates that for a varying white noise, parameters of the double-integrator dynamics are easily estimated above a θ_p of 80% even though an increasing n_p has a slight negative effect on the estimation. Although the influence of System I* on $\Delta\theta$ seems big in comparison to that of System II*, the contribution in the complete closed-loop and complete open-loop systems seems mostly to originate from System II* due to similar shapes of the plots. Note that for System I* & II* parameter T_{l_0} is a fixed, time-invariant value.

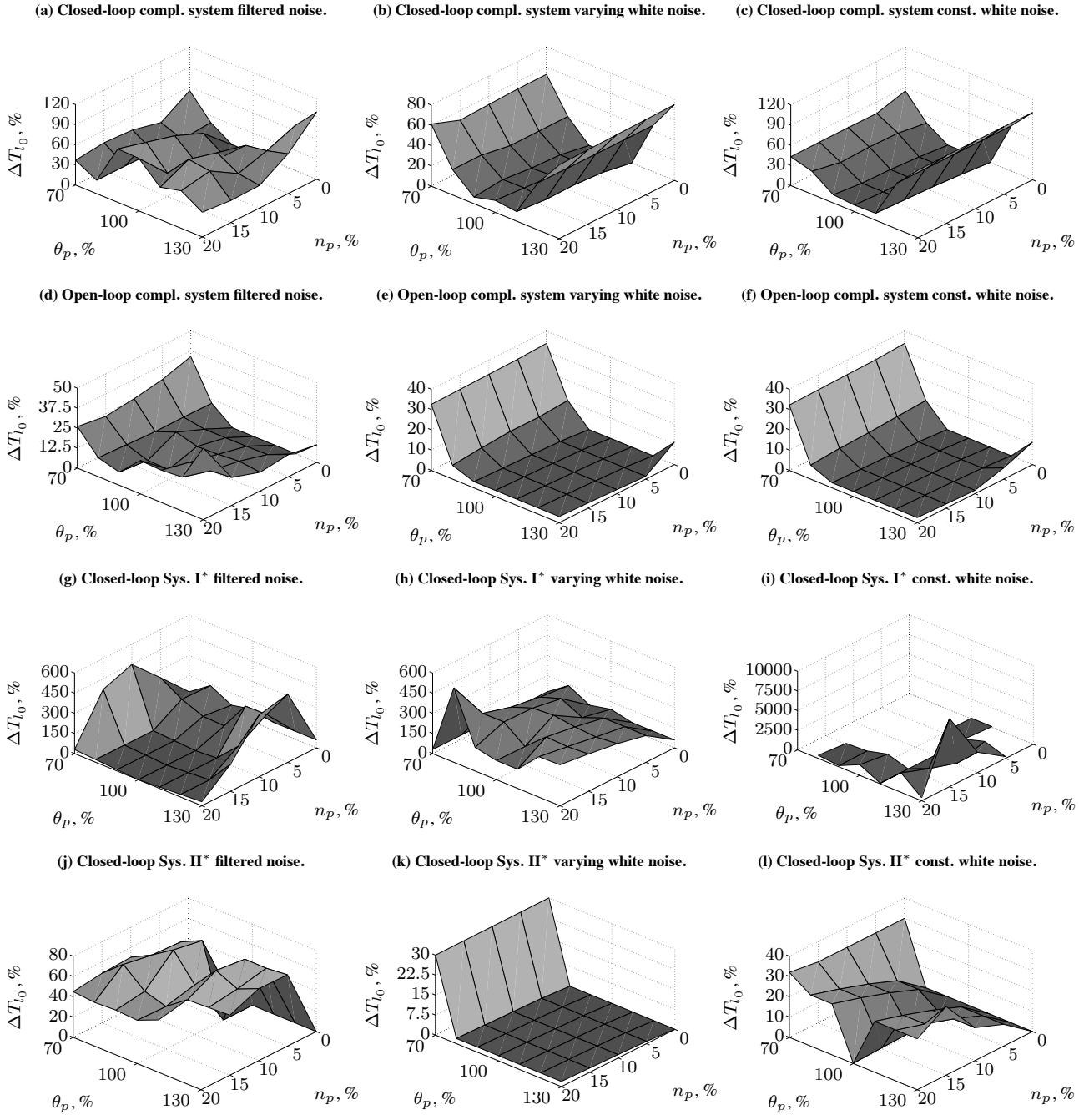


Figure 11. Percentual parameter difference $\Delta\theta$ of T_{l_0} performed on twelve packages of different simulation data according to Table 4 plotted for a range of parameter variance percentages θ_p and remnant variance percentages n_p.

However, when the plots of T_{l_0} are compared to the results for the sigmoid constants of $K_v(t)$ of System I* & II*, no real difference can be found.

In addition to the twelve simulation data sets in Table 4, another eleven data sets are created. In each of these data sets one specific parameter is varied for θ_p, whereas the other parameters are held at their true values (θ_p = 100%) as explained in Sec. III.B. From the parameter estimation results of these data sets it becomes clear that the variation of a single parameter with θ_p has an effect on all parameters. This indicates a correlation between the parameters. Except for the individual θ_p variation data sets of T_{l_1}, T_{l_2}, and ω_{nm}, all data sets show similar results in terms of $\Delta\theta$. As the results for the T_{l_1} data set can be attributed to giving T_{l_1} a value of 0.01 instead of zero (as explained in Sec. III.B), the only significant effects on $\Delta\theta$ can be assigned to the variation of θ_p for T_{l_2} and ω_{nm}. Considering that the only effects for these two parameters are observed at θ_ps of 70% and 130%, it can be concluded that within a range of 80% to 120% for θ_p these two parameters show estimation results similar to the other parameters. This means that within this range no single parameter has a significantly larger contribution to the parameter estimation. Simultaneously, it can be concluded that an increase in human operator variance percentage n_p results in a similar increase in $\Delta\theta$ for each of the eleven data sets. This once again leads to the conclusion that a higher amount of human operator remnant has a negative effect on the parameter estimation results in general.

IV.C. Local Minima

As mentioned in Sec. III.A, the nonlinear estimation problem under consideration in this paper yields a high number of local minima. When parameter estimation is performed, the GN algorithm repeatedly fails to find the global minimum, because it ends up at a local minimum. To show that local minima form a problem, the negative log-likelihood and a comparison of remnant n with prediction error $\epsilon(k)$ are plotted in Fig. 12. Fig. 12a, Fig. 12c, and Fig. 12e show the results of parameter estimation, done with a θ_p of 110%, for the closed-loop control task as defined in Sec. II with a n_p of 10%. The negative log-likelihood in Fig. 12a steadily decreases to a value of 5905. The first and last iteration of the GN algorithm are plotted in Fig. 12c and Fig. 12e. The difference in prediction error and remnant at the first iteration is neatly reduced at the seventh iteration, indicating that there is a high probability that the global minimum is found.

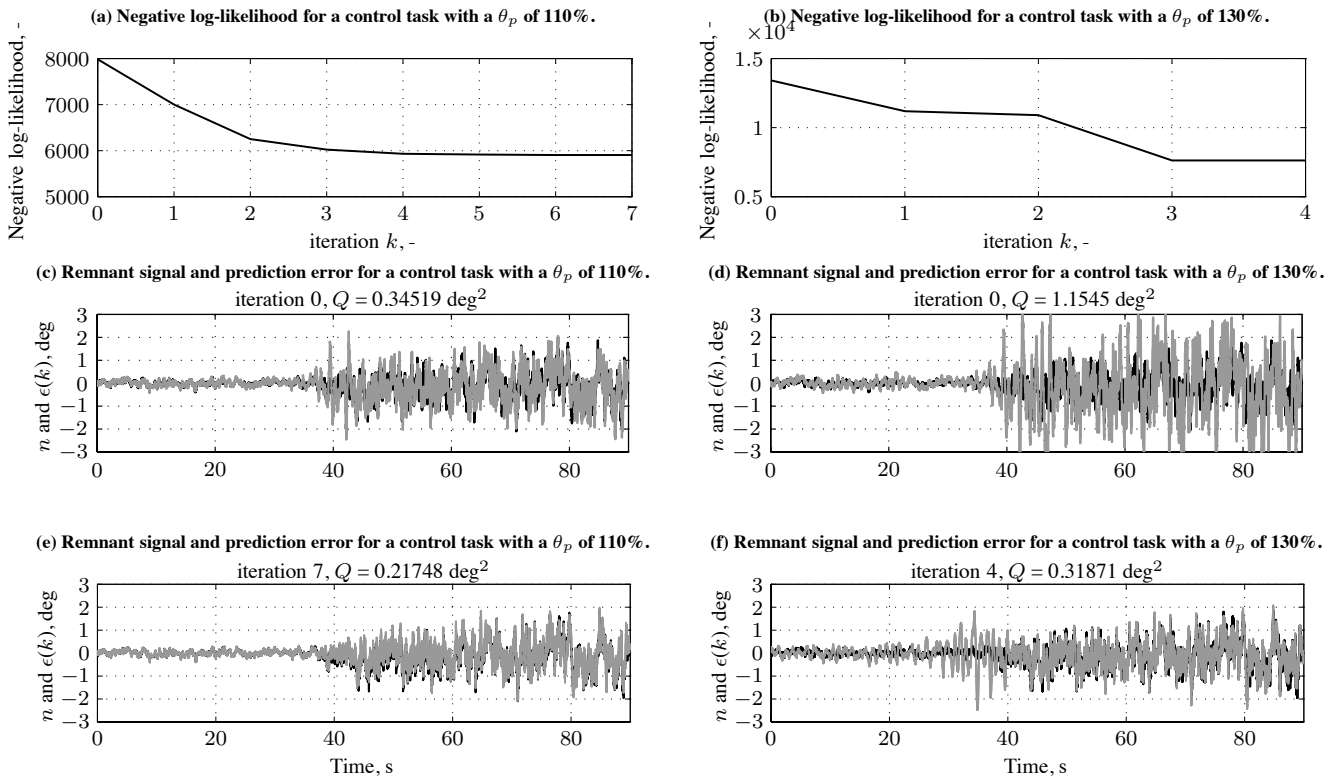


Figure 12. Negative log-likelihood and prediction error results of two parameter estimation runs with a n_p of 10% on simulated data of a closed-loop control task with θ_ps of respectively 110% and 130%.

The small differences that are still visible in Fig. 12e between the prediction error and the remnant are caused by minor differences in the simulation of the data in Simulink® and the parameter estimation in MATLAB®. The differences are related to the fourth-order Runge-Kutta method used for integration. In Fig. 12b, Fig. 12d, and Fig. 12f the results of parameter estimation, done with a n_p of 10%, on data of a closed-loop control task with a θ_p of 130% are shown. The negative log-likelihood for this parameter estimation, plotted in Fig. 12b, erratically reduces to a value of 7625. As this is the same closed-loop control task data set, the negative log-likelihood is expected to drop to a value of 5905 as well. The final value of 7625 indicates that the GN algorithm converged to a local minimum. This is also visible from Fig. 12d and Fig. 12f, as at iteration 4 the prediction error still depicts a distinct difference from the remnant signal.

To understand how the local minima problem relates to the parameter variance percentage θ_p and the remnant variance percentage n_p, the negative log-likelihood values of the final GN iterations for the closed-loop control task are averaged for the three noise realizations that were used before in Sec. IV.B. The results are plotted in Fig. 13. Each of the lines represents a system with a certain amount of remnant n_p. Optimal parameter estimation for each system would result in convergence to the global minimum. From these figures it becomes clear that the global minimum is usually found when θ_p has a value between 90% and 110%. Below and above these values, the negative log-likelihood is higher than the value of the global minimum, which leads to the conclusion that the GN algorithm reduces to a local minimum. It thus seems that a clear relation exists between the initial parameter values and finding the global minimum. When the influence of the remnant is evaluated, it becomes clear that an increasing remnant makes the U-shaped plots flatter, indicating that the difference between the global minimum and local minima becomes smaller. This is especially true for higher values of θ_p. Note that there is a big difference between the system without remnant ($n_p = 0\%$) and the system with a remnant variance percentage n_p of 5%. This suggests that for systems with remnant variance percentages below 5% the difference between the global minimum and the local minima rapidly reduces.

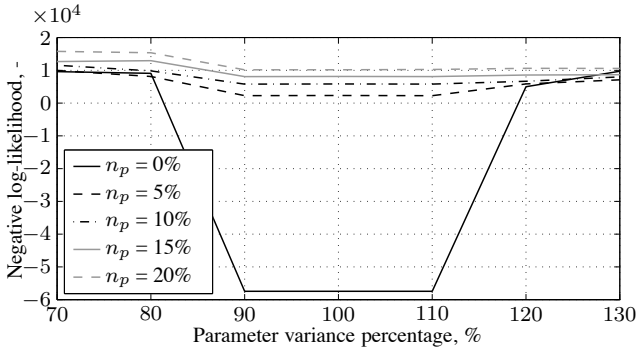


Figure 13. The negative log-likelihood for different parameter variance percentages θ_p and different remnant variance percentages n_p.

In Fig. 14 similar results are shown for the estimation in which only one initial parameter is varied with θ_p, whereas all other initial parameter values are held at their true values (θ_p = 100%). All parameters, except for T_{l_2} and ω_{nm}, show near-constant lines for the negative log-likelihood as shown for K_{v_0} in Fig. 14a. This indicates that if only one of these parameters is varied, the probability that the global minimum is found is likely, even at high and low parameter variance percentages. A cause for this might be that ten out of eleven parameters start with the true value as their initial value. In Fig. 14b and Fig. 14c, the results for T_{l_2} and ω_{nm} are shown. For T_{l_2} local minima are found below a θ_p of 90% and above a θ_p of 120%. For ω_{nm} local minima are found below a value of 80% for θ_p. A higher n_p especially has positive effects for ω_{nm} as its lines become flatter with higher remnant variance percentages. These results for T_{l_2} and ω_{nm} are consistent with the results in Sec. IV.B, where T_{l_2} and ω_{nm} also showed to have detrimental effects on the parameter estimation results. Note, that in Sec. IV.B the boundaries, in which parameter estimation still seems to return proper results, are identified to range from for 80% to 120% for θ_p. Here, these limits seem to be even smaller with values of 90% and 120%. From these results it can be concluded that for the estimation problem considered in this study, the parameters T_{l_2} and ω_{nm} contribute in a negative way to the local minima problem, especially for parameter variance percentages that are further removed from the true values at 100%.

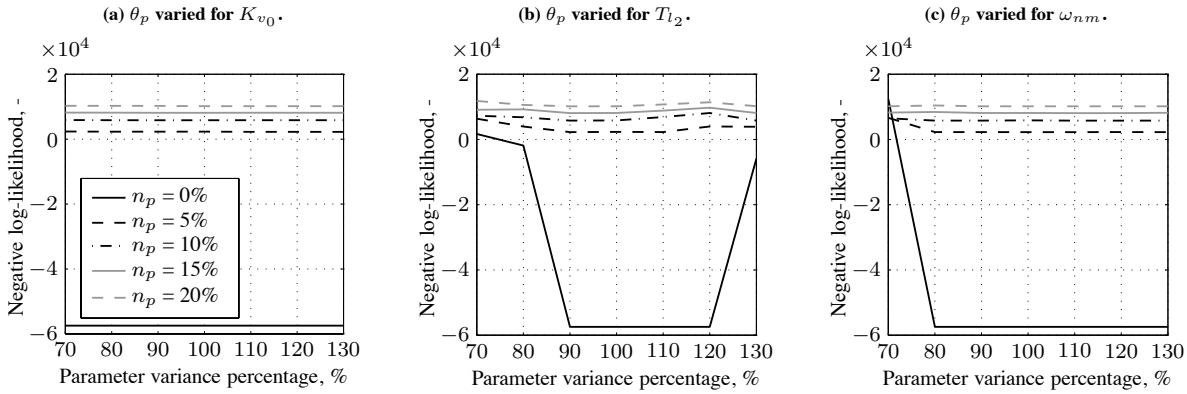


Figure 14. The negative log-likelihood for different parameter variance percentages θ_p and different remnant variance percentages n_p. For the estimations in the figures only the respective parameter is varied for θ_p, all other parameters are held at a θ_p of 100%.

V. Discussion

Parameter estimation with a proposed time-varying time-domain MLE identification procedure was done on simulated data of a closed-loop manual control task. The results of the proposed MLE method were compared to the results of the sliding time window MLE method proposed by Ref. 5. A comparison of the respective VAF values of the methods leads to the conclusion that the proposed method performs better than the sliding time window method. The tested hypothesis in Sec. I can thus be confirmed. Because the MLE method proposed in this paper uses Boltzmann sigmoids to model the time-varying dynamics, the estimation results in a much smoother signal for the estimated time-varying parameters in comparison to the sliding time window MLE method, which often returns rapidly changing results. When regarding computational effort, the proposed method is much faster than the sliding time window method.

The smooth shape of the Boltzmann sigmoids also has its drawbacks in the sense that the Boltzmann sigmoid must be rather steep for the sigmoid to start at the same values of the single-integrator at t_1 and to end at the double-integrator dynamics values at t_2. Although the entire Boltzmann sigmoid is modeled over 40 s, a rapid change occurs in a very short period of time. This is not a big problem, as it is expected that a human operator adapts his or her control behavior rapidly from one set of dynamics to another set. However, if a more gradual transition between the dynamics needs to be modeled, the Boltzmann sigmoid might not be suitable. Furthermore, the choice of a Boltzmann sigmoid imposes restrictions on the shape of the transition between different sets of dynamics, if parameter estimation on polynomials would pose less problems, this might be a better choice due to the innate freedom of a polynomial's shape. Finally, the Boltzmann sigmoid introduces four parameters to the parameter estimation problem for each time-varying parameter that needs to be estimated. It thus rapidly increases the complexity of the estimation problem.

The results of the estimation problem are heavily influenced by the human operator remnant, by the choice of the initial parameter values, and by local minima. Due to the nonlinear behavior of the human operator, the addition of a remnant is unavoidable to model a realistic control task. It is therefore difficult to reduce the negative contribution the remnant has on the proposed MLE method. It should be noted that the research in this study was done by giving all parameters an equal offset with θ_p. This was done to be able to easily compare the results to each other. It might be useful to study what the effects are of taking randomized values of the initial parameter values. The problem related to the initial parameter values can be reduced by, for example, using a genetic algorithm as proposed in Ref. 3. A major drawback of such a genetic algorithm is the amount of computational time needed and the fact that there is no guarantee that the result reduces to initial parameter values that enhance the parameter estimation results. A clear correlation was found between the initial parameter values and finding the global minimum. Increasing the chance of finding acceptable initial parameter values thus decreases the local minima problem. The local minima problem is also related to the gradient-based GN output error method used in this study. Other methods such as convexification,[12] a random search, interval analysis,[14] or simulated annealing could help resolve the local minima problem. The results in this study clearly show that the presence of expected levels of human operator remnant signal has a severe negative effect on the parameter estimation results. Therefore, the estimation method needs to be adjusted to be able to cope with these remnant signal levels. For future research it is thus recommended to study a filter error method[15] which combines the output error method with a Kalman filter. Although a filter error method is mathematical more complex

and computationally intensive, it seems to be a promising method for solving the local minima problem encountered in this study. The filter error method can also help with estimating part of the remnant, which might result in better parameter estimates.

Finally, simulated data was used for the reason that it is easy to know what the true values–and thus the optimal parameter estimates–of the original dynamic system are. Comparing the results of simulated data generated with a Boltzmann sigmoid and a linear function leads to the conclusion that a different transition between single-integrator and double-integrator dynamics influences the proposed estimation method. Therefore, future research should also focus on using experimental data to understand the effect of the proposed MLE method on the difference between a real-life and a simulated system.

VI. Conclusions

In this paper a maximum likelihood estimation method is proposed for the estimation of linear time-varying pilot-vehicle system parameters in the time-domain. The parameter estimation is performed on simulated data of a closed-loop manual control task, which contains time-varying human operator dynamics as well as time-varying vehicle dynamics. The time-varying system behavior is modeled utilizing Boltzmann sigmoids. The proposed method is compared to a sliding time window maximum likelihood estimation method. Although the proposed method performs better than the sliding time window method, it is heavily influenced by the human operator remnant, by the choice of the initial parameter values, and by local minima. Because improvements can be made for each of these influential factors, future research is recommended.

References

[1]McRuer, D. T., Graham, D., Krendel, E. S., and Reisener Jr., W., "Human Pilot Dynamics in Compensatory Systems. Theory, Models, and Experiments with Controlled Element and Forcing Function Variations," Technical Report AFFDL-TR-65-15, Air Force Flight Dynamics Laboratory, Wright-Patterson AFB (OH), 1965.

[2]Nieuwenhuizen, F. M., Zaal, P. M. T., Mulder, M., Van Paassen, M. M., and Mulder, J. A., "Modeling Human Multichannel Perception and Control Using Linear Time-Invariant Models," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 4, July-August 2008, pp. 999–1013.

[3]Zaal, P. M. T., Pool, D. M., Chu, Q. P., Van Paassen, M. M., Mulder, M., and Mulder, J. A., "Modeling Human Multimodal Perception and Control Using Genetic Maximum Likelihood Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 4, July-August 2009, pp. 1089–1099.

[4]Mulder, M., Verspecht, T., Abbink, D. A., Van Paassen, M. M., Balderas S., D. C., Schouten, A., De Vlugt, E., and Mulder, M., "Identification of Time Variant Neuromuscular Admittance using Wavelets," *International Conference on Systems, Man, and Cybernetics (SMC), 2011 IEEE*, October 2011, pp. 1474–1480.

[5]Zaal, P. M. T. and Sweet, B. T., "Estimation of Time-Varying Pilot Model Parameters," *Proceedings of the AIAA Modeling and Simulation Technologies Conference, Portland, OR*, Vol. 1, August 2011, pp. 598–614.

[6]Hess, R. A., "A Preliminary Study of Human Pilot Dynamics in the Control of Time-Varying Systems," *Proceedings of the AIAA Modeling and Simulation Technologies Conference, Portland, OR*, Vol. 2, August 2011, pp. 788–829.

[7]Schoukens, J. and Dobrowiecki, T., "Design of Broadband Excitation Signals with a User Imposed Power Spectrum and Amplitude Distribution," *Instrumentation and Measurement Technology Conference, 1998. IMTC/98. Conference Proceedings. IEEE*, Vol. 2, May 1998, pp. 1002 –1005.

[8]De Vlugt, E., Schouten, A. C., and Van der Helm, F. C. T., "Closed-loop Multivariable System Identification for the Characterization of the Dynamic Arm Compliance using Continuous Force Disturbances: a Model Study," *Journal of Neuroscience Methods*, Vol. 122, No. 2, 2003, pp. 123–140.

[9]Aldrich, J., "R. A. Fisher and the Making of Maximum Likelihood 1912-1922," *Statistical Science*, Vol. 12, No. 3, 1997, pp. 162–176.

[10]Zaal, P. M. T., Pool, D. M., Van Paassen, M. M., Mulder, M., and Mulder, J. A., "Multimodal Pilot Model Identification in Real Flight," *Proceedings of the AIAA Modeling and Simulation Technologies Conference and Exhibit, Chicago, IL*, Vol. 2, August 2009, pp. 802–821.

[11]Chu, Q. P., *Maximum Likelihood Parameter Identification of Flexible Spacecraft*, Ph.D. Dissertation, Faculty of Aerospace Engineering, Delft University of Technology, 1987.

[12]Ljung, L., "Perspectives on System Identification," *Annual Reviews in Control*, Vol. 34, No. 1, March 2010, pp. 1–12.

[13]Fletcher, R., *Practical Methods of Optimization*, A Wiley-Interscience Publication, Wiley, 2nd ed., 2000.

[14]van Kampen, E., *Global Optimization using Interval Analysis. Interval Optimization for Aerospace Applications.*, Ph.D. Dissertation, Delft University of Technology, September 2010.

[15]Raol, J., Girija, G., and Singh, J., *Modelling and Parameter Estimation of Dynamic Systems*, IEE Control Engineering Series, Institution of Electrical Engineers, 2004.

Part II

Preliminary Thesis

Chapter 1

Introduction

Human manual vehicle control behavior is a nonlinear, time-varying, closed-loop process. The time-varying aspect of this behavior is the primary reason that traditional system identification methods are of limited use for contemporary research in comparison to adaptive methods. Proper system identification can help explain the effects of different perceptual modalities on a human's control behavior through the determination of model parameters (e.g. weighing gains and time delays) from experimental data. This knowledge can, for example, be used to design advanced manual control systems. It is thus important to investigate time-variant system identification techniques. The current challenge in the research area is to discover and understand suitable human control behavior parameter estimation methods to further quantify human time-varying manual control. In the past decades the Control and Simulation (C&S) division at DUT has been researching control systems for both aerospace and automotive applications. Effective time-variant parameter estimation methods can significantly help advance research done at the division. Recent research on pilot control behavior by Zaal [2011] and on time-variant neuromuscular admittance by Verspecht [2011] verify and advocate the need for time-varying system identification methods. To understand what aspects of time-varying system identification this thesis aspires to highlight, the research goals are described in Section 1-1 followed by the outline of this thesis in Section 1-2.

1-1 Preliminary Thesis Objectives

The main objective of this thesis is to use Maximum Likelihood Estimation (MLE), a time-domain parameter estimation method, to get an advanced understanding of time-varying parameter estimation and to study the effects of the human muscles tensing on a control device, which will be modeled as a time-variant system.

1-2 Preliminary Thesis Outline

This preliminary report provides the reader with a model proposal for the simulations to be done and with background information required to understand the model. In Chapter 2, a concise overview of relevant system identification theory will be provided. Chapter 3 gives a more in-depth discourse on the MLE system identification and parameter estimation method. An overview of relevant literature is given in Chapter 4. The model proposed for the simulations that are done for this research and the associated considerations are specified in Chapter 5. Finally, some initial conclusions will be drawn in Chapter 6.

Chapter 2

System Identification Fundamentals

To clarify why the research in this report focuses on MLE as a system identification technique, and to give some practical background information on system identification in general, this chapter provides an overview of commonly used mathematical system identification notions necessary for the research in this thesis. Note that it is not intended to give a complete overview of the field in this chapter.

System identification techniques are used to develop parametric models of dynamic systems by solely using experimental data sequences of such a system's input and output quantities [Ledin, 2003]. Two renowned contributions to the research area of system identification are [Ljung, 1999] and [Pintelon & Schoukens, 2001], which are recommended for readers interested in more elaborate and in-depth theories. The information provided in this chapter is divided into three sections. In Section 2-1 the fundamental differences between nonlinear systems and a variety of linear system classes will be described. The important difference between open-loop and closed-loop systems within system identification is discussed in Section 2-2. Section 2-3 explains the differences between working in the frequency-domain and time-domain for parameter estimation methods and gives a concise overview of parametric and nonparametric methods in both domains.

2-1 Nonlinear and Linear Systems

In this section the differences between *nonlinear* and *linear* systems will be described. As stated before, human manual vehicle control behavior is a nonlinear, time-varying, closed-loop process. Contemporary system identification research methods frequently consider this process as (quasi-)linear and time-invariant. As in [Nieuwenhuizen et al., 2008] the process is often modeled with Linear Time-Invariant (LTI) models, such as the parametric Autoregressive Exogenous (ARX) model

(Time-Varying Autoregressive Exogenous (TVARX) models also exist). However, as mentioned in [Zaal et al., 2009] the accuracy of LTI models is often affected by biases and the control tasks require highly specific demands. Therefore, it is useful to explore whether or not other models, such as time-varying or parameter-varying ones, might be able to reduce these shortcomings. For the system identification research in this report it is helpful to investigate various system classes: those for nonlinear and linear systems. The state-space description of a nonlinear system with state variables $x(t)$, input $u(t)$ and time t can be written as:

$$\dot{x}(t) = f(x(t), u(t), t) \tag{2-1}$$

$$y(t) = g(x(t), u(t), t) \tag{2-2}$$

where $\dot{x}(t)$ is referred to as the *state equation* and $y(t)$ as the *output equation*. The input signal u in this chapter should not be confused with the pilot control output u used in chapter 1. For the prevalent LTI class the state-space equations are

$$f(x(t), u(t), t) = Ax(t) + Bu(t) \tag{2-3}$$

$$g(x(t), u(t), t) = Cx(t) + Du(t) \tag{2-4}$$

where the A, B, C and D matrices define the system. Compare Equations (2-3) and (2-4) with the set of Equations for an Linear Time-Varying (LTV) system:

$$f(x(t), u(t), t) = A(t)x(t) + B(t)u(t) \tag{2-5}$$

$$g(x(t), u(t), t) = C(t)x(t) + D(t)u(t) \tag{2-6}$$

where the differences with an LTI system obviously are the time-dependent state-space matrices $A(t)$, $B(t)$, $C(t)$ and $D(t)$. LTV systems are created when a nonlinear system is linearized about a time-varying operating point [Sanyal et al., 2005]. Finally, for the Linear Parameter-Varying (LPV) class the state-space matrices are dependent on a time-varying parameter vector $\Theta(t)$, as becomes clear in Equations (2-7) and (2-8).

$$f(x(t), u(t), \Theta(t), t) = A(\Theta(t))x(t) + B(\Theta(t))u(t) \tag{2-7}$$

$$g(x(t), u(t), \Theta(t), t) = C(\Theta(t))x(t) + D(\Theta(t))u(t) \tag{2-8}$$

The state-space equations for the LTI, LTV and LPV classes as discussed here, are based on those mentioned in [Apkarian et al., 1993]. Another significant observation made in this

article is that for utilization of LTI and LTV classes the state-space data A, B, C and D or $A(t)$, $B(t)$, $C(t)$ and $D(t)$ need to be known beforehand, whereas the LPV class relies on information of the time-varying parameter vector $\Theta(t)$.

2-2 Open-Loop and Closed-Loop Control Task

In Figure 2-1 the general *closed-loop* process for a human operator controlling a vehicle is shown. Here the operator is actively controlling the vehicle dynamics H_c, while following a target signal f_t. This forcing function f_t is used for excitation of the system, and is usually modeled as a multisine signal. The error signal e between the target forcing function f_t and the output y can be displayed on a screen, so that the operator can see the error, minimize it and ultimately optimize his control behavior. The operator's control signal u consists of the linear response H_p, which is combined with a remnant n, a white noise signal, to account for nonlinear behavior.

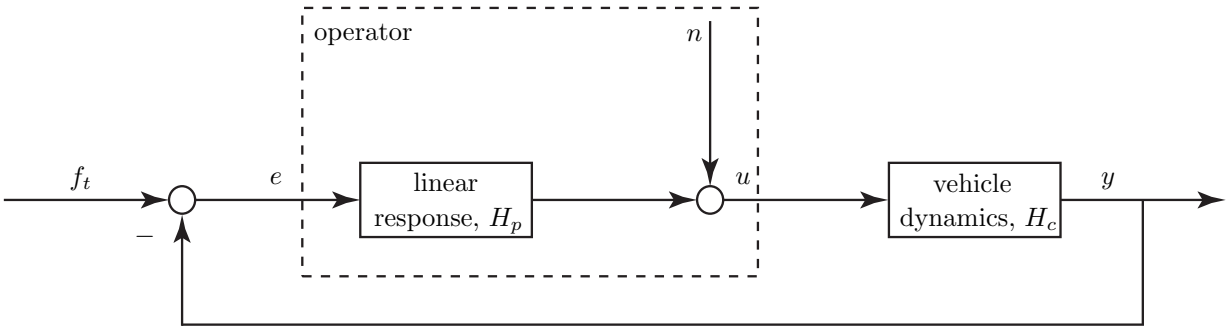


Figure 2-1: Closed-loop manual control task.

With the system identification techniques, the operator part of this simulation is particularly of interest. This is enclosed in the dashed line in Figure 2-1, and is represented as an *open-loop* process in Figure 2-2. Note that the error e and the operator control output signal u are uncorrelated in the open-loop process, while they are correlated in the closed-loop process due to the feedback loop. Because of this correlation difference, and the application of parameter estimation on the open-loop, while actually working in a closed-loop model, it is important to check whether or not the results from the open-loop output error methods are realistic within the closed-loop.

2-3 Frequency- and Time-Domain Parameter Estimation

This section contains a discussion on the utilization of parameter estimation methods. The advantages and disadvantages of system identification in the frequency-domain and the time-domain will be compared. Also, an initial explanation will be given on the choice

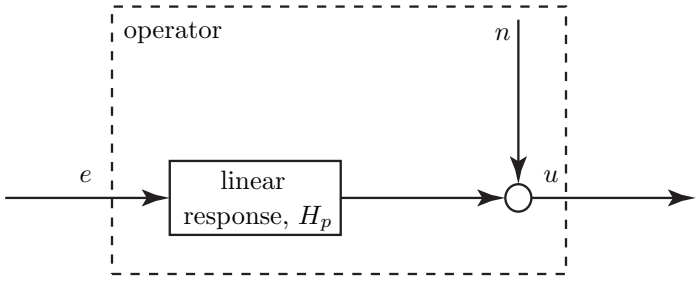


Figure 2-2: The output error open-loop process.

of MLE for the research in this report. Although nonparametric estimation methods are included in an overview of methods in this section, they will not be discussed in detail, because the research focuses on parametric MLE.

System identification can be performed in two complementary domains: in the *frequency-domain* and in the *time-domain*. One of the main advantages of the frequency-domain mentioned by Ljung [2007] is that it is, in certain cases, able to handle continuous-time data, which is not directly possible in the time-domain. Two other important advantages are that no *a priori* information, such as system order and sources of noise, is needed, and that spectral analysis in the frequency domain can be done with relatively fast computation.

In the time-domain, time measurements are sampled and represented by discrete values. A key advantage of the time-domain is that the noise can be separated from signals, due to the utilization of *a priori* information. Moreover, as noted by Ljung & Glover [1981] the time-domain offers a wide range of different models, because almost any *a priori* information about the system can be used to select a model set with a small number of estimation parameters relative to the number of parameters needed to describe a frequency response function. Although this limited number of parameters might lead to more complex estimation procedures, it oftentimes is one of the main reasons why identification in the time-domain is preferred over the identification in the frequency-domain. From these advantages and disadvantages it becomes clear that there is a delicate balance in considering either the frequency-domain or the time-domain in system identification in terms of the main purpose of the system under consideration.

Beside the consideration of identification in one of the domains, a relevant discrimination can be made of parameter estimation in the frequency-domain or parameter estimation in the time-domain as shown in Figure 2-3. This figure shows the steps needed for parameter estimation of pilot models. From [Zaal et al., 2009] and [Zaal & Sweet, 2011] it becomes clear that when fitting of a parametric model is desired, frequency-domain techniques, such as wavelet transformation, require two steps. First, a frequency response estimation is done in the time-domain, followed by the parameter estimation in the frequency-domain. On the other hand, parameter estimation can be performed immediately in the time-domain with time-domain techniques such as MLE. Direct estimation in the time-domain is deemed advantageous, but the Fourier Coefficients (FC) used to estimate the nonparametric

frequency response in the additional step can function as a useful tool to check if the results obtained from time-domain methods are correct.

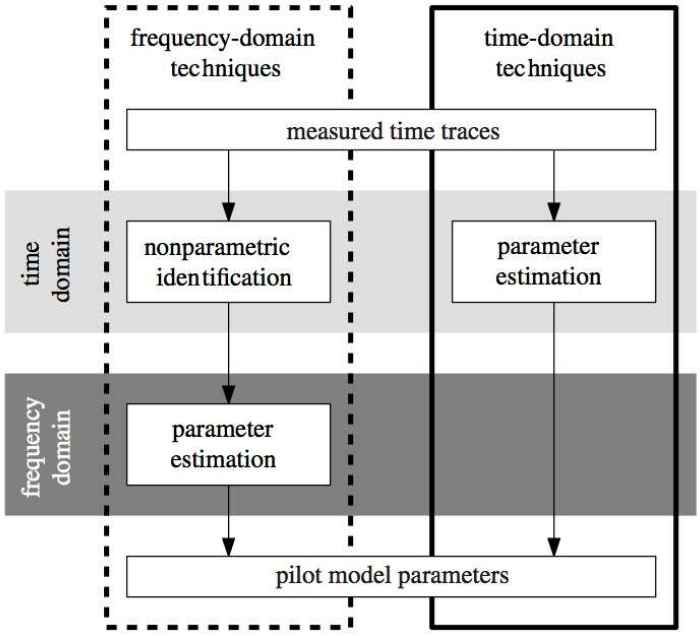


Figure 2-3: Comparison of pilot model parameter estimation methods. [Zaal et al., 2009].

A great number of different methods exist within system identification to perform estimation. These methods have been extensively described in system identification literature, but no clear overview is given due to the ambiguity of some terms used. Moreover, it is difficult to group the methods correctly: beside nonparametric and parametric methods, semiparametric and hybrid methods have also been proposed. In Table 2-1 an attempt is made to provide a general overview of estimation methods used for nonparametric estimation and parametric estimation in the frequency-domain and time-domain. The methods in the table are described by Ljung [1999] and Pintelon & Schoukens [2001]. Fan & Yao [2005] also provide relevant, in-depth information on the individual methods in the table. As the research in this thesis focuses on parameter estimation in the time-domain, Table 2-1 shows that making the choice of working with parameter estimation methods in the time-domain, one is left with just two options: the MLE and Least Squares Estimation (LSE) methods. LSE and MLE are related in the sense that LSE can be seen as a form of MLE when the experimental prediction errors have a normal distribution. In this thesis only MLE will be evaluated. In chapter 3 a more extensive explanation of the MLE method will be given.

Table 2-1: Overview of the utilization of nonparametric and parametric estimation methods in the frequency- and time-domain.

	Frequency-Domain	Time-Domain
Nonparametric Methods	Impulse Response Analysis Step Response Analysis Correlation Analysis Spectral Analysis Wavelet Analysis	Impulse Response Analysis Step Response Analysis Correlation Analysis Direct Estimation Maximum Likelihood Estimation Least Squares Estimation
Parametric Methods	Maximum Likelihood Estimation Least Squares Estimation Spectral Analysis	Maximum Likelihood Estimation Least Squares Estimation

Chapter 3

Maximum Likelihood Estimation

Maximum Likelihood Estimation is a statistical parameter identification method introduced by Ronald Aylmer Fisher in 1912. In this study, it will be applied to estimate parameters of dynamic models. Both [Zaal et al., 2009] and [Zaal & Sweet, 2011] provide elaborate descriptions of the method, which will be used to introduce the MLE method in Section 3-1. The quality assessment of the MLE can be done with the Cramér-Rao Inequality discussed in Section 3-2. For optimization of estimation methods a variety of smoothing algorithms is available as clearly depicted by Morrison [1969]. In Section 3-3 the Gauss-Newton (GN) algorithm is described for optimization of the MLE method, because this gradient-based optimization method is the classical approach for solving MLE problems [Zaal et al., 2009].

3-1 Definition of the Method

The first step that needs to be taken to find an estimate of model parameters with MLE, is the definition of a parameter vector Θ as in Equation (3-1).

$$\Theta = [unknown\ parameters]^\top \tag{3-1}$$

All unknown model parameters that are intended to be estimated, should be stated in this parameter vector. Next, the estimate $\hat{\Theta}$ of parameter vector Θ can be found by maximizing the likelihood function $L(\Theta)$ in Equation (3-2), which is defined as the joint conditional Probability Density Function (PDF) of the prediction error for m measurements of a control signal $u(k)$.

$$L(\Theta) = f(\epsilon(1), \epsilon(2), \ldots, \epsilon(k), \ldots, \epsilon(m) \mid \Theta) \tag{3-2}$$

In Equation (3-2) the *residuals* or prediction error $\epsilon(k)$ is the difference between the measured control signal $u(k)$ and the estimated control signal $\hat{u}(k)$ at discrete instants k. Assuming the measurement noise n is an additive zero-mean Gaussian white noise signal (this assumption will be explained in Section 3-2), the conditional probability density function for one measurement of prediction error $\epsilon(k)$ is given in Equation (3-3).

$$f(\epsilon(k) \mid \Theta) = \frac{1}{\sqrt{2\pi {\sigma_n}^2}} e^{-\frac{\epsilon^2(k)}{2\sigma_n^2}} \tag{3-3}$$

In this Equation σ_n is the standard deviation for the measurement noise signal n. To maximize the conditional probability density function, the negative natural logarithm is minimized, which provides a more straightforward optimization problem. When a global minimum of the negative log-likelihood is found, the resulting parameter vector $\hat{\Theta}_{ML}$ is the maximum likelihood estimate. Doing this for all prediction errors in $L(\Theta)$ results in Equation (3-4).

$$\hat{\Theta}_{ML} = \arg\min_{\Theta} \left[-\ln L(\Theta)\right] = \arg\min_{\Theta} \left[\frac{m}{2} \ln {\sigma_n}^2 + \frac{1}{2\sigma_n^2} \sum_{k=1}^{m} \epsilon^2(k)\right] \tag{3-4}$$

As noted by Zaal et al. [2009] Equation (3-4) summarizes the parameter estimation problem with MLE. Depending on the application (e.g. model or data) Equation (3-4) can represent a linear or a highly nonlinear optimization problem.

3-2 The Cramér-Rao Inequality

The quality of the identified parameters obtained with an MLE method can be assessed with the Cramér-Rao Inequality. If the additive measurement noise n from Figures 2-1 and 2-2 has an unbiased Gaussian distribution, it is possible to use the Mean Square Error (MSE) matrix, which is formulated as the probability in Equation (3-5) according to [Ljung, 1999].

$$P = E\left(\left[\hat{\Theta}(\epsilon(k)) - \Theta_0\right]\left[\hat{\Theta}(\epsilon(k)) - \Theta_0\right]^\top\right) \tag{3-5}$$

Here $\hat{\Theta}$ is the estimated parameter vector and Θ_0 is the true value of the parameter vector. The probability in Equation (3-5) can only be calculated if the true value Θ_0 of the parameter vector is known, and thus imposes a prerequisite for using the MSE matrix. The Cramér-Rao Inequality states that the covariance matrix of any unbiased estimator is at least as high as the inverse of the Fisher Information Matrix (FIM) $M_{\Theta\Theta}$ as shown in Equation (3-6). An unbiased estimator of which the variance asymptotically approaches this so-called Cramér-Rao Lower Bound (CRLB) when the sample size increases, is said

to be efficient. Because of this, MLE is conceived as the best possible estimator for large sample sizes. The FIM $M_{\Theta\Theta}$ is stated in Equation (3-7), is symmetrical, and should be positive definite, meaning that the matrix has full rank, to be inverted for the GN optimization described in Section 3-3.

$$E\left(\left[\hat{\Theta}(\epsilon(k)) - \Theta_0\right]\left[\hat{\Theta}(\epsilon(k)) - \Theta_0\right]^\top\right) \geq M_{\Theta\Theta}^{-1} \tag{3-6}$$

$$\begin{aligned} M_{\Theta\Theta} = E\left(\left[\frac{\partial}{\partial\Theta}\ln L(\Theta)\right]\left[\frac{\partial}{\partial\Theta}\ln L(\Theta)\right]^\top\right)\Bigg|_{\Theta=\Theta_0} = \\ -E\left[\frac{\partial^2}{\partial\Theta^2}\ln L(\Theta)\right]\Bigg|_{\Theta=\Theta_0} \end{aligned} \tag{3-7}$$

When the joint conditional PDF of Equation (3-3) is substituted in the likelihood function $L(\Theta)$ of Equation (3-7), the relation in Equation (3-8) can be obtained for the FIM, as stated in [Zaal et al., 2009].

$$M_{\Theta\Theta} = \frac{1}{\sigma_n^2}\sum_{k=1}^{m}\left(\frac{\partial\epsilon(k)}{\partial\Theta}\right)^2 \tag{3-8}$$

3-3 Gauss-Newton Optimization

In this section the GN algorithm will be introduced to optimize parameter estimation of unconstrained problems with MLE. As mentioned in [Chu, 1987] the GN method is probably one of the most used algorithms in combination with MLE and further optimizes the negative log-likelihood of the likelihood function L. It is a gradient-based method and Equation (3-9) is its iterative parameter estimate update equation.

$$\hat{\Theta}(k+1) = \hat{\Theta}(k) - \alpha(k)M_{\Theta\Theta}^{-1}(\hat{\Theta}(k))\frac{\partial L(\hat{\Theta}(k))}{\partial\Theta} \tag{3-9}$$

The line-search parameter α in this equation has a value $0 \leq \alpha \leq 1$, and is updated before each iteration to ensure optimal minimization of the likelihood function L along the vector defined by $M_{\Theta\Theta}^{-1}(\hat{\Theta}(k))\frac{\partial L(\hat{\Theta}(k))}{\partial\Theta}$. The nonlinear estimation problems under consideration in this thesis yield a high number of local minima [Ljung, 2010]. The GN algorithm is very sensitive to these local minima in the sense that when it converges to a certain local minimum, finding the global minimum is impossible. The incorporation of a line-search in

the GN method is advocated in [Fletcher, 2000] to increase the chance of convergence to the global minimum. Fletcher [2000] mentions two problems related to utilization of the GN algorithm. First of all, rank deficiency can occur in the state-space matrices of the system, which may cause the algorithm to fail, for example by converging to a non-stationary limit-point. A second cause of failure or extremely slow convergence is contributed to large residual problems (i.e. large initial parameter estimates), which primarily affect nonlinear problems with small-sized samples. Despite these problems, the line-search GN method remains a straightforward steepest-descent approach and has, up until now, provided satisfactory results in combination with MLE. Hybrid methods with the GN algorithm can also be used to obtain better results. In the case a hybrid method is used, the GN descent is usually applied in the final stage of optimization [Grouffaud et al., 1997].

Chapter 4

Literature on Parameter Estimation

An assortment of papers will be considered in this chapter to provide a background for the research this thesis is contributing to. Blanchard [2010] very accurately describes the importance of parameter estimation: *Parameter estimation is an important problem, because in many instances parameters cannot be physically measured, or cannot be measured with sufficient accuracy in real time applications. Rather, parameter values must be inferred from available measurements of different aspects of the system response.* In Section 4-1 parameter estimation methods are discussed based on a numerous amount of papers. Section 4-2 starts with a concise history of the research area of pilot control behavior and gives a short description of three papers that are essential for the further research done in this thesis considering pilot control behavior. In each of these three sections the model used, the parameter estimation method and the results of the respective papers will be discussed. The conclusions on the literature study in this chapter are given in Section 4-3.

4-1 Time-Varying Parameter Estimation Methods

Parameter estimation is oftentimes subject to a high computational load, due to the large amount of parameters that needs to be estimated. This is especially true for LTV systems, where the linearization of time-varying parameters add to the amount of parameters that need to be estimated. An assortment of search and selection algorithms has thus been developed to determine which parameters are most significant for a good estimation. Due to extensive research done in the field of parameter estimation in LTV systems, a considerable amount of literature has been written. In this section the most important parameter estimation methods and their aspects will be given with a focus on methods for LTV systems. To solve basic parameter estimation problems a wide variety of methods has

been proposed. In Section 4-1-1 an indirect estimation method is discussed: orthogonal functions. Direct parameter estimation methods for LTV systems are also plentiful. Bayesian approaches are often used for parameter estimation, where PDFs are being considered representations of uncertainty [Blanchard, 2010]. Chan & Zhang [2011] broadly classify the direct estimation methods into three categories: adaptive filtering or Kalman filtering (Section 4-1-2 and Section 4-1-3 respectively), Basis Expansion Modeling (BEM) (Section 4-1-4), and Weighted Least Squares (WLS) (Section 4-1-5).

4-1-1 Orthogonal Functions

One of the most comprehensively discussed methods is the utilization of orthogonal functions. This can be traced back to Chen & Hsiao [1975] and Rao & Sivakumar [1975] who used Walsh functions for parameter estimation in LTI systems, where Tzafestas [1978] applies Walsh functions to a time-varying system with parameters modeled as a finite-order polynomial. The main advantage of orthogonal functions is that they transform the integration of signals into a simpler integration of these functions by making use of the so-called *operational matrix of integration* of the corresponding orthogonal functions [Rémond et al., 2008]. The orthogonal functions in Table 4-1 are mainly used for parameter estimation and mainly focus on LTI systems.

Table 4-1: Overview of frequently used orthogonal functions for parameter estimation with corresponding references to some examples of research.

Orthogonal Functions	Examples of Research
Block-Pulse functions	Kwong & Chen [1981] Palanisamy & Bhattacharya [1981] Hwang & Guo [1984a]
Chebyshev polynomials	Paraskevopoulos [1983] Paraskevopoulos & Kekkeris [1983] Shih [1983]
Fourier series	Paraskevopoulos et al. [1985]
Hermite polynomials	Paraskevopoulos & Kekkeris [1988]
Laguerre polynomials	Paraskevopoulos & King [1976] Clement [1982] Hwang & Shih [1982]
Legendre polynomials	Chang & Wang [1983] Hwang & Guo [1984b] Paraskevopoulos [1985]
Walsh functions	Rao & Sivakumar [1975] Tzafestas [1978] Chen & Hsiao [1975]

Many of the functions used for parameter identification turn out to be (hypergeometric)

orthogonal polynomials. These polynomials are applied to solve the state equations of a linear system. The state and forcing functions of the system identification problem are usually replaced by a series of one of these polynomials together with their expansion coefficients in the differential equations. The differential equations of the state space system which govern the dynamic behavior of the system are then transformed in a set of algebraic equations which are relatively easy to solve. This transformation is the main difficulty, which is solved by the integration properties of the orthogonal functions. Because of the importance of orthogonal polynomials in parameter estimation research and because of the opaqueness of the links between these polynomials, Appendix A provides a concise overview of how these polynomials relate to each other. In more recent research, many of these orthogonal functions have been applied to parameter estimation in mechanical systems, for example in [Pacheco & Steffen Jr., 2002], [Rémond et al., 2008] and [Morais et al., 2008]. Due to the transformation performed by the orthogonal functions, the functions can be classified as an indirect method of parameter estimation. Orthogonal functions are also frequently used in time-varying nonlinear estimation problems in addition to the Volterra series, Wiener series and Polynomial Chaos (PC), three other nonlinear estimation methods.

4-1-2 Adaptive Estimation

Adaptive estimation (or filtering) makes use of past measurements, which is exactly the reason that the convergence speed is limited. Examples of adaptive estimation are the Least Mean Squares (LMS) and Recursive Least Squares (RLS) methods, which estimate parameters recursively from the input and measured output. As discussed in Chapter 2 LSE can be seen as a form of MLE when the errors have a normal distribution. In this broad classification MLE thus also belongs to adaptive estimation. As MLE was already discussed in Chapter 3 it will not be further discussed in this section. In more recent years, Local Polynomial Modeling (LPM) techniques have successfully been introduced to the field of engineering, for example by Y. Zhu & Pagilla [2003] and Chan & Zhang [2011]. LPM was originally applied as a nonparametric method, but these papers show that it can be a useful method for linear parametric systems as well. Y. Zhu & Pagilla [2003] give a clear explanation of how the method works: *Local polynomial approximation in a finite time interval is used to represent the unknown time-varying parameters. The coefficients of the polynomials are estimated locally instead of the unknown time-varying parameter. The accuracy of the approximation depends on the order of the polynomial and the width of the time interval, which can be chosen.* The paper also shows how LSE and gradient algorithms can be modified to include the LPM method. The results of [Chan & Zhang, 2011], which uses LPM combined with a Variable Bandwidth Selection (VBS), show superior performance in comparison with conventional recursive LSE and generalized random walk Kalman Filters (the latter will be discussed in Section 4-1-3). Because of the positive results and the possibility to augment the LPM method, it might be useful to see if this can be implemented for the MLE method. In this section the most prevalent

methods of adaptive estimation have been discussed. Other methods for LPV system identification have also been proposed. Some examples are: modeling parameters with smoothing splines [H. Zhu & Wu, 2007], wavelet methods [Verspecht, 2011], ensemble methods [MacNeil et al., 1992], neural network methods [J. Guo & Dong, 2011], forgetting factor methods [Asutkar et al., 2009a] and hybrid methods.

4-1-3 Kalman Filtering

The Kalman Filter (KF) is an optimal recursive estimator for linear systems with Gaussian noise based on Linear Dynamical System (LDS) theory. The KF originated as an alternative to the static Wiener Filter (WF). It is a Bayesian method, and estimates the state of a system using the state-space method from the input measurement data under influence of noise. Important contributions to the research on KFs include [Nahi, 1969], [Brown, 1983] and [Gelb, 1999]. Another type of filter is the Extended Kalman Filter (EKF), which is used for parameter estimation in nonlinear systems. A qualitative study on the properties of KFs is done by L. Guo [1990], who notes that the parameter tracking errors with a KF are small when parameter variation and noise are small. The main drawback of KFs and EKFs is that they do not give good results when the parameters are varying too fast, which is usually the case with a human operator. A solution to this problem has been proposed by Wittenmark [1979] by combining a conventional KF with a coarse estimator to estimate time-varying parameters. Asutkar et al. [2009b] also show that fast-varying parameters can successfully be estimated using a KF when the system is modeled as a TVARX system. Another problem often encountered in parameter estimation is the large amount of data that has to be processed, resulting in heavy computational loads to solve the problem. The Ensemble Kalman Filter (EnKF) is designed with Monte Carlo approximation theory in such a way that it can handle large problems [Evensen, 2009]. KFs are also often combined with generalized random walks, which describe the variations of the system state [Chan & Zhang, 2011]. These generalized random walk KFs are frequently used in real applications because of their simplicity and efficiency.

4-1-4 Basis Expansion Modeling

In BEM, time-varying parameters are expressed as a weighted sum of basis vectors and approximated by a linear combination of known basis functions. The basis functions used for the estimation are dominant in terms of the performance of the estimation [Chan & Zhang, 2011]. An optimal selection is, however, not always possible and the basis functions usually have to be chosen before the identification. An advantage of the BEM is that it is able to handle fast-varying parameters. In [Sanyal et al., 2005] temporal basis expansion approaches are used for parameter estimation in an LTV system: the dynamic stiffness of the elbow is estimated. The authors note that a major disadvantage of BEM is

the large increase in the number of parameters that must be estimated. A term selection algorithm, variously called Least Absolute Shrinkage and Selection Operator (LASSO) or Basis Pursuit Denoising (BPDN), eliminates unnecessary terms and is used to mitigate this large number of parameters. In [Sanyal et al., 2005] a smooth variation of the parameters is achieved by modeling the variations with orthogonal polynomials such as the Chebyshev or Legendre polynomial (also refer to Appendix A). Periodic variations are then expanded onto sinusoids, and sharp variations of the parameters are modeled through Walsh functions or Haar wavelets (square-shaped wavelets). The LASSO algorithm functioned correctly, but unfortunately the setup of this BEM experiment led to experimental data lacking time variations. A recommendation was to use Laguerre filters for the basis functions, which might be able to cope with the transient changes. Although more focused on adaptive estimators, Abu-Naser & Williamson [2006] discuss the convergence of basis functions and define the Persistent Excitation (PE) condition. This PE condition together with passivity are requirements for the basis function algorithms to converge. That these conditions hold for LTV systems is shown in [Abu-Naser & Williamson, 2007].

4-1-5 Weighted Least Squares

The WLS method is similar to conventional LSE, but employs kernels or windows to assign larger weights to local data and smaller weights to remote data. The method is shortly discussed by Chan & Zhang [2011]. The time-varying parameters are estimated by minimizing a weighted sum-of-square estimation errors. Key to the performance of the WLS method is the selection of the window size or kernel bandwidth. Although attempts have been made to improve the WLS method [Campi, 1994], automatic data-driven kernel-bandwidth selection for WLS remains a difficult problem, which hinders its practical implementation.

4-2 Pilot Control Behavior

4-2-1 A Short History of Pilot Control Behavior

The research done in [McRuer et al., 1965] and [McRuer & Jex, 1967] on quasi-linear pilot models and the crossover theorem form the so-called cornerstones of the research on pilot control behavior. McRuer's ideas gradually developed during the 1970s, and other researchers contributed by focusing on single- and multi-loop identification methods in both the frequency- and time-domains. A focus on time series analysis, started in the 1970s, and continued in the early 80s, in for example, [Altschul et al., 1984] and [Biezad & Schmidt, 1984]. The research area, however, suffered from a lack of new, innovative contributions, due to the limitations in system identification theory, a shortfall of significant insights in human pilot behavior, and the modeling thereof. This continued until the 1990s when new

developments in the area of system identification by Ljung [1999] and Pintelon & Schoukens [2001] on the one hand, and a particular focus on pilot control behavior at DUT on the other, resulted in new contributions on topics such as model validation, identification with FC, and LTI models. Due to limitations of these methods, contemporary research now shows a shift from LTI models to LTV and LPV models. Nowadays, research is also being done on system identification methods that have not previously been combined with pilot model parameter estimation, such as LSE, wavelet, and MLE techniques.

4-2-2 Linear-Time Invariant Model Estimation

In this section, the paper of [Nieuwenhuizen et al., 2008] on human multichannel perception and control in LTI systems will be reviewed to provide a background for the perception and control with MLE, which will be discussed in Section 4-2-3. In the paper a new two-step identification method using LTI is compared with the conventional FC method. According to the paper, the use of FC introduces certain limitations *in terms of the resolution in the frequency domain, the variance of the identified frequency response functions, and the design of the forcing functions*, which are reduced with the LTI method. In Figure 4-1 the closed-loop manual control task is shown, with forcing function f_t, tracking error signal e, state signal x, error frequency response function H_{pe}, state frequency response function H_{px}, remnant signal n, control signal u, disturbance forcing function f_d and system dynamics H_c. The proposed identification method consists of the following steps:

1. Estimating the frequency response functions H_{pe} and H_{px};

2. Determining the multichannel model parameters and fitting them to the estimated frequency response functions.

As mentioned in Section 1-1, one of the goals of this thesis is to decrease the measurement time needed to come up with estimation results of equivalent quality. A relevant notion in this paper is that for estimation long measurement times are needed to observe sufficiently low frequencies.

The paper compares the proposed LTI identification method with the conventional FC, validates them offline with Monte Carlo simulations and applied to experimental data from the SIMONA Research Simulator (SRS). The LTI method turned out to be significantly better in estimating the pilot model parameters, due to the assumption of a model structure and the incorporation of the pilot remnant n. The proposed method was also better in handling higher levels of pilot remnant in comparison to the FC method, and also resulted in a lower variance of the parameter estimates.

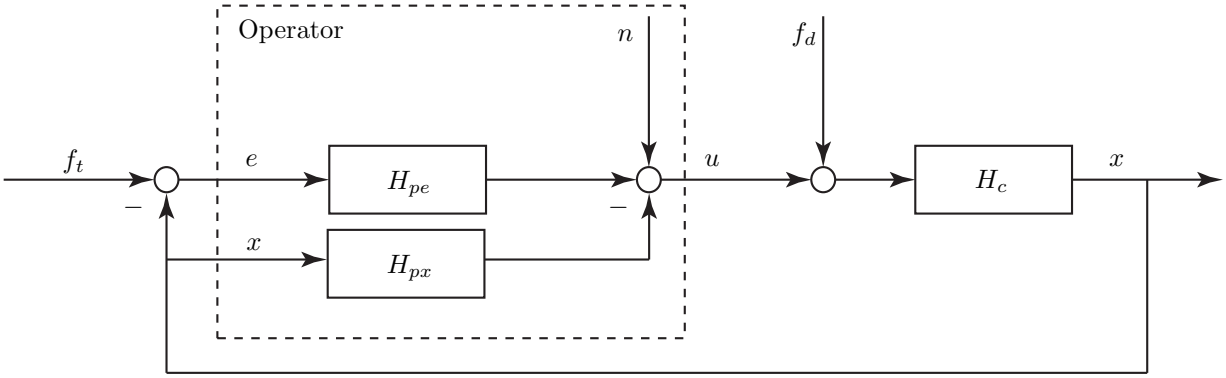


Figure 4-1: Multiloop closed-loop manual control task. Based on [Nieuwenhuizen et al., 2008].

4-2-3 Genetic Maximum Likelihood Estimation

Following the paper of [Nieuwenhuizen et al., 2008], the importance of researching new identification methods for human control behavior was underlined. In [Zaal et al., 2009] a new MLE method augmented with a Genetic Algorithm (GA) is introduced to increase the probability of finding the global minimum of the nonlinear optimization problem. The classical MLE method, uses a gradient-based algorithm, and is very dependent on the initial parameter guess. Experimental data of a disturbance-rejection task for aircraft pitch attitude control is used in this paper for application of the MLE method. In this section the paper will be discussed, and to illustrate the parameter estimation problem the multichannel pilot model will be described in detail. The closed-loop aircraft pitch control task is shown in Figure 4-2 and can easily be compared to Figure 4-1.

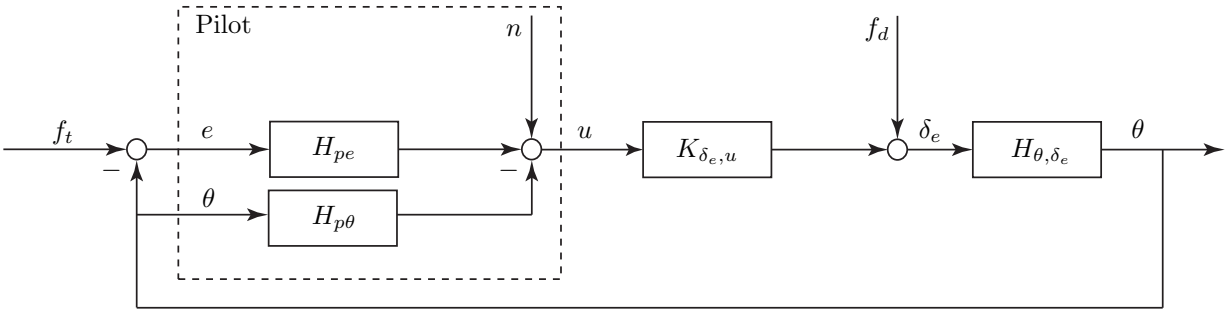


Figure 4-2: Multiloop closed-loop aircraft pitch control task. Based on [Zaal et al., 2009].

In this paper the MLE method is used as a time-domain technique, which results in less restrictions on the design of forcing functions. In order to use spectral analysis for comparison of the results, the decision was made to use the sums of 10 sine waves, so-called multisines, for the target forcing function f_t and disturbance forcing function f_d in Figure

4-2:

$$f_t(t) = \sum_{k=1}^{10} A_t(k) sin(\omega_t(k)t + \phi_t(k)) \tag{4-1}$$

$$f_d(t) = \sum_{k=1}^{10} A_d(k) sin(\omega_d(k)t + \phi_d(k)) \tag{4-2}$$

where A_t and A_d are the sinusoidal amplitudes, ω_t and ω_d the sinusoidal frequencies, and ϕ_t and ϕ_d the sinusoidal phase shifts. The pitch control dynamics H_{θ,δ_e} are controlled by the pilot, which consists of linear frequency response functions H_{pe} and $H_{p\theta}$ as in Equations (4-3) and (4-4).

$$H_{pe}(j\omega) = K_v \frac{(1 + j\omega T_{lead})^2}{(1 + j\omega T_{lag})} e^{-j\omega\tau_v} H_{nm}(j\omega) \tag{4-3}$$

$$H_{p\theta}(j\omega) = (j\omega)^2 H_{sc}(j\omega) K_m e^{-j\omega\tau_m} H_{nm}(j\omega) \tag{4-4}$$

Equation (4-3) describes the pilot response to visual motion cues and is similar to the error frequency response function H_p in Figure 2-1. The equation contains visual-perception gain K_v, visual-lead time constant T_{lead}, visual-lag time constant T_{lag} and visual-perception delay τ_v. The neuromuscular system is modeled as a second-order mass-spring-damper system with transfer function H_{nm} in Equation (4-5).

$$H_{nm}(j\omega) = \frac{\omega_{nm}^2}{\omega_{nm}^2 + 2\zeta_{nm}\omega_{nm} j\omega + (j\omega)^2} \tag{4-5}$$

Here ω_{nm} is the neuromuscular natural frequency and ζ_{nm} the neuromuscular damping ratio. The pitch-motion perception channel $H_{p\theta}$ in Equation (4-4), which in fact is the state frequency response H_{px} from Figure 4-1 for pitch, contains a transfer function for the semicircular canals of the vestibular system H_{sc}, a motion-perception gain K_m, and a motion-perception time delay τ_m. The semicircular-canal dynamics are given in the paper by Equation (4-6) with semicircular-canal time constants $T_{sc_1} = 0.11s$ and $T_{sc_2} = 5.9s$.

$$H_{sc}(j\omega) = \frac{1 + j\omega T_{sc_1}}{1 + j\omega T_{sc_2}} \tag{4-6}$$

Other relevant parameters in Figure 4-2 are the tracking error signal e, pitch angle θ, pilot remnant signal n, control signal u, pitch stick gain $K_{\delta_e,u}$, and elevator deflection δ_e. The parameter vector Θ in Equation (4-7) contains the eight parameters that will be estimated.

$$\Theta = [K_v \;\; T_{lead} \;\; T_{lag} \;\; \tau_v \;\; K_m \;\; \tau_m \;\; \zeta_{nm} \;\; \omega_{nm}]^\top \tag{4-7}$$

The MLE method is applied to the model given in Figure 4-2. To achieve a higher probability of finding the global minimum of the optimization problem a stochastic search technique, the GA, is used. Search techniques like the GA were developed to avoid problems *associated with local minima by searching broad regions of the parameter space at the cost of greatly increased convergence time* [Westwick & Kearney, 2003]. A GA is based on the principle of survival of the fittest, as considered in evolutionary biology, by application of random genetic functions such as selection, mating, crossover, and mutation. Due to these functions, the population evolves toward increasingly better solutions, which means they have a better fit to the optimization problem. In Figure 4-3 the GA is visualized based on the paper. A population of parameter sets is shown depicted by circles with different radii indicating their respective values. From this population a selection is made, where the members with highest fitness have higher probability of being selected. The parameters of each member are then coded into binary genes within their lower and upper bounds, which are taken to be 0 and 2 respectively in this example. For each parameter set two members mate. During mating random crossover of the genes occurs with a certain probability, usually set to 0.7. The offspring will then mutate with a certain probability, which is typically very low (0.01 in this case). After decoding all offspring from binary codes into real values, a new population is selected based on their fitness. The genetic optimization continues until a specified number of iterations have been performed. A bigger population results in more accurate results, but in that case the GA also needs more computational power. Therefore, the gene pool is usually limited in size and the resulting parameter estimates are relatively inaccurate. Furthermore, the GAs are not deterministic and provide different results each time they are used. Because of these reasons the paper indicates that GAs should not be used as the sole estimation method for pilot model parameter estimation.

To enhance the parameter estimates, the GN algorithm as described in Section 3-3 is used to get even closer to the global minimum. Application of the GA also prevents possible unstable behavior of the GN algorithm, which might occur with large initial parameter errors.

The genetic MLE method is evaluated by estimating parameters of a pilot model from an experiment in the SRS at DUT. The results in this paper establish that the augmentation of the GA significantly increases the usefulness of the GN algorithm, as the GN algorithm, on its own, only finds the global minimum in 20% of all tested initial conditions. While more computational power is needed with the GA, it ensures more accurate parameter estimates. On the other hand, GN optimization clearly increases the accuracy after the GA is applied. The genetic and GN algorithm thus create mutual benefits, and the global minimum was found in 90% of the cases. This paper also investigates the robustness of the method when introduced to increasing levels of pilot remnant. The results show that the average biases of the parameters stay within the 99% confidence interval, but increase for higher levels or remnant. This is partly attributed to the fact that for high levels of remnant the remnant is colored instead of pure white noise. The experimental results in the paper also show that the PDF of the remnant signal has an almost perfect Gaussian

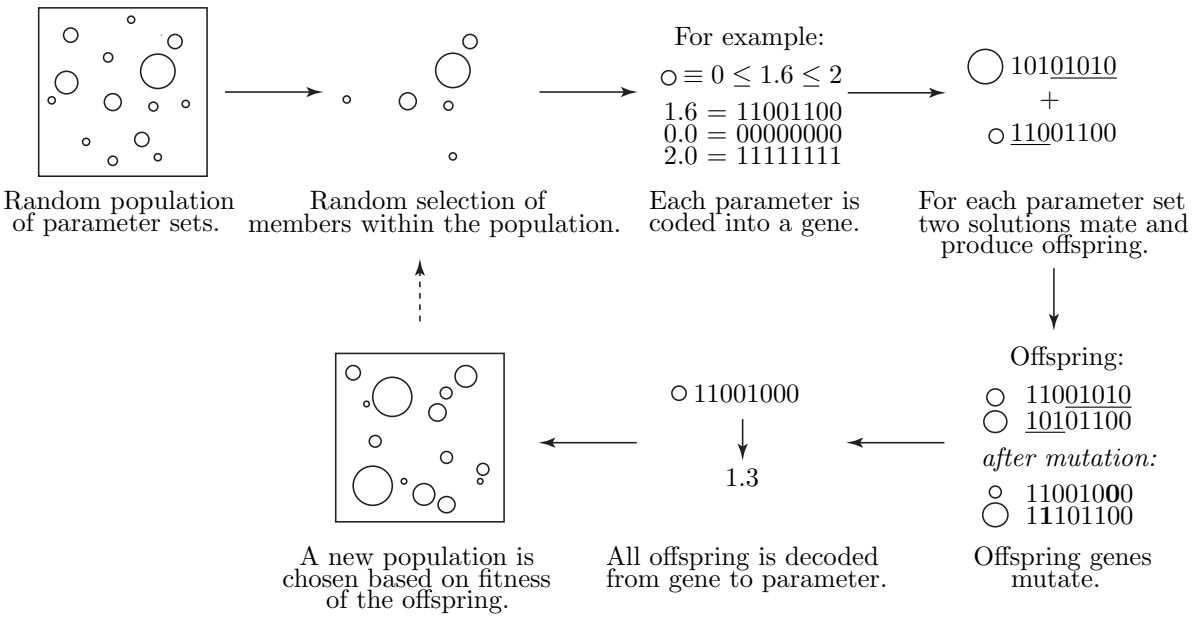


Figure 4-3: Schematic representation of the different steps of the genetic algorithm used to minimize the likelihood function. Based on [Zaal et al., 2009].

distribution with zero mean, which is necessary to use the MSE matrix as described in Section 3-2.

4-2-4 Estimating Time-Varying Pilot Model Parameters

To get closer to more realistic human control behavior [Zaal & Sweet, 2011] take a step away from the stationary human control elements and compare two methods for estimating time-varying parameters: a two-step wavelet method and a windowed MLE method. The aim of discussing this paper in this literature research is to qualitatively compare MLE with another parameter estimation method. The wavelet method uses the often used Morlet wavelet. The method is described in detail in the paper and it is not necessary to further elaborate on its mathematical theory in this chapter to provide a qualitative assessment. The closed-loop manual control task with time-dependent transfer function is shown in Figure 4-4.

The forcing function f_t is chosen to be designed as a multisine signal equal to Equation (4-1). This choice is straightforward, as it is easier to perform spectral analysis and compare the frequency-domain wavelet method with the time-domain MLE method. The time-varying vehicle dynamics $H_c(s,t)$ of the control task are given in Equation (4-8).

$$H_c(s,t) = \frac{K_c(t)}{T_c(t)s^2 + s} \tag{4-8}$$

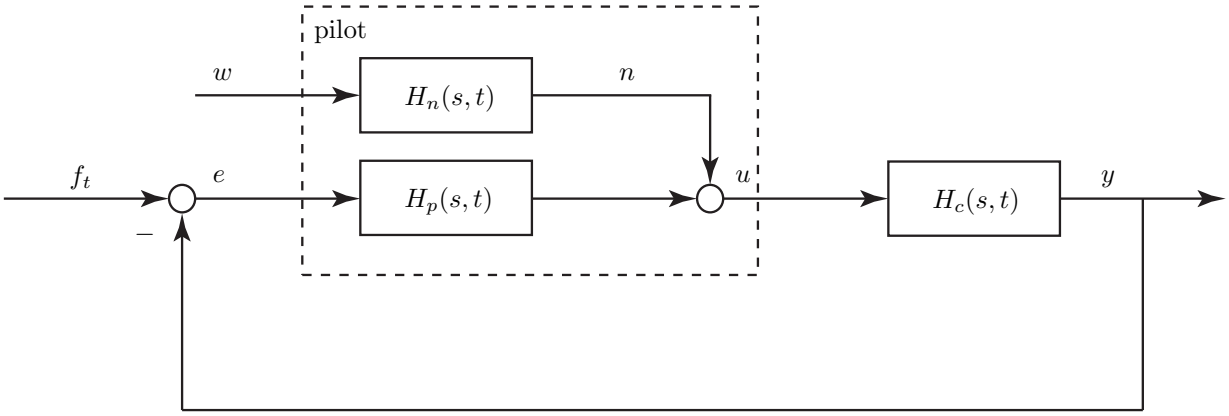


Figure 4-4: Simulated closed-loop compensatory control task. Based on [Zaal & Sweet, 2011].

Here K_c(t) is a time-varying gain and $T_c(t)$ a time-varying time constant. The vehicle dynamics act as a single integrator below the break frequency of $1/T_c(t)$ rad/s and as a double integrator above this frequency. The time-varying pilot dynamics are given by Equation (4-9), and are comparable to Equation (4-3).

$$H_p(s,t) = K_v(t)(1 + T_{lead}(t)s)e^{-s\tau_v}H_{nm}(s) \tag{4-9}$$

In Equation (4-9) the pilot visual gain $K_c(t)$ and the pilot lead time constant $T_{lead}(t)$ are time-dependent. To imitate a realistic time-varying pilot remnant n, an unbiased Gaussian white noise signal w is fed through the time-varying low-pass filter $H_n(s,t)$ as in Equation (4-10):

$$H_n(s,t) = \frac{K_n(t)}{T_n s + 1} \tag{4-10}$$

where $K_n(t)$ is a time-varying pilot remnant gain and T_n a fixed time constant. For the experiment, the pilot remnant gain was calculated to keep a certain pre-set power ratio between remnant and control signal ($P_n = \sigma_n^2/\sigma_u^2$) for the entire simulation. The other four time-varying parameters $K_v(t)$, $T_{lead}(t)$, $K_c(t)$ and $T_c(t)$ are kept constant for the first 30 s of the run, after which they either increase or decrease linearly in value for the next 40 s. At 70 s the parameters are kept at a constant value again. The to-be-estimated parameters are found in parameter vector Θ of Equation (4-11).

$$\Theta = [K_v(t) \;\; T_{lead}(t) \;\; \tau_v \;\; \zeta_{nm} \;\; \omega_{nm}]^\top \tag{4-11}$$

The windowed MLE method used in this study is performed at every time step t_i using a sliding time window of length Δt. Choosing a too small Δt negatively affects the accuracy of the parameter estimates, but a Δt too large has a negative effect on the detection of

small variations in pilot model parameters. In this paper 20 seconds is chosen as the length of the time window. To reduce computational efforts the GA as introduced in section 4-2-3 is omitted and only the GN algorithm is used. As discussed in earlier sections, the GN algorithm on its own is very dependent on the initial parameter errors, and is unable to recover when it becomes unstable under influence of large initial parameter errors. The initial parameter sets are constructed for every time step from the simulated pilot model parameters in this paper.

The two-step wavelet method and the windowed MLE method are assessed by applying them to simulated data. To evaluate the effect of the pilot remnant on the parameter estimates, simulations of 90 s are done with and without pilot remnant signal applied to the model. The results are compared by looking at the frequency responses and their respective magnitude and phase plots. For the case without pilot remnant, the wavelet method shows unwanted edge effects and oscillatory behavior in time at higher frequencies. Compared to the wavelet method, MLE has difficulty handling the rapid transition from constant to linear at 30 s. It shows a more gradual change, which is mainly attributed to the averaging of the values with the sliding time window. This effect is confirmed by data of the crossover frequency and phase margin. Because the rate of change of the vehicle dynamics is lower at 70 s, the MLE method accurately estimates the change at this particular instant. Adding the pilot remnant to the simulation resulted in significant adverse effects for the parameters estimated with the wavelet method, even for a low remnant-to-control-signal power ratio P_n. For higher levels of pilot remnant the bias and variance of the estimates of the wavelet method were too high. No parameter estimates could be obtained with the wavelet method, while the MLE algorithm produced reliable estimates. This paper opts to enhance the MLE method by estimating the parameters as a function of time directly to get around the slow response to rapid changing dynamics created by the sliding time window. While the two-step wavelet method did not respond well to pilot remnant, many things can still be changed to try to improve the method in more extensive research.

4-3 Literature Study Conclusions

From the literature study performed in this chapter several conclusions can be drawn. These conclusions will serve as a foundation for the further research done in this thesis. It has become clear that the research on pilot model parameter estimation can be traced back to the 1960s, and has evolved in the succeeding decades. In contemporary research two trends are identified. The first one is a shift towards time-varying and parameter-varying systems. The second one is a necessity of evaluating an assortment of parameter estimation methods to seek solutions for the limitations present-day scholars face in their research. Four categories of parameter estimation methods useful for LTV systems have been defined: orthogonal functions, adaptive estimation or KF, BEM, and WLS. From the overview this classification provides, it becomes clear that a variegated number of methods exist with their own advantages and disadvantages. MLE is an adaptive estimation method and it

would be wise for future studies to compare the performance of other methods with the results of the MLE method that will be obtained by this study. The pioneering work done in the past few years in the field of pilot model parameter estimation has led to practical insights. In [Nieuwenhuizen et al., 2008] LTI systems show a superior performance of estimating pilot model parameters compared to the utilization of FC. From [Zaal et al., 2009] it becomes clear that MLE in the time-domain is a promising algorithm for pilot model parameter estimation. Combined with a GA (computational intensive) and a GN algorithm (unstable on its own under influence of large initial parameter errors), the MLE algorithm has a high chance of finding the global minimum and consequently returns accurate estimates. To let the models realistically approach the human operator even more, [Zaal & Sweet, 2011] make a tentative attempt at investigating time-varying pilot model parameters. Their comparison of a two-step Morlet wavelet method and a windowed MLE method resulted in bad results for the wavelet method under influence of a pilot remnant, while the MLE provides accurate results. The drawback of the windowed MLE method is that the method fails to act on rapid changes of the time-varying parameters. The main reason for this is the use of a sliding time window, and not because of the MLE algorithm itself. Keeping these studies in mind, the MLE algorithm seems promising for extended research with nonlinear time-varying parameters.

Chapter 5

Methodology

In this chapter the scientific methodology for the research will be described in order to work towards the goals set in Section 1-1. Before going into detail, a practical roadmap is provided in Table 5-1 to exhibit the three steps that will be taken in the research.

Table 5-1: Roadmap: the three steps that will be taken to reach the thesis objectives.

Step 1	An open-loop neuromuscular model will be set up to generate data. One parameter will be modeled as a first-order polynomial. MLE will be applied to the generated data to estimate the parameters of the first-order polynomial. After these simulations the order of the polynomial will be increased.
Step 2	The open-loop neuromuscular model will be extended to an open-loop pilot model to generate data. In the beginning one parameter will be modeled as a polynomial. MLE will be applied to the generated data to estimate the parameters of the first-order polynomial. Depending on the results, more parameters will be modeled as polynomials. The pilot model will be extended to a closed-loop model.
Step 3	A multimodal pilot model will be made to generate data with a selection of time-varying parameters modeled as polynomials. MLE will be applied to the generated data to estimate the parameters of the first-order polynomial.

Step 2 and **Step 3** in Table 5-1 are more or less extensions of **Step 1** and are dependent on the results found during their respective preceding step. The execution and simulation of all three steps will be discussed in Chapter 7.

Chapter 6

Conclusions and Recommendations

In the past few years MLE has been investigated and has also been successfully applied to linear pilot models for parameter estimation in time-invariant and time-varying systems. The MLE method has shown to be reliable and consistent in comparison with classical system identification methods, and has shown to perform better than certain wavelet methods. However, no significant research into the estimation of pilot model parameters has been performed.

The purpose of this thesis is to get a better understanding of the performance of the MLE method when applied to simulated data of a human operator.

For immediate reaction of the control systems on certain control behavior, too much data is currently needed to accurately estimate the necessary parameters, and to subsequently determine what reaction of the system is needed. To achieve both goals, an initial open-loop model structure has been setup consisting of an multisine excitation signal, the neuromuscular system, an optional pilot remnant, and a simple set of vehicle dynamics. It is certain that further extension of this model will take place, but in what manner this will happen is dependent on the results of the initial simulation runs. The nonlinear parameters will be introduced in this first model by using third-order polynomials for parameters in the transfer function for the neuromuscular system. The MLE algorithm can be applied directly or via a sliding window method. The latter has shown to bring along unfavorable averaging effects, which makes windowed MLE less accurate compared to direct application. It is also decided to apply both the genetic and GN algorithms in the initial runs, to see what happens. If the genetic algorithm requires too much computational power, in terms of how fast the parameter estimation will take place, it might be considered to only use the GN algorithm in a later stage.

There are many ways of designing a pilot model. As the scope of this research is limited, a foundation for different recommendations has already manifested from the preliminary

research. Although the effect of different excitation functions on pilot model systems has already been researched in [Zaal et al., 2008], more research on the effects of the MLE method and different systems needs to be done. Also, more research should be done on the modeling of the pilot remnant signal. For basic research it might be enough to look at zero-mean Gaussian white noise signals, but as shown by Zaal & Sweet [2011] more advanced and accurate pilot remnant signals can be created for a more realistic model. Finally, it will be interesting to look at identification of LPV systems as the research area develops. For the fundamentals of LPV systems in general [Tóth, 2010] is a good reference. For current research looking at the research area of statistical learning theory might be beneficial to the research on MLE and optimization through regression. A starting point for this would be [Vapnik, 1998] and [Schölkopf & Smola, 2002].

Part III

Final Thesis

Chapter 7

Preliminary Simulation Model and Estimation Results

In this chapter the step-by-step process of setting up the simulation and the execution of the simulations will be described. In Section 7-1 the initial simulation model, with a time-varying neuromuscular frequency $\omega_{nm}(t)$, is described. An adjusted simulation is discussed in Section 7-2. For this adjusted simulation a system representing typical neuromuscular dynamics time-varying gain $K(t)$ is modeled using polynomial and sigmoid functions. To illustrate how the simulations operate, interim results will be discussed throughout this entire chapter. Note, however, that the results of the complete simulation will be discussed in Chapter 8.

7-1 Initial Simulation Model: A Time-Varying Neuromuscular Frequency

For **Step 1** as defined in Chapter 5, a simple single-input, single-output open-loop system model is proposed in Figure 7-1. This model has two purposes. First of all, it will act as a foundation on which, depending on the simulation results, further augmentation can be performed to make the model more complex. A second goal is to understand what happens at the core of the time-varying, parametric MLE. This is done by looking at how time-varying dynamics can be estimated using simulated data. The simulation will be set up in MATLAB® and Simulink®. In Chapter 4 it became clear that careful design of such a model is important. The crucial parts of this model are the control task with the corresponding target forcing function and the time-varying dynamics.

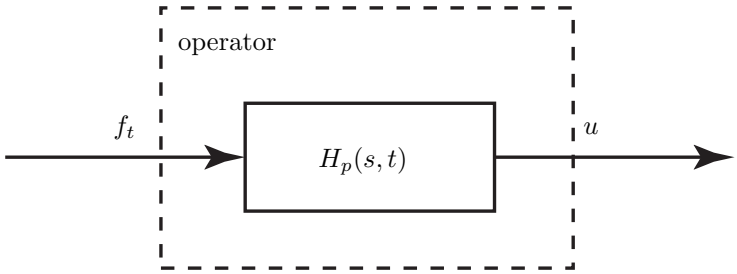


Figure 7-1: Open-loop control task limited to the dynamics of the operator. The addition of nonlinear human behavior to this model will be discussed in Section 7-1-1.

7-1-1 Generating Simulation Data

Before parameter estimation can be performed, data is needed to perform the estimation on. For the research in this thesis this data will be generated by simulating the model of Figure 7-1. This section describes how this data was generated. The parameter estimation of this data will be discussed in Chapter 7-1-2.

Defining the Forcing Function

The target forcing function f_t used to excite the system in Figure 7-1 is taken from Zaal et al. [2009]. This forcing function, a multisine, is based on sine waves with different frequencies as in Equation (4-1). The sine wave frequencies are defined as integer multiples of the experimental measurement time base frequency, $\omega_m = 2\pi/T_m$ with $T_m = 81.92$ seconds. Table 7-1 shows the values for integer factors n_t, target sinusoid frequency ω_t, target sinusoid amplitude A_t and target sinusoid phase shift ϕ_t. This multisine is plotted in Figure 7-2.

Table 7-1: Properties of multisine target forcing function f_t.

k, -	n_t, -	ω_t, rad/s	A_t, deg	ϕ_t, rad
1	6	0.460	0.698	1.288
2	13	0.997	0.488	6.089
3	27	2.071	0.220	5.507
4	41	3.145	0.119	1.734
5	53	4.065	0.080	2.019
6	73	5.599	0.049	0.441
7	103	7.900	0.031	5.175
8	139	10.661	0.023	3.415
9	194	14.880	0.018	1.066
10	229	17.564	0.016	3.479

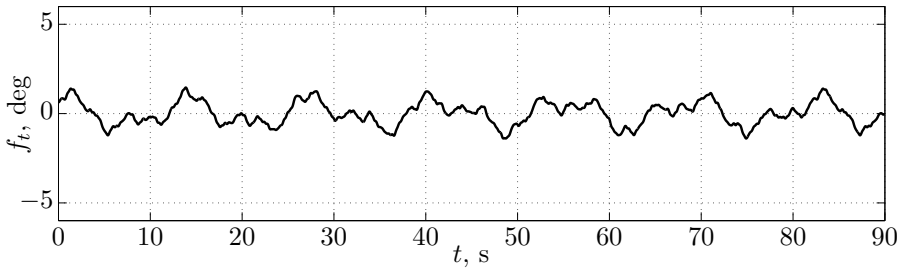


Figure 7-2: The multisine target forcing function f_t.

Varying the Neuromuscular Frequency ω_{nm} in Time

The operator response H_p was modeled as visual response H_{pe} or motion perception channel $H_{p\theta}$ in Section 4-2-3. The underlying basic system in the Equations for H_{pe} and $H_{p\theta}$ is the human Neuromuscular System (NMS), which can be modeled as a second-order mass-spring-damper system as in Equation (4-5). Because the NMS is one of the aspects of human control behavior where time-varying dynamics occur [Zaal et al., 2009], H_p will be set equal to H_{nm}, while varying ω_{nm} over time t. If this works, it is intended to make ζ_{nm} time-varying as well at a later stage. The operator response $H_p(s,t)$ in Figure 7-1 with *only* $\omega_{nm}(t)$ as a time-varying parameter, is stated in Equation (7-1).

$$H_p(s,t) = H_{nm}(s,t) = \frac{\omega_{nm}^2(t)}{s^2 + 2\zeta_{nm}\omega_{nm}(t)s + \omega_{nm}^2(t)} \tag{7-1}$$

For this first setup $\omega_{nm}(t)$ is modeled as a first-order polynomial with constants ω_{nm_0} and ω_{nm_1} as in Equation (7-2).

$$\omega_{nm}(t) = \omega_{nm_0} + \omega_{nm_1}t \tag{7-2}$$

To simulate the data, a state-space system—as defined by Equations (2-5) and (2-6)—is created with matrices $A(t)$, $B(t)$, $C(t)$ and $D(t)$ in Equations (7-3), (7-4), (7-5) and (7-6) respectively.

$$A(t) = \begin{bmatrix} -2\zeta_{nm}\omega_{nm}(t) & -({\omega_{nm}}^2(t)) \\ 1 & 0 \end{bmatrix} \tag{7-3}$$

$$B(t) = \begin{bmatrix} 1 & 0 \end{bmatrix}^\top \tag{7-4}$$

$$C(t) = \begin{bmatrix} 0 & {\omega_{nm}}^2(t) \end{bmatrix} \tag{7-5}$$

$$D(t) = \begin{bmatrix} 0 \end{bmatrix} \tag{7-6}$$

In the $A(t)$ and $C(t)$ matrices the ω_{nm} is defined by Equation (7-2), which makes the matrices in the state-space system time-varying. Note that only the $A(t)$ and $C(t)$ in fact

have time-dependent entries. The matrices $B(t)$ and $D(t)$ consist of constant entries. For the state-space system to generate the control signal vector u, the control system in Figure 7-3 is used, where $1/s$ serves as an integrator.

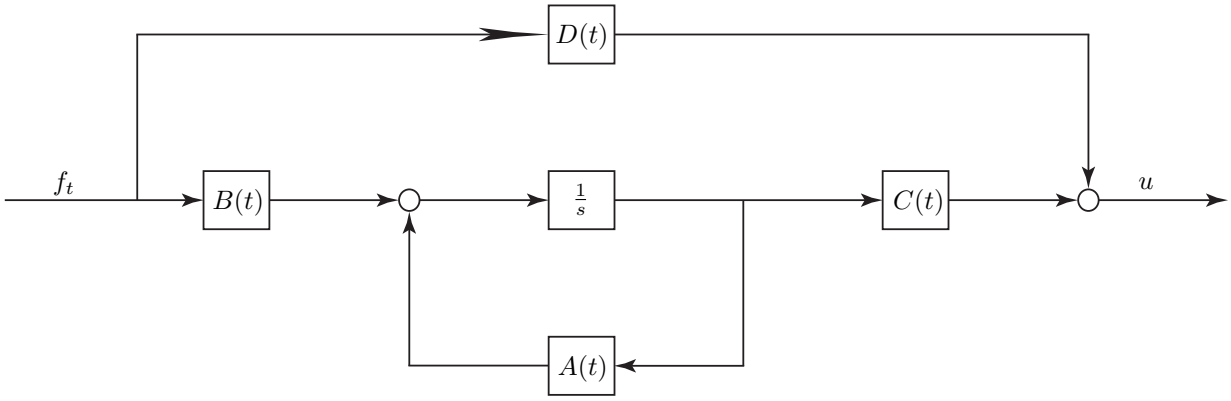


Figure 7-3: Control system used to simulate control signal u based on state-space matrices.

To check whether the data from the state-space method is generated correctly, another method is used to generate the exact same data. Based on Equation (7-7) the control system in Figure 7-4 is created.

$$X(s) = s^2 + 2\zeta_{nm}\omega_{nm}(t)s + \omega_{nm}^2(t) \tag{7-7}$$

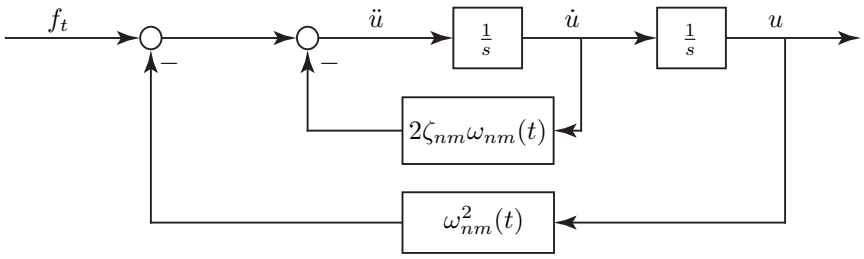


Figure 7-4: Control system used to simulate control signal u based on the system input.

Adding Noise to the Simulation

To add white noise to the simulation, as in Figure 2-2, the Random Number block in Simulink® is used. This block generates a preset sequence of normally distributed random numbers. When noise is added to the simulation, sequences with a mean μ of 0 and a variance σ^2 of 1 will be used. If it is necessary to add different sequences of noise to the simulation, a different Seed can be chosen in the Random Number block. This can, for example, be useful in the comparison of different noise additions. To scale the noise to an appropriate level, each value of the sequence generated by the Random Number block is

multiplied by a scalar so that the variance percentage of the control signal with noise u and the control signal without noise has a value between 10% to 15%. This means that u will vary between 10% to 15% from the control signal without noise, which compares to experimental results according to [Zaal & Sweet, 2011].

7-1-2 Estimating Parameters

With the simulated control signal data that was generated in the previous section, it is now possible to estimate the parameters. The parameter vector Θ at this instance is stated in Equation (7-8).

$$\Theta = \begin{bmatrix} \omega_{nm_0} & \omega_{nm_1} & \zeta_{nm} \end{bmatrix}^\top \tag{7-8}$$

It is necessary to set *a priori* initial parameter estimation values for each parameter. Furthermore, the state-space matrices from Equations (7-3), (7-4), (7-5) and (7-6) need to be known. From these matrices, the so-called sensitivity matrices can be determined. Sensitivity matrices show how the coefficients in the state-space matrices change when the value of a certain parameter also changes. The sensitivity of the entries of the state-space matrices $A(t)$, $B(t)$, $C(t)$ and $D(t)$ to changes in each parameter in Θ are calculated. This is done by taking the partial derivative with respect to each model parameter matrix entry a_{i,j^*} in the i-th row and the j^*-th column of a certain matrix as in Equation (7-9).

$$S = \left[\frac{\partial a_{i,j^*}}{\partial \Theta(p)} \right] \tag{7-9}$$

The sensitivity matrices for state-space matrices $A(t)$ and $C(t)$ from Equations (7-3) and (7-5) with ω_{nm} as in Equation (7-2) are stated in Equations (7-10) to (7-15). Note that the sensitivity matrices for $B(t)$ and $D(t)$ from Equations (7-4) and (7-6) consist of zeros in this case and are therefore not stated.

$$S_{A_{\omega_{nm_0}}}(t) = \begin{bmatrix} \frac{\partial A_{1,1}(t)}{\partial \omega_{nm_0}} & \frac{\partial A_{1,2}(t)}{\partial \omega_{nm_0}} \\ \frac{\partial A_{2,1}(t)}{\partial \omega_{nm_0}} & \frac{\partial A_{2,2}(t)}{\partial \omega_{nm_0}} \end{bmatrix} = \begin{bmatrix} -2\zeta_{nm} & -2\omega_{nm_0} - 2\omega_{nm_1} t \\ 0 & 0 \end{bmatrix} \tag{7-10}$$

$$S_{A_{\omega_{nm_1}}}(t) = \begin{bmatrix} \frac{\partial A_{1,1}(t)}{\partial \omega_{nm_1}} & \frac{\partial A_{1,2}(t)}{\partial \omega_{nm_1}} \\ \frac{\partial A_{2,1}(t)}{\partial \omega_{nm_1}} & \frac{\partial A_{2,2}(t)}{\partial \omega_{nm_1}} \end{bmatrix} = \begin{bmatrix} -2\zeta_{nm} t & -2\omega_{nm_0} t - 2\omega_{nm_1} t^2 \\ 0 & 0 \end{bmatrix} \tag{7-11}$$

$$S_{A_{\zeta_{nm}}}(t) = \begin{bmatrix} \frac{\partial A_{1,1}(t)}{\partial \zeta_{nm}} & \frac{\partial A_{1,2}(t)}{\partial \zeta_{nm}} \\ \frac{\partial A_{2,1}(t)}{\partial \zeta_{nm}} & \frac{\partial A_{2,2}(t)}{\partial \zeta_{nm}} \end{bmatrix} = \begin{bmatrix} -2\omega_{nm_0} - 2\omega_{nm_1} t & 0 \\ 0 & 0 \end{bmatrix} \tag{7-12}$$

$$S_{C_{\omega_{nm_0}}}(t) = \begin{bmatrix} \frac{\partial C_{1,1}(t)}{\partial \omega_{nm_0}} & \frac{\partial C_{1,2}(t)}{\partial \omega_{nm_0}} \end{bmatrix} = \begin{bmatrix} 0 & 2\omega_{nm_0} + 2\omega_{nm_1} t \end{bmatrix} \tag{7-13}$$

$$S_{C_{\omega_{nm_1}}}(t) = \begin{bmatrix} \frac{\partial C_{1,1}(t)}{\partial \omega_{nm_1}} & \frac{\partial C_{1,2}(t)}{\partial \omega_{nm_1}} \end{bmatrix} = \begin{bmatrix} 0 & 2\omega_{nm_0} t + 2\omega_{nm_1} t^2 \end{bmatrix} \tag{7-14}$$

$$S_{C_{\zeta_{nm}}}(t) = \begin{bmatrix} \frac{\partial C_{1,1}(t)}{\partial \zeta_{nm}} & \frac{\partial C_{1,2}(t)}{\partial \zeta_{nm}} \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix} \tag{7-15}$$

These sensitivity matrices are needed to calculate Equation (7-16), which upon integration can be used in Equation (7-17).

$$\frac{d\dot{x}(t)}{d\Theta(p)} = S_{A_{\Theta(p)}} x(t) + A(t) \frac{dx(t)}{d\Theta(p)} + S_{B_{\Theta(p)}} u(t) \tag{7-16}$$

$$\frac{dy(t)}{d\Theta(p)} = S_{C_{\Theta(p)}} x(t) + C(t) \frac{dx(t)}{d\Theta(p)} + S_{D_{\Theta(p)}} u(t) \tag{7-17}$$

The prediction error $\epsilon(k)$ as defined in Section 3-1 can be used to calculate the covariance matrix Q of the prediction error as in Equation (7-18) for m measurements and k parameters.

$$Q = \frac{1}{m} \sum_{k=1}^{m} \epsilon(k)^\top \epsilon(k) \tag{7-18}$$

With the inverse of the covariance matrix Q—also known as the precision matrix—the likelihood gradient dL and the FIM can be calculated according to Equations (7-19) and (7-20). Note that Equation (7-20) is a special form of Equation (3-8) used for signals with an unbiased Gaussian distribution.

$$dL = \left(-\frac{dy(t)}{d\Theta(p)} \right)^\top Q^{-1} \epsilon(k) \tag{7-19}$$

$$M_{\Theta\Theta} = \left(-\frac{dy(t)}{d\Theta(p)} \right)^\top Q^{-1} \left(-\frac{dy(t)}{d\Theta(p)} \right) \tag{7-20}$$

The likelihood function L can be calculated according to Equation (3-4). After a first run through the estimation procedures with the *a priori* initial parameter estimate values, the updated parameter estimate values are found through Equation (3-9). As mentioned in Section 3-3 it is possible to use an algorithm and a line-search vector to increase the accuracy of the parameters. For this first simulation the GN algorithm is used with a line-search vector α that looks at the likelihood for a starting value α_0 and an end value α_i. The line-search always starts at $\alpha_0 = 0$ and $\alpha_1 = 1$. Now, with $\alpha_i = \{1 \dots k\}$, when

$$L(\alpha_0) > L(\alpha_i) \quad \longrightarrow \quad L = L(\alpha_i)$$

and, when

$$L(\alpha_0) < L(\alpha_i) \quad \& \quad L(\alpha_0) > L\left(\frac{\alpha_0 + \alpha_i}{2}\right) \quad \longrightarrow \quad L = L\left(\frac{\alpha_0 + \alpha_i}{2}\right).$$

However, when

$$L(\alpha_0) < L(\alpha_i) \quad \& \quad L(\alpha_0) < L\left(\frac{\alpha_0 + \alpha_i}{2}\right) \quad \longrightarrow \quad \alpha_{i+1} = \frac{\alpha_0 + \alpha_i}{2}$$

the line-search algorithm starts from the beginning but now with $\alpha_i = \alpha_{i+1}$.

7-1-3 Initial Simulation Results

The creation of simulation data according to Section 7-1-1 is now done using the values in Table 7-2. These values are based on [Nieuwenhuizen et al., 2008], where the neuromuscular frequency ω_{nm} has a value of 12.0 rad/s and the neuromuscular damping ζ_{nm} is set to 0.3. McRuer et al. [1965] indicated values of 16.5 rad/s for ω_{nm} and 0.12 for ζ_{nm}, so the values used in [Nieuwenhuizen et al., 2008] are reasonable. With the neuromuscular frequency constants in Equation (7-2), the value of 12.0 rad/s is approximated by using ω_{nm_0} as a starting value and a slope with the value of ω_{nm_1}, so that after $t = T_m = 81.92$ seconds the value of 12.0 rad/s is achieved. Note that this does not necessarily approach a realistic time-variation of ω_{nm} in an experimental situation. A more realistic time-variation can be modeled at a later stage when the polynomial increases to a higher-order.

Table 7-2: Model Parameter Values.

Parameter		Value
ω_{nm_0}	neuromuscular frequency constant 0	0.0100 rad/s
ω_{nm_1}	neuromuscular frequency constant 1	0.1464 rad/s^2
ζ_{nm}	neuromuscular damping	0.3000 -

In Figures 7-3 and 7-4 two different methods were shown to generate the same simulation data. The results of generating the data for control signal u from both methods with the values of Table 7-2 are plotted in Figure 7-5. From the plot it becomes clear that the signals overlap, and are thus equal to each other proving the data generated for the simulation is correct. The Simulink® model used to generate this data can be found in Appendix B. In the first 20 seconds of the plot of Figure 7-5 the transient—which results from simulating the model with a zero initial state—has a substantial influence on the model response.

Because this transient behavior will influence the parameter estimation, the signal was generated for twice T_m and the first part (one time T_m) (from 0 seconds to 81.92 seconds in Figure 7-5) will not be used for the parameter estimation. This kind of omittance is also performed for parameter estimation with real experimental data to ensure that the transient behavior—resulting from the initial stabilization that is typically found in human operators—has no contribution to the final parameter estimation on the one hand, and to omit having to estimate the initial state of the model on the other hand. If the plot for the neuromuscular frequency ω_{nm} is taken in account, as in Figure 7-6, a problem arises: for that part of the data used for the MLE, the part with the white background, starts at ω_{nm} = 12 rad/s and ends at ω_{nm} = 24 rad/s. To solve this problem and to let the polynomial end at ω_{nm} = 12 rad/s instead, the polynomial is changed. As it is not necessary to vary ω_{nm} from 0 seconds to 81.92 seconds, ω_{nm} is held constant at 5 rad/s as can be seen in Figure 7-7. The appropriate values for the polynomial constants can be found in Table 7-3. Because the parameter estimation starts after $t = T_m$ = 81.92 seconds, ω_{nm_0} is not equal to 5 rad/s, but is set to -2 rad/s.

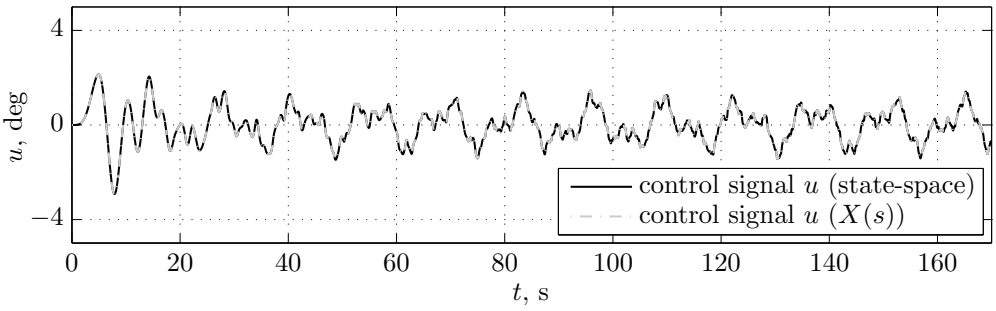


Figure 7-5: Control signal data generated with state-space matrices and system input respectively. Transient behavior is clearly visible in the first 20 seconds of these plots.

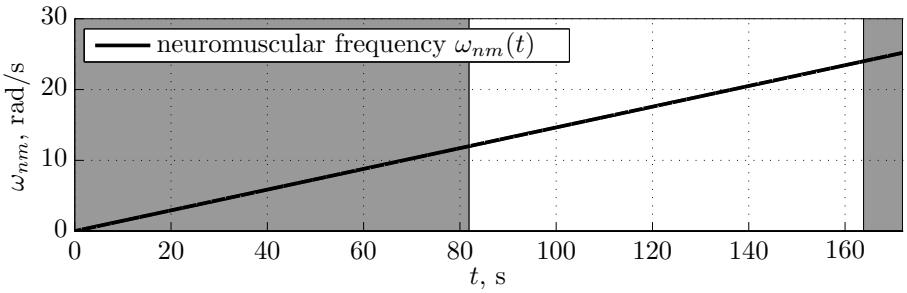


Figure 7-6: Neuromuscular frequency ω_{nm} according to Equation (7-2) with values from Table 7-2.

MLE can now be performed for the neuromuscular frequency ω_{nm} as in Figure 7-7 and neuromuscular damping ζ_{nm} with values according to Table 7-3. In Table 7-4 various simulation results are displayed. Three sets of initial conditions are used to estimate

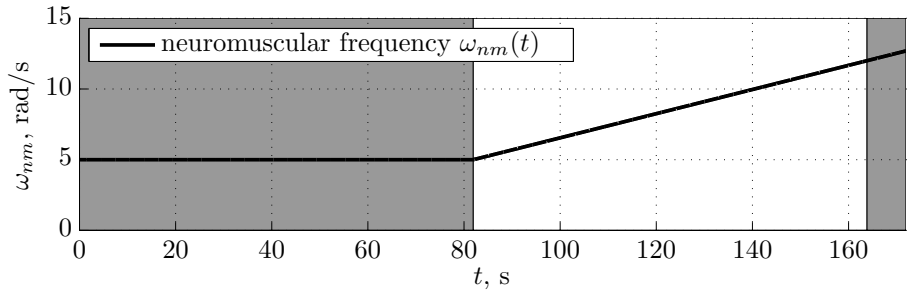
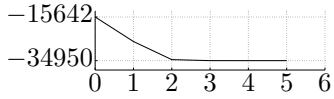
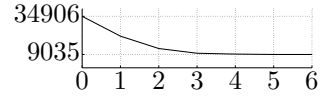
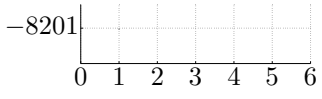


Figure 7-7: Neuromuscular frequency ω_{nm} with values from Table 7-3 and held constant for the first 81.92 seconds.

Table 7-3: Model Parameter Values.

Parameter		Value
ω_{nm_0}	neuromuscular frequency constant 0	-2.0000 rad/s
ω_{nm_1}	neuromuscular frequency constant 1	0.0854 rad/s^2
ζ_{nm}	neuromuscular damping	0.3000 -

Table 7-4: Parameter Estimation Values: the plots in the last column show the number of iterations of the GN algorithm versus the value of the negative log-likelihood at each iteration.

Parameter	True Value	Initial Value	Estimated Value	Negative Log-Likelihood
ω_{nm_0}	-2.0000 rad/s	-2.0000 rad/s	-1.7432 rad/s	
ω_{nm_1}	0.0854 rad/s^2	0.0800 rad/s^2	0.0817 rad/s^2	
ζ_{nm}	0.3000 -	0.3000 -	0.3133 -	
ω_{nm_0}	-2.0000 rad/s	-1.0000 rad/s	-0.8609 rad/s	
ω_{nm_1}	0.0854 rad/s^2	0.0200 rad/s^2	0.0183 rad/s^2	
ζ_{nm}	0.3000 -	0.2000 -	0.2496 -	
ω_{nm_0}	-2.0000 rad/s	0.2000 rad/s	0.2000 rad/s	
ω_{nm_1}	0.0854 rad/s^2	0.2000 rad/s^2	0.2000 rad/s^2	
ζ_{nm}	0.3000 -	0.2000 -	0.2000 -	

the true (simulated) values. The parametric MLE is performed in such a way that the estimation stops when the difference between the negative log-likelihood values of an iteration is smaller than 1.00. From Table 7-4 it becomes apparent that somewhat accurate results are only obtained when initial values that are very close to the true values are chosen. The results of the second set of values in Table 7-4 are of the same order of magnitude as the true values, but the estimations are not very accurate. Moreover, the likelihood for the second set of values is much higher compared to the likelihood of the first set. For the third initial condition in Table 7-4 the GN algorithm already stops after one iteration, thus returning the initial values as a result. The estimated values in the first and second set are

in the neighborhood of the true value. MLE on simulated data with the addition of noise resulted in similar results: there is almost no difference between estimation with or without noise. These results are expected as the used model in Figure 2-2 is a strongly simplified model with only one set of dynamics—the H_{nm} dynamics—and pure white noise. The simulated data in this case is limited to a time frame, but for a longer time frame more accurate results are expected: in theory the estimated value should be reached exactly when the simulated data is generated for an infinite time.

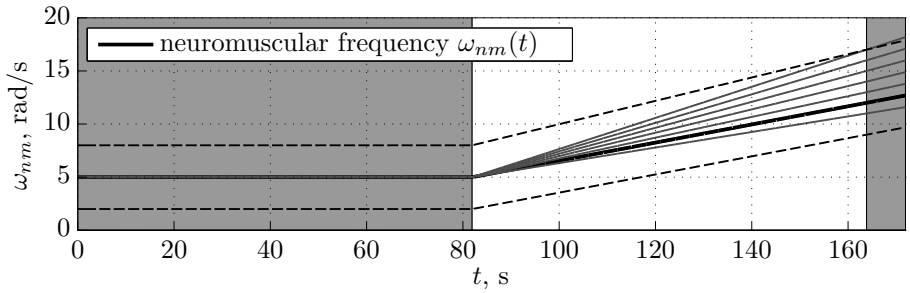


Figure 7-8: Different neuromuscular frequencies ω_{nm} with varying ω_{nm0} and ω_{nm1}. The frequencies are held constant for the first 81.92 seconds and correspond to the respective control signal responses in 7-9 and 7-10.

To check whether the parameters ω_{nm} and ζ_{nm} contain enough variation for parameter estimation to have effect, the values of the parameters are changed to see how much difference this yields in terms of the control signal u. For the neuromuscular frequency ω_{nm} different values are plotted in Figure 7-8. The gray lines are neuromuscular frequencies held constant at 5 rad/s, with different slopes—only ω_{nm1} is varied in this case—starting from 81.92 seconds. This results in neuromuscular frequencies with different slopes ranging from $\omega_{nm} = 11$ rad/s to $\omega_{nm} = 17$ rad/s. In Figure 7-9 the simulated model output u is plotted with the true model parameters from Table 7-3 (black line). The corresponding control signal output u under influence of these variations in ω_{nm} is also plotted in Figure 7-9 (gray lines). The dashed, black lines in Figure 7-8 are held constant for the first 81.92 seconds at values of 2 rad/s and 8 rad/s respectively, after which they vary. The plot in Figure 7-10 shows the control output u according to Table 7-3 (black line) together with the control outputs of these variations (dashed black lines). The influence of ω_{nm1} seems very small, as there is almost no visible variation in the control outputs in Figure 7-9. The plot in Figure 7-10 shows some variation in the first 30 seconds, after which almost no significant variation in u is visible. This means that varying ω_{nm0} only has a minimal effect, which mostly affects the transient part. The control output u has almost no variation due to ω_{nm} and this result poses a serious problem for qualitative parameter estimation. In Figure 7-11, the value for ζ_{nm} from Table 7-3 is plotted (black line) against values ranging from $\zeta_{nm} = 0.1$ to $\zeta_{nm} = 1.0$. The difference with the effect of varying ω_{nm} on the model output is clearly visible. It can be concluded that ω_{nm} does not contain enough variation for parameter estimation to have effect. Although varying ζ_{nm} as a polynomial (instead of ω_{nm}) seems relatively promising, a different approach is taken which will be explained in

Section 7-2. Note that as MLE is performed with the data from $t = 81.92$ seconds to $t = 163.84$ seconds, only the data for this period of time was used for creating Figures 7-9 and 7-11. The plots have been relabeled to range from $t = 0$ seconds to $t = 90$ seconds.

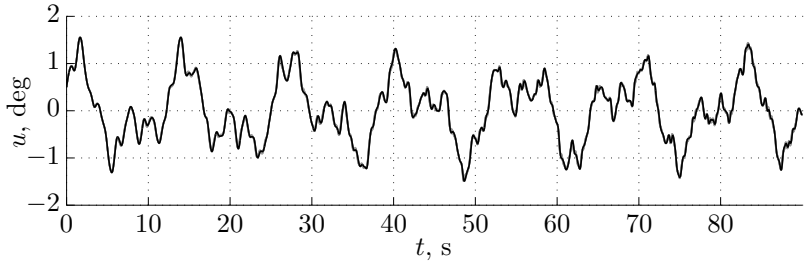


Figure 7-9: The control signal u as defined by Table 7-3 (black line) plotted with the control signal u for varied neuromuscular frequency constants corresponding to the gray lines in Figure 7-8.

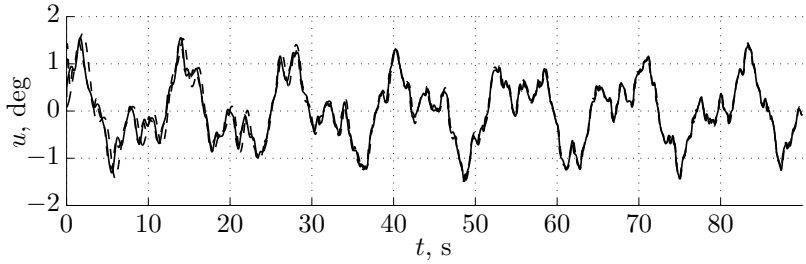


Figure 7-10: The control signal u as defined by Table 7-3 (black line) plotted with the control signal u for varied neuromuscular frequency constants corresponding to the dashed, black lines in Figure 7-8.

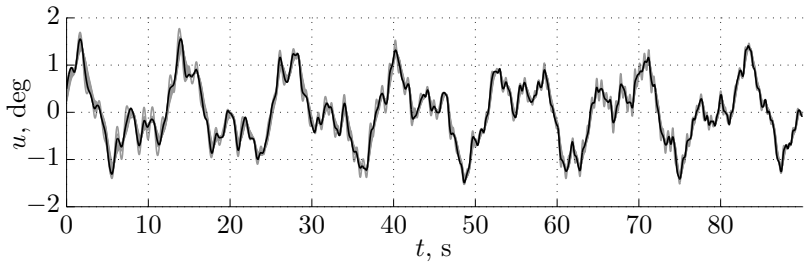


Figure 7-11: The control signal u as defined by Table 7-3 (black line) plotted with the control signal u for varied neuromuscular damping ζ_{nm} with constant values between 0.1 and 8.0 (gray lines).

7-2 Redefining the Simulation Model: A Time-Varying Gain

Because of the lack of variation in the simulated model output due to variations in the neuromuscular frequency ω_{nm}, as described in Section 7-1-3, a different approach will be taken in this Section to redefine the proposed model. The neuromuscular frequency ω_{nm} and neuromuscular damping ζ_{nm} will be kept constant and a time-varying gain $K(t)$ will be added to Equation (7-1) to obtain Equation (7-21). This is done to ensure that the model output u is more strongly affected by the applied time-varying model dynamics.

$$H_p(s,t) = K(t)H_{nm}(s) = K(t)\frac{\omega_{nm}^2}{s^2 + 2\zeta_{nm}\omega_{nm}s + \omega_{nm}^2} \tag{7-21}$$

Note that, excluding a delay term, Equation (7-21) resembles a human operator model appropriate for control of a system with NMS dynamics [McRuer et al., 1965].

7-2-1 Estimating Parameters with the Gain as a Polynomial

For the time-varying gain $K(t)$, polynomials with different orders of magnitude were created to understand how the MLE will be influenced by varying the order of a polynomial and the number of parameters. This is important because, ultimately, the modeled signal should be representative for time-varying human manual control behavior. In this thesis, the transition between two constant sets of control behavior is studied: the human controller changes its behavior from single-integrator dynamics to double-integrator dynamics. The approximation of such a transition between two sets of dynamics becomes more accurate with higher-order polynomials. The polynomials modeled starting at $K(81.92\text{ s}) = 0.8$ and ending at $K(163.84\text{ s}) \approx 0.3$ are displayed in Figure 7-12, with their accompanying constant values in Table 7-5. With these values the gain $K(t)$ is modeled after the pilot visual gain $K_v(t)$ in [Zaal & Sweet, 2011]. The polynomial constants are defined according to Equation (7-22). The parameter vector Θ for a fourth-order polynomial is stated in Equation (7-23). Note that the fourth-order polynomial in Figure 7-12b is the highest-order polynomial in this set, thus providing the best representation of an experimental ramp-like signal. The sensitivity matrices for the model, given by Equation (7-21), are calculated for all parameters in Equation (7-22).

$$K(t) = K_0 + K_1 t + K_2 t^2 + K_3 t^3 + K_4 t^4 \tag{7-22}$$

$$\Theta = \begin{bmatrix} K_0 & K_1 & K_2 & K_3 & K_4 & \omega_{nm} & \zeta_{nm} \end{bmatrix}^\top \tag{7-23}$$

Using the first-, second-, third- and fourth-order polynomials for $K(t)$ with Equation

Table 7-5: True values for $K(t)$ polynomials according to Equation (7-22) and plotted in Figure 7-12.

Parameter	First-Order Values	Second-Order Values	Third-Order Values	Fourth-Order Values
K_0	1.2997	0.7693	0.7576	0.8508
K_1	-0.0061	0.0039	0.0048	-0.0128
K_2	0.0000	-4.3035 x 10^{-5}	-5.6568 x 10^{-5}	4.0498 x 10^{-4}
K_3	0.0000	0.0000	5.2427 x 10^{-8}	-4.1238 x 10^{-6}
K_4	0.0000	0.0000	0.0000	1.2146 x 10^{-8}

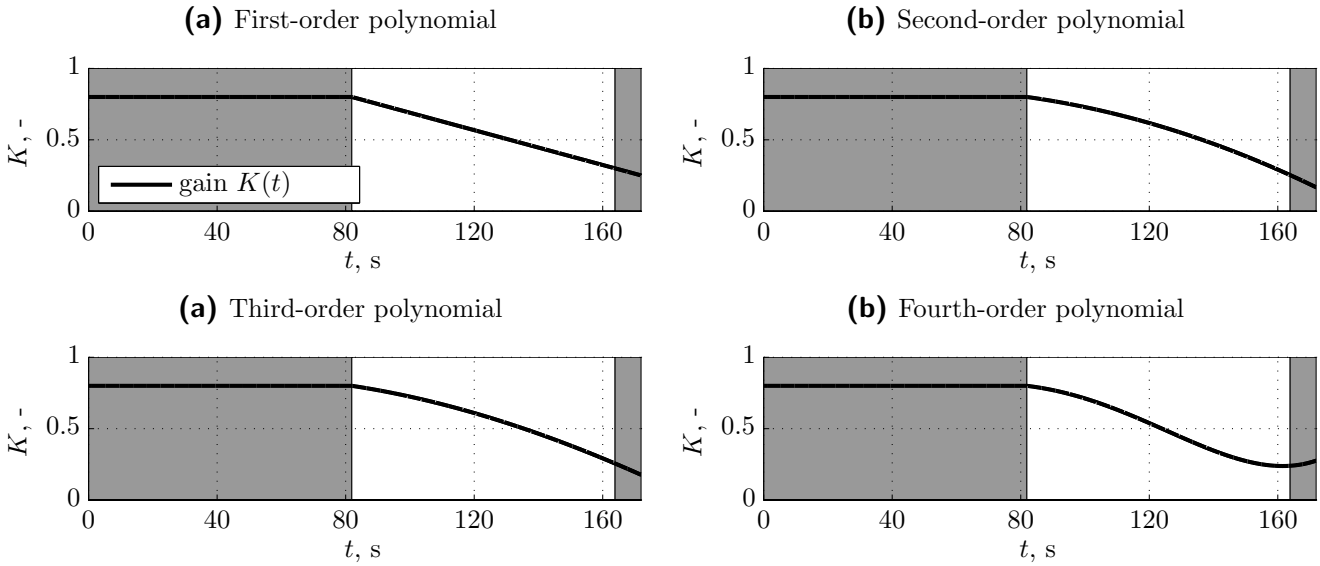
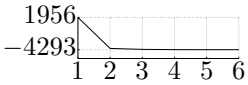
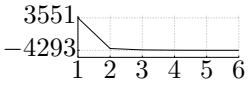
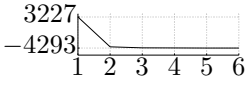
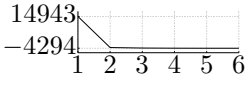


Figure 7-12: Gain signals $K(t)$ used in Equation (7-21) and modeled as polynomials with different orders of magnitude. Only the part in white is used for MLE to avoid transient behavior.

(7-22) in a model according to Figure 2-2 data is generated with a noise n signal according to Section 7-1-1. The parameters are then estimated from model simulation data using the proposed MLE procedure. To check whether or not the prediction errors $\epsilon(k)$ have a zero-mean Gaussian distribution, the PDFs are checked using Seeds ranging from 0 to 19. As the PDFs all show similar results with the expected zero-mean Gaussian distribution, one plot with Seed 19 is available in Figure 7-13. The parameter estimation results of one of the MLE runs are shown in Table 7-6. For this run, the initial values are set to a value of 1 in the same order of magnitude of the true value. Although the negative log-likelihood already approaches the minimum after one iteration for each polynomial-set and stops after six iterations for each set, the results in Table 7-6 show that the estimated values of the polynomial coefficients deteriorate when a higher-order polynomial is used in the model. This can be seen by comparing the true values with the estimated values in the table. The estimated values of the time-invariant parameters, however, approach the true values for each polynomial-set in Table 7-6 (with an exception for the value of ζ_{nm} in the fourth-order

polynomial set).

Table 7-6: Parameter Estimation Values: the plots in the last column show the number of iterations of the GN algorithm versus the value of the negative log-likelihood at each iteration. The noise n for this data set is generated with $\sigma_n^2 = 0.15$ deg^2 and Seed 19.

Parameter	True Value	Initial Value	Estimated Value	Negative Log-Likelihood
K_0	1.2997 x 10^0 -	1.0000 x 10^0 -	1.2904 x 10^0 -	
K_1	-0.0061 x 10^0 -	-0.0010 x 10^0 -	-0.0061 x 10^0 -	
ω_{nm}	12.0000 x 10^0 rad/s	10.0000 x 10^0 rad/s	11.6190 x 10^0 rad/s	
ζ_{nm}	0.3000 x 10^0 -	0.1000 x 10^0 -	0.3233 x 10^0 -	
K_0	0.7693 x 10^0 -	0.1000 x 10^0 -	0.8189 x 10^0 -	
K_1	0.0039 x 10^0 -	0.0010 x 10^0 -	0.0029 x 10^0 -	
K_2	-4.3035 x 10^{-5} -	-1.0000 x 10^{-5} -	-3.9137 x 10^{-5} -	
ω_{nm}	12.0000 x 10^0 rad/s	10.0000 x 10^0 rad/s	11.5675 x 10^0 rad/s	
ζ_{nm}	0.3000 x 10^0 -	0.1000 x 10^0 -	0.3232 x 10^0 -	
K_0	0.7576 x 10^0 -	0.1000 x 10^0 -	-0.8506 x 10^0 -	
K_1	0.0048 x 10^0-	0.0010 x 10^0 -	0.0028 x 10^0 -	
K_2	-5.6568 x 10^{-5} -	-1.0000 x 10^{-5} -	-4.3739 x 10^{-5} -	
K_3	5.2427 x 10^{-8} -	1.0000 x 10^{-8}-	2.8762 x 10^{-5} -	
ω_{nm}	12.0000 x 10^0 rad/s	10.0000 x 10^0 rad/s	11.6239 x 10^0 rad/s	
ζ_{nm}	0.3000 x 10^0 -	0.1000 x 10^0 -	0.3227 x 10^0 -	
K_0	0.8508 x 10^0 -	0.1000 x 10^0 -	2.6590 x 10^0 -	
K_1	-0.0128 x 10^0 -	-0.0100 x 10^0 -	-0.0727 x 10^0 -	
K_2	4.0498 x 10^{-4} -	1.0000 x 10^{-4} -	0.0011 x 10^0 -	
K_3	-4.1238 x 10^{-6} -	-1.0000 x 10^{-6} -	-8.0117 x 10^{-6} -	
K_4	1.2146 x 10^{-8} -	1.0000 x 10^{-8} -	1.9815 x 10^{-8} -	
ω_{nm}	12.0000 x 10^0 rad/s	10.0000 x 10^0 rad/s	11.6248 x 10^0 rad/s	
ζ_{nm}	0.3000 x 10^0 -	0.1000 x 10^0 -	0.2496 x 10^0 -	

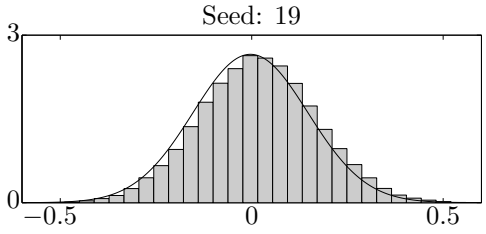


Figure 7-13: Probability Density Function of the prediction errors $\epsilon(k)$ for a model using a fourth-order polynomial gain $K(t)$ according to Table 7-5. The noise n for this data set is generated with $\sigma_n^2 = 0.15$ deg^2 and Seed 19.

Performing different MLE runs of the third- and fourth-order polynomial data sets showed that many matrices used for the calculations suffered from singularity problems. This can

be traced back to the decreasing order of magnitude in the gain constants for higher-order polynomials—i.e. the gain constants become very small—and the mathematical accuracy of MATLAB®. When very small constants are considered and used for calculations in the sensitivity matrices, the resulting values are even smaller. MATLAB® only has a finite amount of accuracy and runs into problems with very small values. Although the polynomials in Figure 7-12 are modeled to start at a value of 0.8 and descend to a value of approximately 0.3, a range of different signals with different start and ending values was also tested to see what effect this would have on parameter estimation. This resulted in the notion that higher-order polynomials with coefficients in the order of magnitude of 10^{-10} and lower give rise to an ill-conditioned FIM. Obtaining good estimates with MLE in these cases is not possible, which forms a new problem for the simulation, as multiple time-varying parameters are needed for extended simulation models. Another problem worth noting is that with higher-order polynomials it becomes increasingly difficult to set the *a priori* initial values for the parameters. Ideally, any *a priori* value should be able to give an accurate estimate. In practice, the *a priori* information not only influences the number of iterations needed to minimize the negative log-likelihood, but can even make it impossible to perform a MLE in cases where the *a priori* values for certain parameters are chosen too far away from the true values.

7-2-2 Estimating Parameters with the Gain as a Boltzmann Sigmoid

Although higher-order polynomials are suitable to accurately model the type of time-variation in model parameters that is expected for experimental measurements, the problems related to the small constants of these polynomials led to the decision to look at another suitable mathematical function: the Boltzmann sigmoid, a function often used in biology and environmental analysis [Motulsky & Christopoulos, 2004]. During the selection of the Boltzmann sigmoid, the requirement that the function should be differentiable in order for parameter estimation to take place was taken into account. A plot of the Boltzmann sigmoid can be found in Figure 7-14 and this sigmoid is defined in Equation (7-24) as implemented for modeling the time variation of the gain $K(t)$ of the same model considered in Section 7-2-1. The corresponding parameter vector Θ is stated in Equation (7-25).

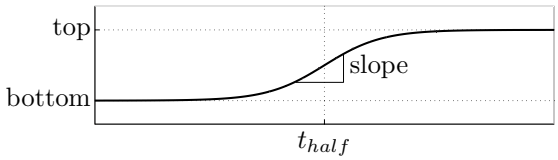


Figure 7-14: The Boltzmann sigmoid.

$$K(t) = bottom + \frac{top - bottom}{1 + \exp\left(\frac{t_{half} - t}{slope}\right)} = K_1 + \frac{K_0 - K_1}{1 + \exp\left(\frac{K_2 - t}{K_3}\right)} \tag{7-24}$$

$$\Theta = \begin{bmatrix} K_0 & K_1 & K_2 & K_3 & \omega_{nm} & \zeta_{nm} \end{bmatrix}^\top \tag{7-25}$$

An advantage of the Boltzmann sigmoid in comparison to higher-order polynomials is that it always has a maximum of four parameters that need to be estimated: the top, the bottom, the time at which the sigmoid is halfway, and the slope of the sigmoid. The possibility to merge two sigmoidal signals—a so-called double sigmoid—might also be useful for modeling time-varying parameters. The restriction to a sigmoidal shape can, however, also bring along disadvantages when human control behavior with sudden changes is considered. In Figure 7-15 the Boltzmann sigmoid is used to generate a gain signal similar to that considered in Section 7-2-1. The corresponding values of the Boltzmann sigmoid are stated in Table 7-7. A rather steep slope is chosen in comparison to the polynomial gain signal in Figure 7-12b because the sigmoid would otherwise not exactly reach the top and bottom values at $t = 81.92$ seconds and $t = 163.84$ seconds, which is important in this case because these parameters need to be estimated.

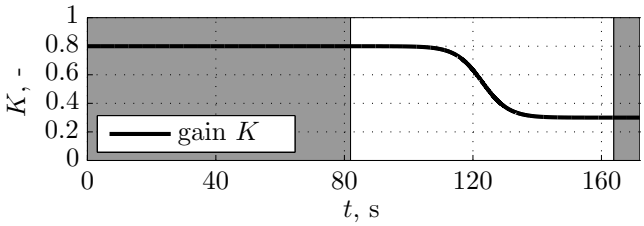


Figure 7-15: Gain signal $K(t)$ modeled as a Boltzmann sigmoid with Equation (7-24). Only the part in white is used for MLE to avoid transient behavior.

Table 7-7: Boltzmann Sigmoid Model Gain Parameter Values.

Parameter		Value
K_0	sigmoid top gain constant	0.80 -
K_1	sigmoid bottom gain constant	0.30 -
K_2	sigmoid t_{half} gain constant	122.88 s
K_3	sigmoid slope gain constant	-4.12 -
ω_{nm}	neuromuscular frequency	12.00 rad/s^2
ζ_{nm}	neuromuscular damping	0.30 -

Using the Boltzmann sigmoid for $K(t)$ from Figure 7-15 in a model according to Figure 2-2, data is generated with a noise signal n in accordance with Section 7-1-1. The MLE that was performed on this data failed: because the first iteration only returned NaN values for all parameter estimates. The plots in Figure 7-16 were created to check if the parameters used for data generation contain enough variation. The plots show that each parameter contains enough variation. The fading of K_0 and strengthening of K_1 with a switch point around 40.96 seconds can be attributed to the shape of the sigmoid, as K_0 describes the top and K_1 describes the bottom. For K_2 and K_3 the variation is small and only apparent

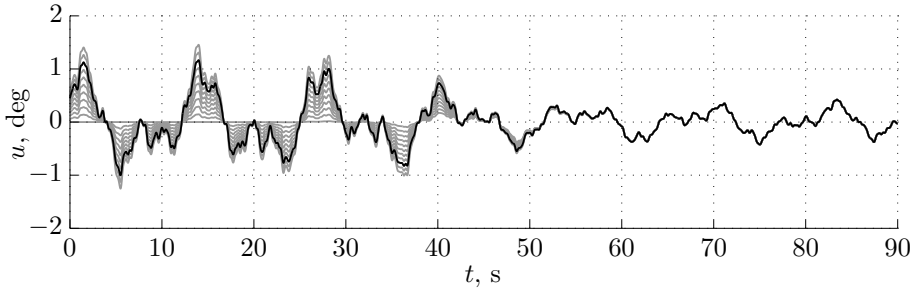
around 40.96 seconds where the t_{half} and slope are important for the sigmoidal shape. The variations in the parameters for the sigmoidal $K(t)$ are, however, adequate, so the problem concerning the MLE failure must be found elsewhere.

The problem was found to be related to the sensitivity matrices and the mathematical $top - bottom$ parameter relationship in Equation (7-24) that was used for describing the time-varying gain parameter. To avoid this problem, Equation (7-24) is replaced with Equation (7-26).

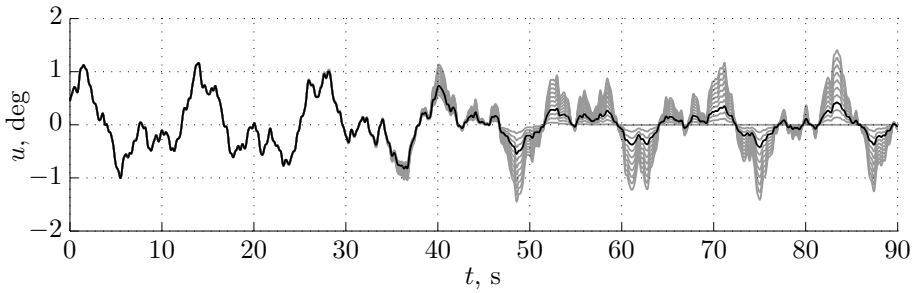
$$K(t) = bottom + \frac{top - bottom}{1 + \exp\left(\frac{t_{half} - t}{slope}\right)} = K_1 + \frac{K_0^*}{1 + \exp\left(\frac{K_2 - t}{K_3}\right)} \tag{7-26}$$

Note that Equation (7-26) only replaces Equation (7-24) in the MLE to generate a different set of sensitivity matrices. This implies that a different model parametrization. Running the simulation to generate data is still done with Equation (7-24). The MLE performed with Equation (7-26) produced satisfactory results. To check whether or not the prediction errors $\epsilon(k)$ have a zero-mean Gaussian distribution, the PDF is checked using Seeds ranging from 0 to 19. The PDFs all showed similar results with the expected zero-mean Gaussian distribution. The parameter estimation results of one of the MLE runs (Seed 19) are shown in Figures 7-17 and 7-18. Take into account that, for this example, K_0^* has a true value of 0.5, with $K_0 = 0.8$ and $K_1 = 0.3$ from Table 7-7. For this run the initial values were set to a value of 1 in the same order of magnitude as the true value. The negative log-likelihood is plotted per iteration in Figure 7-17 and clearly shows that it decreases towards a minimum value. Figure 7-18 shows the convergence of all parameters to their respective true values. In Figure **??** the convergence of the prediction error $\epsilon(k)$ to the noise signal n is clearly visible. The values for covariance Q, accompanying each plot per iteration, show how the covariance decreases from $Q \approx 0.031$ deg^2 to a value of $Q \approx 0.025$ deg^2, which—as expected—is close to the value of the variance $\sigma_n^2 = 0.025$ deg^2 used to generate the noise for the simulated data. With the analysis done in this Section, it can be concluded that the gain $K(t)$ can be correctly modeled as a Boltzmann sigmoid with the proposed model parametrization. The Boltzmann sigmoid will thus be used for this purpose.

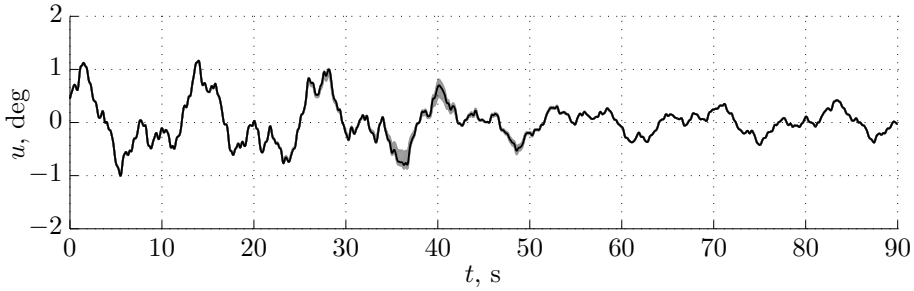
(a) K_0 between 0.0 and 1.0.



(b) K_1 between 0.0 and 1.0.



(c) K_2 between 115 s and 122 s.



(d) K_3 between 1 and 10.

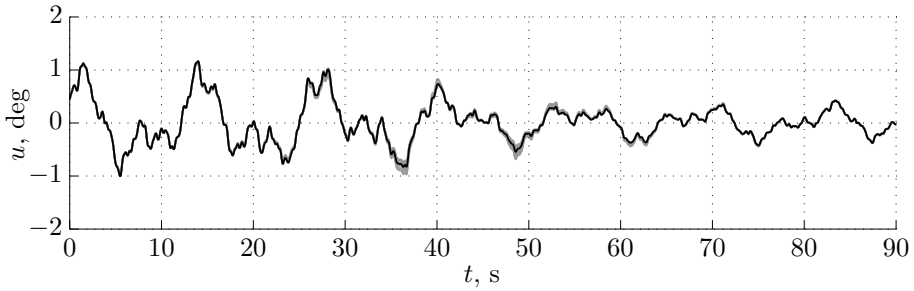


Figure 7-16: The control signal u as defined by Table 7-7 (black line) plotted with the control signal u for varied gain constants (gray lines). Noise was generated with $\sigma_n^2 = 0.025$ deg^2 and Seed 19.

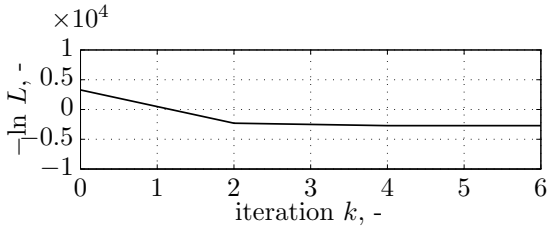


Figure 7-17: Negative log-likelihood for a simulation with the Boltzmann sigmoid.

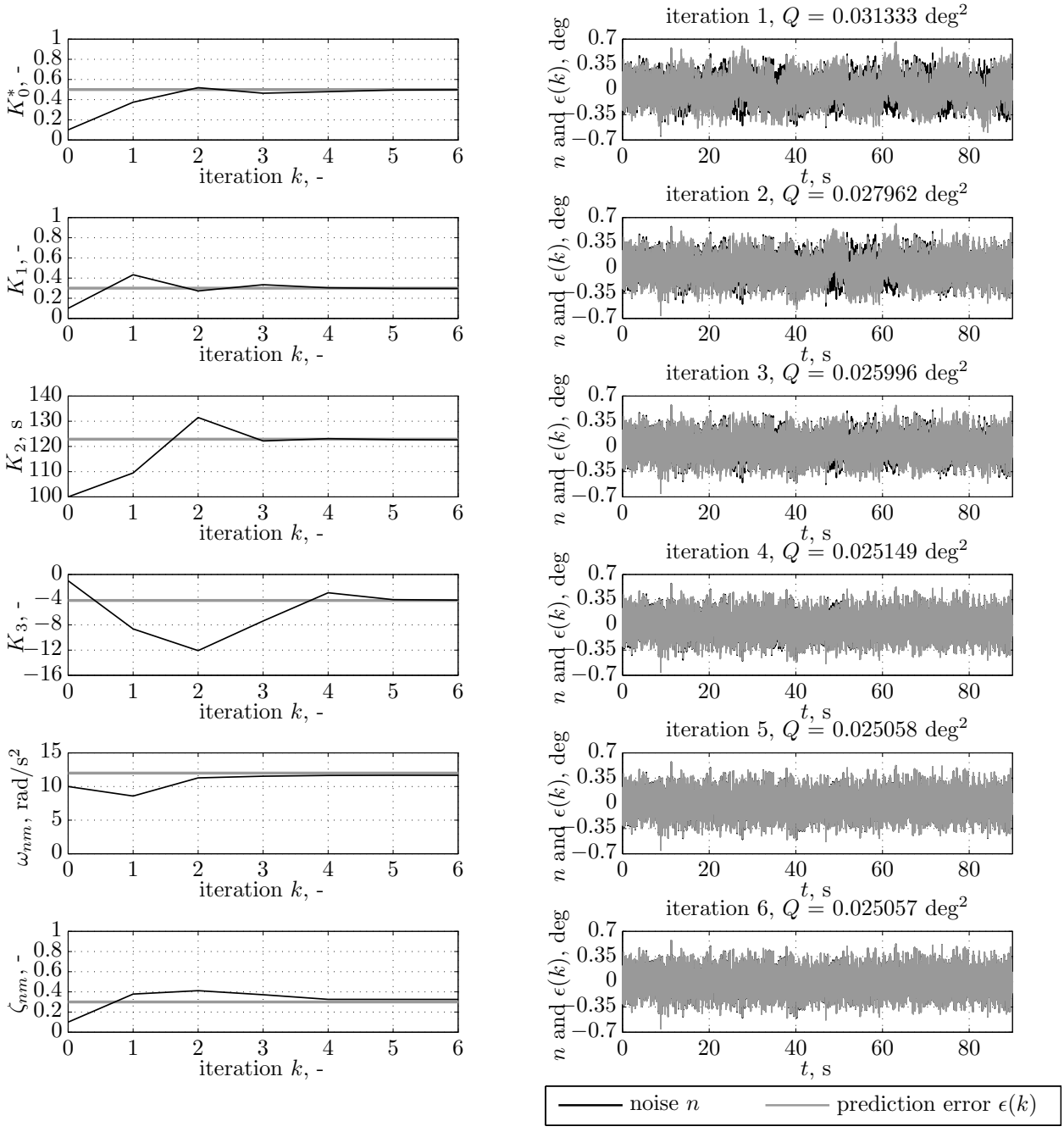


Figure 7-18: Estimated parameter values compared to their true values for a simulation with the Boltzmann sigmoid.

7-3 Expanding the Simulation Model

The Boltzmann sigmoid used to simulate a time-varying gain $K(t)$ in Section 7-2-2 demonstrated that it is possible to estimate the parameters of such a sigmoid rather accurately. Therefore, based upon [Zaal & Sweet, 2011], the choice was made to use the Boltzmann sigmoid to expand the model with another time-varying parameter: the lead time constant T_{lead}. This will be discussed in Section 7-3-1. In Section 7-3-2 the human operator part of the simulation model is completed by adding a time delay. With the addition of the lead time constant T_{lead} and the time delay, the transfer function for the human operator H_p is equal to Equation (4-9). This resembles a human operator model appropriate for NMS dynamics.

7-3-1 Adding a Boltzmann Sigmoid for the Lead Time Constant

In this section the lead time constant T_{lead} will be added to the human operator model. The addition of T_{lead} will change Equation (7-1) into Equation (7-27). Gain K is written as K_v in this equation, because it represents the human operator visual gain. The time-varying lead time constant T_{lead} will be modeled according to Equation (7-24) with constants T_{lead_0}, T_{lead_1}, T_{lead_2} and T_{lead_3} respectively. This results in parameter vector Θ in Equation (7-28).

$$H_p(s,t) = K_v(t)\left(T_{lead}(t)s + 1\right)\frac{\omega_{nm}^2}{s^2 + 2\zeta_{nm}\omega_{nm}s + \omega_{nm}^2} \tag{7-27}$$

$$\Theta = \begin{bmatrix} K_{v_0} & K_{v_1} & K_{v_2} & K_{v_3} & T_{lead_0} & T_{lead_1} & T_{lead_2} & T_{lead_3} & \omega_{nm} & \zeta_{nm} \end{bmatrix}^\top \tag{7-28}$$

The lead time constant T_{lead} is modeled to reflect the variation in this parameter considered in [Zaal & Sweet, 2011]. Both sigmoids for the visual gain K_v and the lead time constant T_{lead} are plotted in Figure 7-19. The corresponding Boltzmann sigmoid values of the lead time constant are given in Table 7-8.

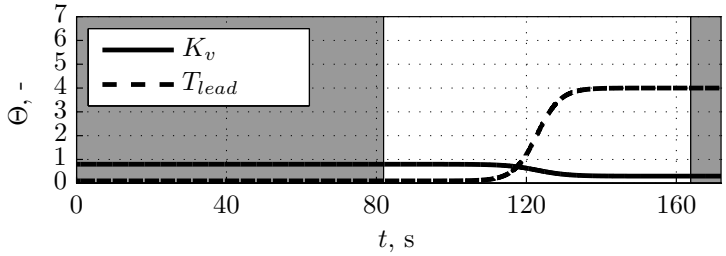


Figure 7-19: Lead time constant signal $T_{lead}(t)$ modeled as a Boltzmann sigmoid in accordance with Equation (7-24). Only the part in white is used for MLE to avoid transient behavior.

Table 7-8: Boltzmann Sigmoid Model Lead Time Parameter Values.

Parameter		Value
T_{lead_0}	sigmoid top time constant	4.00 -
T_{lead_1}	sigmoid bottom time constant	0.10 -
T_{lead_2}	sigmoid t_{half} time constant	122.88 s
T_{lead_3}	sigmoid slope time constant	3.12 -

The simulation data with both K_v and T_{lead} was created with Seed 19 and variance σ_n^2 of 0.07 deg^2 to maintain a remnant variance percentage $n_p = (\sigma_n^2/\sigma_u^2) \cdot 100\%$ of 10%. Parameter estimation is now performed with initial values of 1 in the same order of magnitude as the true value of each parameter in Θ. The results of this estimation are shown in Figures 7-20, 7-21 and 7-22. Although the likelihood in Figure 7-21 steadily decreases towards a minimum, and the prediction error $\epsilon(k)$ in Figure 7-22 becomes more equal–but not fully equal–to the noise n, the estimated values of the parameters of gain sigmoid K_v diverge from the true values at iteration 4. The resulting estimated parameters for this sigmoid therefore become useless. At the same time the parameters of the lead time constant T_{lead} stay close to their true values: T_{lead_1} converges to its true value at the fourth iteration, while T_{lead_2} and T_{lead_3} show a small bias in comparison to the parameters of K_v. In Figure 7-22 a difference is also clearly visible. In the plot of iteration 3 there are some distinct peaks in the prediction error between 35 and 50 seconds, these peaks disappear at iteration 4, but the relatively accurate prediction error from 50 to 81.92 seconds at iteration 3 is less accurate at iteration 4. Note that the covariance Q lowers, but does not get near to a σ_n^2 of 0.07 deg^2. The problem with the parameter estimation was traced back to the initial parameter value of the t_{half} parameter of each sigmoid, in this case K_{v_2} and T_{lead_2}. When one of these values drops below 110 seconds, the parameter estimation does not converge to the true parameter values. The same happens when an initial value of 135 seconds or higher is chosen. This suggests that there is leeway for approximately 12 seconds below and above the true value in which the chosen initial parameter value of K_{v_2} and T_{lead_2} results in appropriate parameter estimation. The upper and lower bound of this parameter are thus important for parameter estimation. To show that the parameter estimation works, Figures 7-23, 7-24 and 7-25 display the results of the parameter estimation done with initial values of 1 in the same order of magnitude as the true value of each parameter in Θ, except for K_{v_2} and T_{lead_2}, which were both set to a value of 134 seconds.

7-3-2 Adding a Time Delay

The last step in completing the human operator model, as proposed by Zaal & Sweet [2011], is to add a time delay. Compare Equation (7-27) used in the previous section with Equation (4-9). The only difference between these equations is the time delay $e^{-s\tau_v}$. In [Zaal & Sweet, 2011] a value of 0.20 seconds is given to the visual-perception time delay τ_v. This value will be used here as well. As mentioned in [Jones, 2004] *the time-delay*

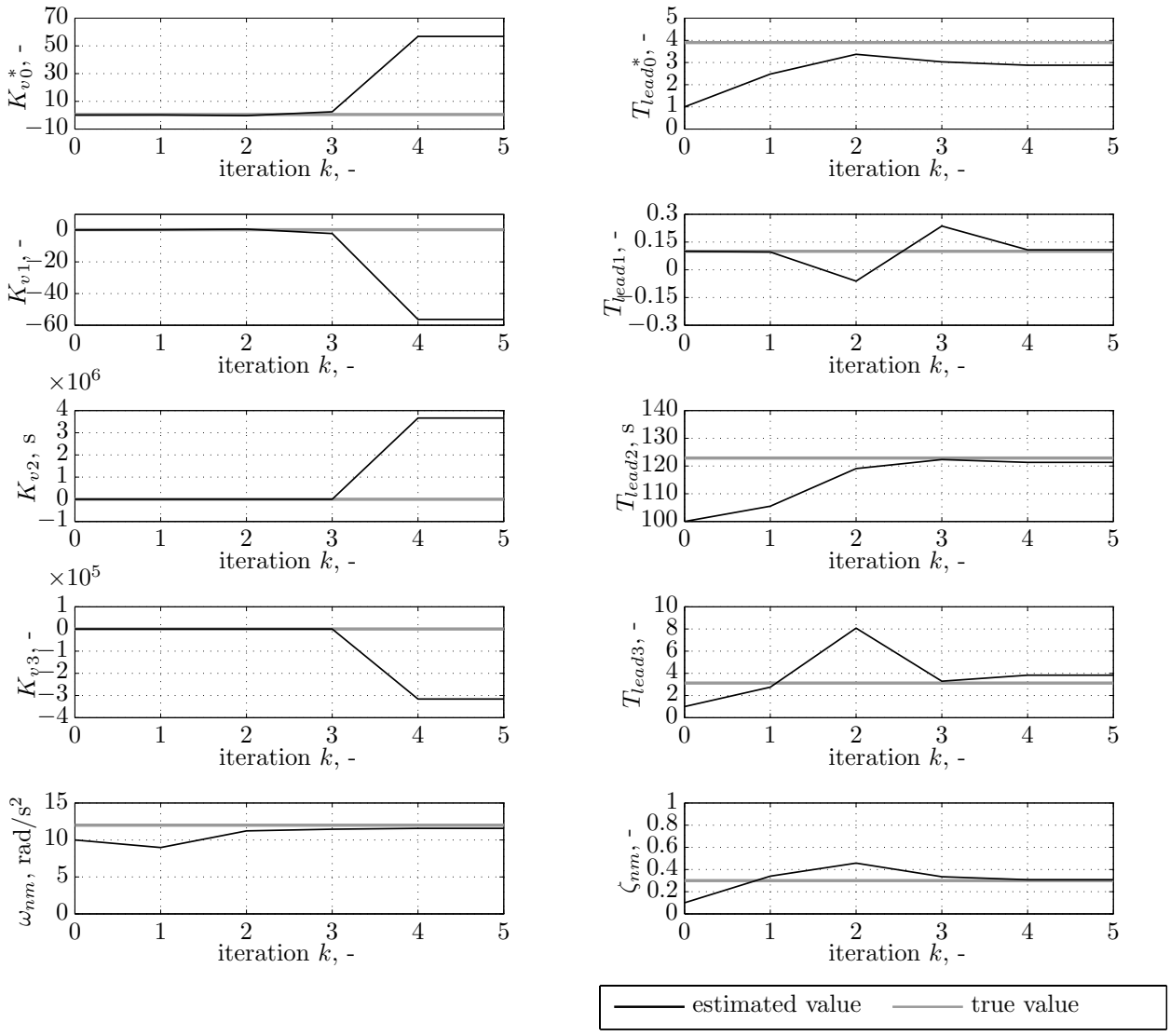


Figure 7-20: Estimated parameter values compared to their true values for a simulation with the Boltzmann sigmoid.

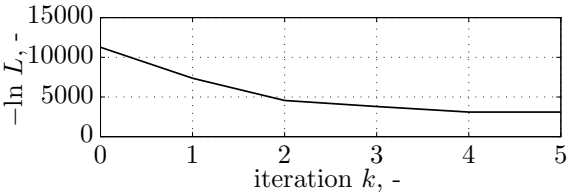


Figure 7-21: Negative log-likelihood for a simulation with the Boltzmann sigmoid.

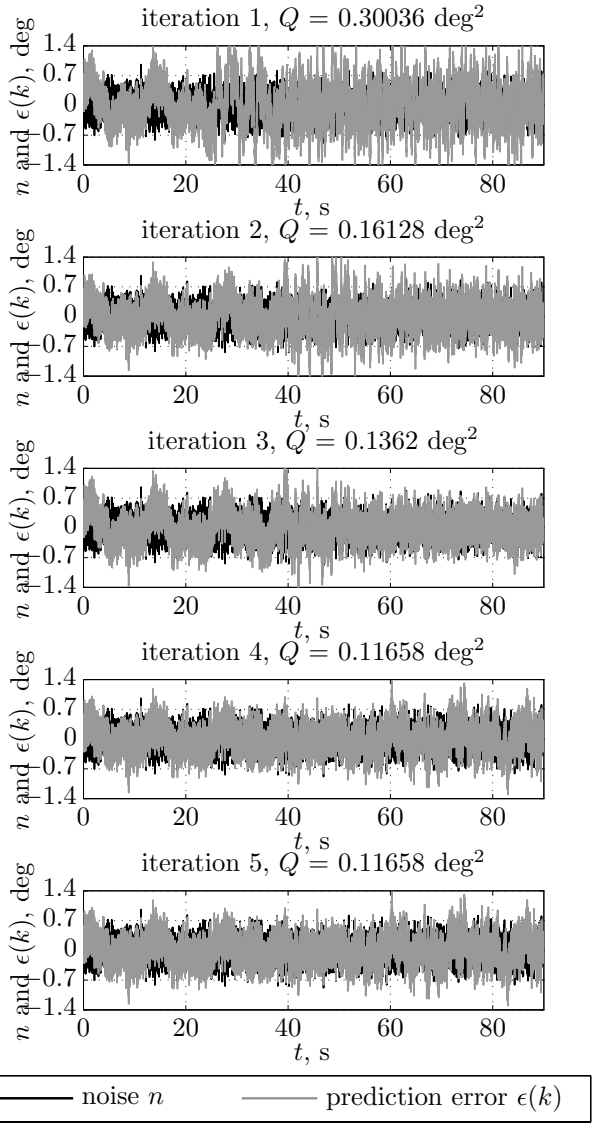


Figure 7-22: The prediction error $\epsilon(k)$ compared to the noise n for a simulation with the Boltzmann sigmoid. ($\sigma_n^2 = 0.07$ deg^2)

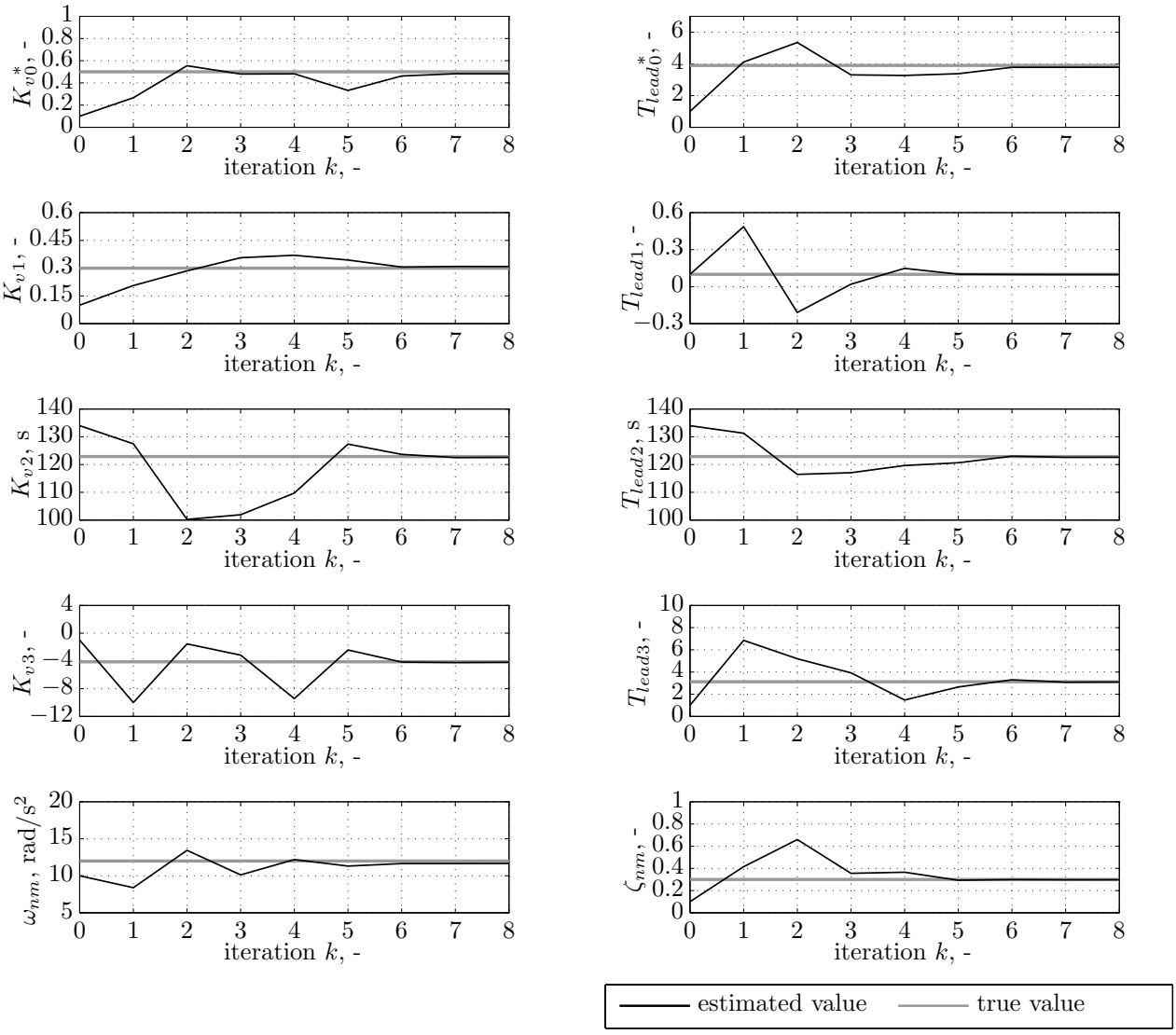


Figure 7-23: Estimated parameter values compared to their true values for a simulation with two Boltzmann sigmoids.

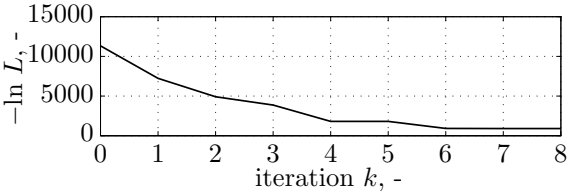


Figure 7-24: Negative log-likelihood for a simulation with the Boltzmann sigmoid.

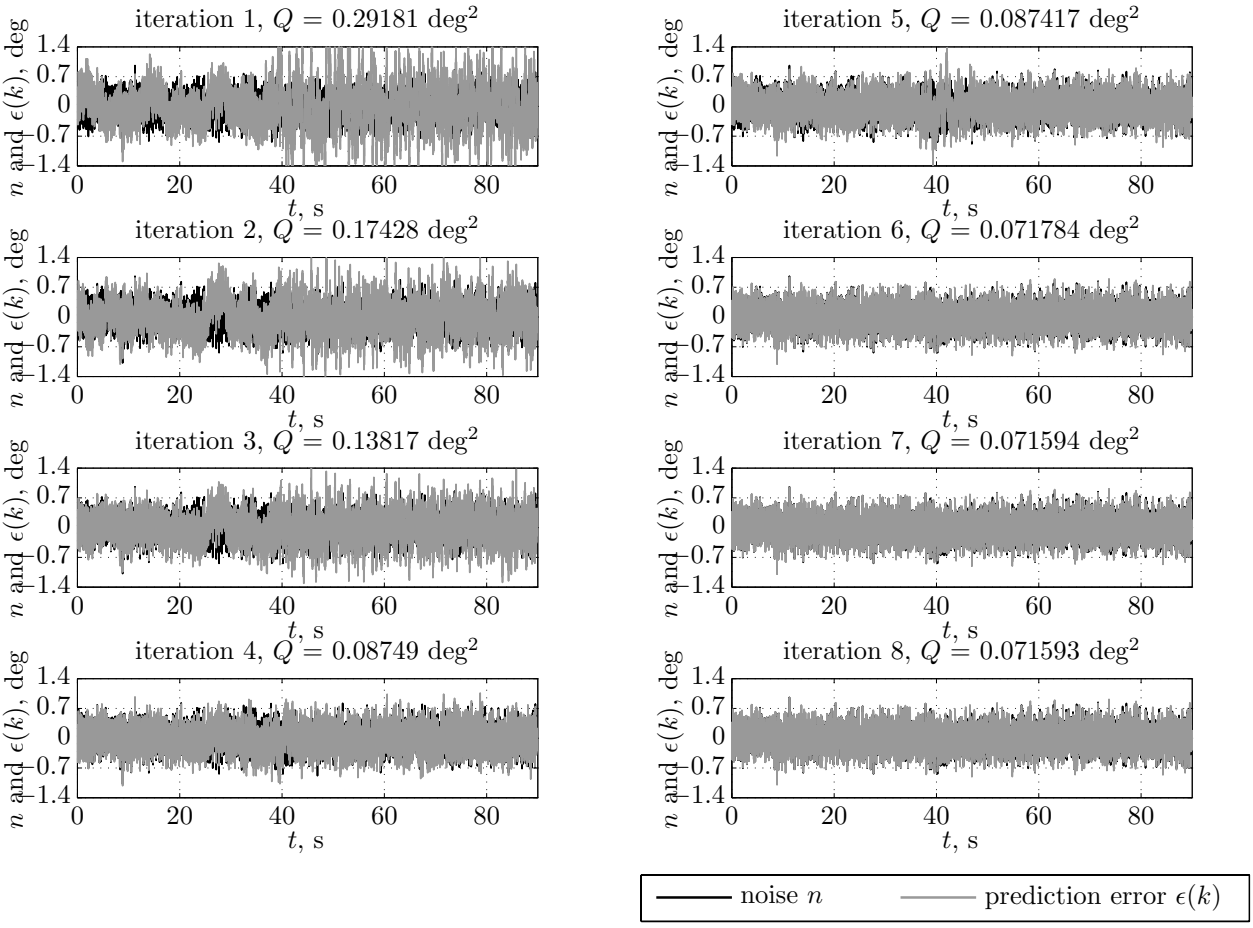


Figure 7-25: The prediction error $\epsilon(k)$ compared to the noise n for a simulation with two Boltzmann sigmoids. ($\sigma_n^2 = 0.07$ deg^2)

is a special structure parameter that when unknown greatly complicates the identification problem. Although the time delay is known for the simulated data in this research, during parameter estimation it is assumed to be unknown. It is thus important keep in mind that adding a time delay might complicate the parameter estimation.

For simulating the data, a Transport Delay block is used in Simulink®. There is an option to set the order of a Padé approximation for linearization in the Transport Delay block, which will not be used, as the Transport Delay block generates a more accurate time delay without the linearization. For the parameter estimation, however, a Padé approximation is necessary to avoid the irrational $e^{-s\tau_v}$ term from causing mathematical problems. The Padé approximation is used to approximate an exponential time delay transfer function with a rational transfer function. The Padé approximation used for the parameter estimation is defined in Equation (7-29) as a ratio between two polynomials:

$$e^{-s\tau_v} \approx H_{pade}(s) = \frac{1 - h_1 s\tau_v + h_2 s^2 {\tau_v}^2 + \ldots \pm h_k s^k {\tau_v}^k}{1 + h_1 s\tau_v + h_2 s^2 {\tau_v}^2 + \ldots + h_k s^k {\tau_v}^k} \tag{7-29}$$

where h_k is the Padé approximation coefficient at discrete instant k. For the modeling of delays associated with human manual control behavior, a fifth-order of approximation is usually chosen for sufficient approximation of the time delay [Zaal et al., 2009]. In Equation (7-30) the transfer function for the human operator used for parameter estimation is stated. The order of approximation of H_{pade} influences the number of states used during parameter estimation. The state-space matrices and the sensitivity matrices increase in size as the order of the Padé approximation increases, resulting in a more complex parameter estimation problem.

$$H_p(s,t) = K_v(t)(1 + T_{lead}(t)s)H_{pade}(s)H_{nm}(s) \tag{7-30}$$

Parameter estimation can now be performed on simulation data generated with a model containing $H_p(s,t)$ as defined by (7-30). For the first run, a first-order Padé approximation is chosen. The effect of adding a time delay is–even with a first-order Padé approximation–apparent: the initial values of K_{v_1}, T_{lead_0}, τ_v, ω_{nm} and ζ_{nm} need to be *very* close to the true values for parameter estimation to produce good results. The upper and lower bounds of these initial values are thus very close to each other, which forms an important drawback. When parameter estimation is performed with a first-order Padé approximation it was also noted that the parameter estimation often suffers from local minima in the likelihood function.

Chapter 8

Final Simulation Model: A Closed-Loop System

In Chapter 7 a step-by-step approach is used to build and expand the simulation model, while explaining the corresponding interim results. In this chapter, the completed simulation model and the associated results will be discussed. In Section 8-1 the system will be augmented with the correct vehicle dynamics. Problems related to integration and time delays will be discussed in Section 8-2 and Section 8-3, respectively. In Section 8-4 an explanation will be given on how zero-mean Gaussian white noise will be converted to a time-varying human operator remnant signal. Following this, a short explanation will be given for changing the values of ω_{nm} and ζ_{nm} in Section 8-5. Finally, the final simulation model will be analyzed by checking its time traces, its Power Spectral Density (PSD) plots, and its frequency responses in Section 8-6.

The final simulation model is modeled after Figure 4-4 with Equation (4-9) representing the system. The Simulink® model used to execute the simulation with its corresponding subsystems can be found in Appendix B. This section is structured as follows. In Section 8-1 vehicle dynamics are added to the open-loop system, after which the system can be closed with a feedback loop. Section 8-2 explains the shift from integration of the state-space equations with the Euler method towards integration with a fourth-order Runge Kutta method. Some important issues related to time delays in the closed-loop system are discussed in Section 8-3. In Section 8-4 the utilization of a low-pass filter for the noise is explained. To get the model in line with the model used in [Zaal & Sweet, 2011], in Section 8-5 the neuromuscular frequency and damping values are set equal to the values used by Zaal & Sweet [2011]. Finally, Section 8-6 is intended to give the reader additional information about the system by discussing the system's time traces and frequency responses.

8-1 Adding Vehicle Dynamics

First of all, to complete the model, the vehicle dynamics H_c are added. The vehicle dynamics are defined as in Equation (4-8) with a time-varying gain $K_c(t)$ and a time-varying vehicle dynamics time constant $T_c(t)$. These two time-varying parameters will be modeled as Boltzmann sigmoids as plotted in Figure 8-1 with the values in Table 8-1. The sigmoids were modeled after the vehicle dynamics parameters in [Zaal & Sweet, 2011]. With the added vehicle dynamics it is possible to form a closed-loop system by subtracting the system output signal y from the target forcing function f_t to calculate the tracking error e, which is used as input for the system. Although the open-loop system with added vehicle dynamics resulted in the expected results, the results of the closed-loop system indicated that the system becomes unstable. After analysis, the problem was traced back to the lead generation of the human operator in H_p. Zaal & Sweet [2011] correctly note that *the values for the lead time constants of the vehicle and pilot model dynamics are equal throughout the run. This means that a perfect pilot compensation is assumed for the double-integrator vehicle dynamics at higher frequencies; that is, the open-loop dynamics ($H_{ol} = H_pH_c$) have single-integrator characteristics for the entire frequency range.* This is correct, but the generation of lead with a T_{lead_1} as small as 0.01 seconds for the portions of the simulation where K_c/s dynamics are controlled in this research, results in unstable system behavior. It is therefore chosen to set T_{lead_1} to a value of zero. When T_{c_1} is set to a value of zero or a small value such as 0.01 seconds, the system also becomes unstable. This difference is caused by the definition of the state-space systems of H_p and H_c of the model. For the state-space system of H_c, fractions with divisions by T_c are used. It is therefore not possible to set T_c to zero. The state-space system of H_p does not suffer from such a mathematical relation. To ensure a stable system, the value of T_{c_1} is thus held at 0.01 seconds, which means that the vehicle dynamics H_c are under influence of a minor contribution of double integration dynamics in comparison with the H_p dynamics. The slope constant of the sigmoids, K_{c_3} and T_{c_3} respectively, were chosen in such a way that the top and bottom were reached at 20 seconds before and after t_{half}: at 102.88 seconds and 142.88 seconds, or at 20.96 seconds and 60.96 seconds when the first realization of T_m is not considered. Note that all other sigmoids in this chapter will be modeled to reach the top and bottom values at these instances.

8-2 From Euler to Fourth-Order Runge-Kutta Integration

When data was generated with the final simulation model for the first time, the control signal u diverges to infinite values as a sinusoid that becomes increasingly larger over time. The problem was traced back to the integration of the state-space and sensitivity equations. The integration of these equations for the data generated in Chapter 7 was done with the Euler method–the simplest form of the Runge-Kutta method. Because of the extra complexity the feedback loop in the final simulation model adds, utilization of the

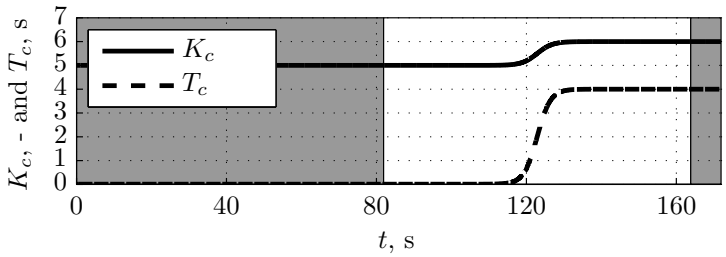


Figure 8-1: The vehicle dynamics gain signal $K_c(t)$ and the vehicle dynamics time constant signal $T_c(t)$ modeled as Boltzmann sigmoids in accordance with Equation (7-24). Only the part in white influences the MLE, as to avoid transient behavior.

Table 8-1: Boltzmann Sigmoid Model Vehicle Dynamics Gain and Time Constant Parameter Values.

Parameter		Value
K_{c_0}	sigmoid top gain constant	6.00 -
K_{c_1}	sigmoid bottom gain constant	5.00 -
K_{c_2}	sigmoid t_{half} gain constant	122.88 s
K_{c_3}	sigmoid slope gain constant	2.02 -
T_{c_0}	sigmoid top time constant	4.00 -
T_{c_1}	sigmoid bottom time constant	0.01 -
T_{c_2}	sigmoid t_{half} time constant	122.88 s
T_{c_3}	sigmoid slope time constant	1.77 -

Euler method resulted in incorrect results. To solve this problem, the fourth-order method of Runge-Kutta is chosen. In Simulink® this is easily done by changing the configuration parameters. For the parameter estimation, however, the Runge-Kutta method has to be implemented manually. The fourth-order method of Runge-Kutta is defined by Equations (8-1) and (8-2) according to Vuik et al. [2007] and is used for the integrations needed to calculate the state-space equations (Equation (2-5)) and the sensitivity equations (Equation (7-16)). Vuik et al. [2007] mention that the fourth-order Runge-Kutta method also has attractive stability properties.

$$y_{k+1} = y_k + \frac{1}{6}[k_{rk_1} + 2k_{rk_2} + 2k_{rk_3} + k_{rk_4}] \tag{8-1}$$

In Equation (8-1), the four k_{rk} parameters are predictors, which are given by Equation (8-2). Also, h^* is the size of the interval.

$$\begin{aligned} k_{rk_1} &= h^* f(t_k, y_k) \\ k_{rk_2} &= h^* f(t_k + \frac{1}{2}h^*, y_k + \frac{1}{2}k_{rk_1}) \\ k_{rk_3} &= h^* f(t_k + \frac{1}{2}h^*, y_k + \frac{1}{2}k_{rk_2}) \\ k_{rk_4} &= h^* f(t_k + h^*, y_k + k_{rk_3}) \end{aligned} \tag{8-2}$$

The necessity of using this fourth-order Runge-Kutta method brings along severe disadvantages in terms of mathematical complexity, which increases the time needed for calculation.

8-3 Time Delay Issues

Because of the closed-loop system, the time delay is now also indirectly fed back and processed with the system input. As mentioned in Section 7-3-2 the introduction of a time delay often comes with drawbacks. During data simulation with the current system it proved impossible to obtain a variance percentage of 10% between control signal u with noise and the signal before noise was added. The variance percentage reaches a maximum value between 4% and 5%. When the Transport Delay block–used in Simulink® to induce the time delay–is removed from the system, the 10% variance percentage is easily obtained by using a gain to get the noise up to this value. The problem was traced back to the values used to generate the visual-perception gain K_v and the phase margin ϕ_m. In Section 7-2-2 K_{v_0} and K_{v_1} were set to values of 0.8 and 0.3 respectively. In the final system, the open-loop phase margin ϕ_m is equal to 8.24 degrees for a K_{v_0} of 0.8. The phase margin reaches a value of zero when K_{v_0} is 0.82. Because the chosen value of 0.8 contributes to a phase margin close to zero, the system shows highly oscillatory behavior, which is the cause of the problem with the variance percentage. In [Zaal & Sweet, 2011] the values of K_v were delicately chosen as 0.7 and 0.4 to generate the system. When these values are used for the K_v sigmoid, the system poses no problems for scheduling the variance percentage to the desired 10%. The open-loop phase margin ϕ_m has a value of 28.92 degrees for the system with K_{v_0} is 0.7. Using the values from [Zaal & Sweet, 2011], the newly defined K_v sigmoid is plotted in Figure 8-2 and its corresponding values are available in Table 8-2. Note that the new visual-lead time sigmoid, which was also changed in Section 8-1, is also plotted in Figure 8-2. The corresponding values of the visual-lead time sigmoid are also stated in Table 8-2.

Another issue posed by the time delay, is found in the difference between the time delay the Transport Delay block in Simulink® generates, compared to the Padé approximation used during parameter estimation. First of all, an offset of 0.02 seconds is added to the parameter estimation initial value for the time delay to compensate for the difference with

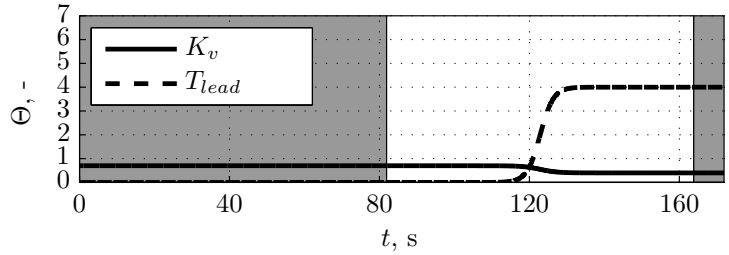


Figure 8-2: The new visual-perception gain signal $K_v(t)$ and the new visual-lead time signal $T_{lead}(t)$ modeled as a Boltzmann sigmoid in accordance with Equation (7-24). Only the part in white is used for MLE to avoid transient behavior.

Table 8-2: The New Parameter Values for the Visual-Perception Gain and the Visual-Lead Time Boltzmann Sigmoid.

Parameter		Value
K_{v_0}	sigmoid top gain constant	0.70 -
K_{v_1}	sigmoid bottom gain constant	0.40 -
K_{v_2}	sigmoid t_{half} gain constant	122.88 s
K_{v_3}	sigmoid slope gain constant	-2.30 -
T_{lead_0}	sigmoid top time constant	4.00 -
T_{lead_1}	sigmoid bottom time constant	0.00 -
T_{lead_2}	sigmoid t_{half} time constant	122.88 s
T_{lead_3}	sigmoid slope time constant	1.77 -

the time delay of the Transport Delay block while the other initial values are all set to their true values. The value of the Transport Delay block time delay is still 0.20 seconds, but for the parameter estimation initial value a time delay of 0.21 seconds is used as the 'true value' instead. The respective control signals are plotted in Figure 8-3. Just like before, to avoid transient behavior only the area between 81.92 seconds and 163.84 seconds is used. At first sight, it might seem that the control signal from the simulation and control signal from parameter estimation completely overlap. They do most of the time, but sometimes there is a small difference due to the difference in calculation of the time delay. This difference becomes bigger around the change in dynamics for the part where control of double integrator dynamics is considered. Although the Transport Delay block has an option to linearize the delay with a Padé approximation, this did not help in reducing the difference. An attempt to solve this problem was made to replace the Transport Delay block with the Padé approximation from the parameter estimation. This, however, resulted in results with highly oscillatory behavior attributed to the integration in Simulink®. Because of the difference between the Transport Delay block of the model and the Padé approximation used during parameter estimation, an analysis is done to determine what the significance of the difference between the time delay for the simulated data and the time delay used for parameter estimation is. The prediction error $\epsilon(k)$ between the estimated control signal $\hat{u}$ and the simulated control signal u without noise is plotted in

Figure 8-4. Note that 0 seconds in this plot corresponds to 81.92 seconds in Figure 8-3, as the first 81.92 seconds are cut off because this signal is not used. The same prediction error $\epsilon(k)$ is plotted in Figure 8-5 to illustrate that it is small compared to an amount of filtered noise n (the addition of filtered noise will be further explained in Section 8-4). It is notable that the prediction error is smaller than 0.05 degrees for the first 30 seconds, until it nears the sigmoid's t_{half} value, which is equal to 40.96 seconds. It thus seems that the difference between the Transport Delay of Simulink® and the Padé approximation used in the parameter estimation becomes larger when the sigmoids start altering the dynamics. From the plot it also becomes clear that the error from 0 to 30 seconds is somewhat bigger compared to the error after 50 seconds. Because the prediction error is small compared to the noise, as can be verified from Figure 8-5, the effect due to this modeling difference is neglected.

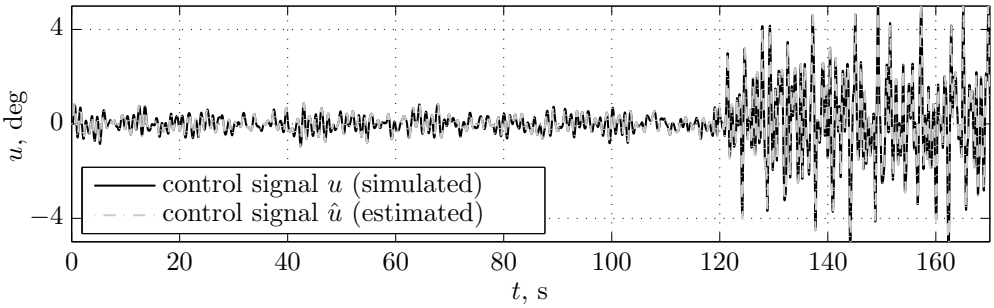


Figure 8-3: Control signal data from the simulation and from parameter estimation (without noise).

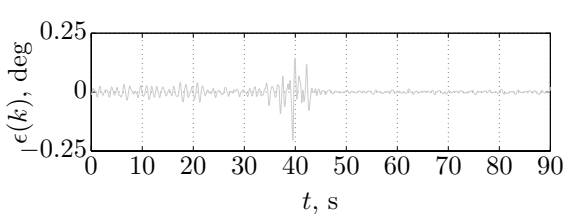


Figure 8-4: The prediction error plotted here is the difference between the simulated control signal u (without noise) and the estimated control signal $\hat{u}$.

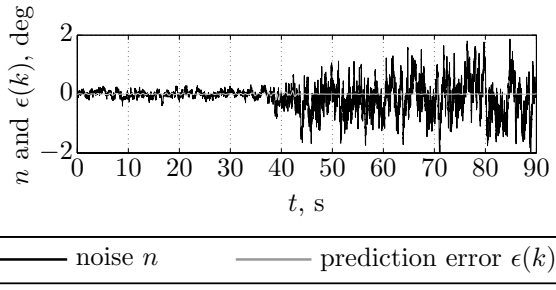


Figure 8-5: The prediction error from Figure 8-4 plotted against filtered noise n.

8-4 Filtering the Noise: A Human Operator Remnant Signal

Up until now, time-invariant white noise has been added to the control output u to simulate a human operator remnant signal. However, pure static white noise does not exist in

nature. To model a more realistic human operator remnant signal, the zero-mean Gaussian white noise signal is passed through a time-varying low-pass filter in accordance with [Zaal & Sweet, 2011] and Equation (4-10) with a T_n of 0.20 seconds. The time-varying low-pass filter is necessary to keep the remnant variance percentage n_p around 10% for both the single-integrator and the double-integrator dynamics part. The time-varying noise gain $K_n(t)$ is plotted in Figure 8-6. The corresponding values are available in Table 8-3 with the top and bottom gain constants modeled to keep the remnant variance percentage n_p at 10% before and after the change the sigmoids induce. Note that because the different parameters change around t_{half}, the variance percentage around this t_{half} may deviate from the 10% value. The remnant signal n was plotted before in Figure 8-5. Note the difference of the filtered noise with the pure white noise band used in previous chapters. For the generation of the remnant signal Seed 19 is used in Simulink® with a variance σ_n^2 of 1 deg^2. Due to the dynamics of the low-pass filter, it is not expected that the variance of the residuals for parameter estimation will tend to go to the value for σ_n^2.

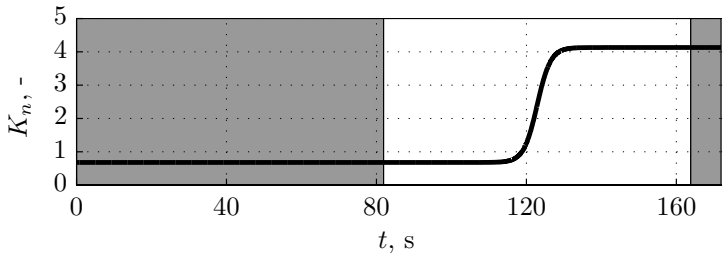


Figure 8-6: The noise gain signal $K_n(t)$ modeled as a Boltzmann sigmoid in accordance with Equation (7-24). Only the part in white influences the MLE, as to avoid transient behavior.

Table 8-3: Boltzmann Sigmoid Model Noise Gain Values.

Parameter		Value
K_{n_0}	sigmoid top gain constant	4.13 -
K_{n_1}	sigmoid bottom gain constant	0.68 -
K_{n_2}	sigmoid t_{half} gain constant	122.88 s
K_{n_3}	sigmoid slope gain constant	1.79 -

8-5 Changing the Neuromuscular Frequency and Damping

To get the final model completely in line with the model used in [Zaal & Sweet, 2011], the neuromuscular frequency ω_{nm} and neuromuscular damping ζ_{nm} need to be changed as well. In Chapter 7 values of 12.00 rad/s^2 and 0.30 were used, respectively, based on [Nieuwenhuizen et al., 2008]. However, in [Zaal & Sweet, 2011] a value of 10.00 rad/s^2 is used for ω_{nm} and a value of 0.20 for ζ_{nm}. For clarity, these values are stated in Table 8-4.

Table 8-4: Adjusted Neuromuscular Frequency and Damping Values.

Parameter		Value
ω_{nm}	neuromuscular frequency	10.00 rad/s^2
ζ_{nm}	neuromuscular damping	0.20 -

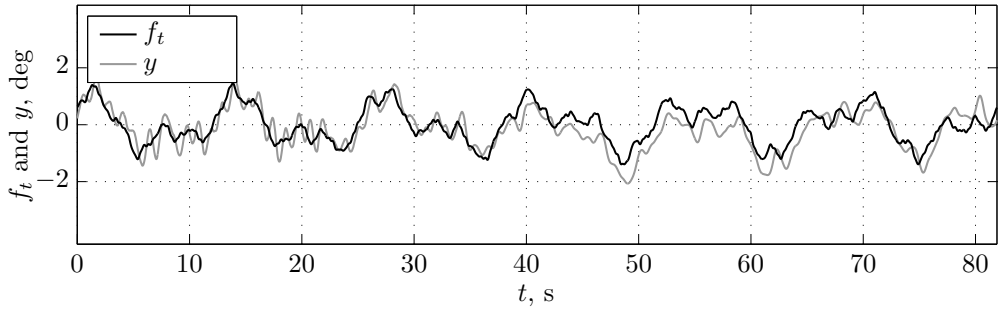
8-6 Time Traces and Frequency Responses of the System

Now that the final simulation model is complete, simulation data can be generated and the system created can be analyzed. The most important time traces of the system are plotted in Figure 8-7. In Figure 8-7a the target forcing function f_t is plotted together with the system output signal y. From the plot it become clear that the human operator model controls the vehicle in such a way that the target forcing function is followed. As expected a difference is clearly visible between the single-integrator dynamics before the change induced by the sigmoids at 20.96 seconds and the double-integrator dynamics after the change induced by the sigmoids at 60.96 seconds. During the single-integrator dynamics, the human operator causes overshoots, but adapts fast enough to follow the target forcing function. After the change to double-integrator dynamics, it is clearly harder for the human operator to let the vehicle follow the target forcing function. This is phenomenon is also visible from the error e between the target forcing function and the system output signal. This error is plotted in Figure 8-7b.

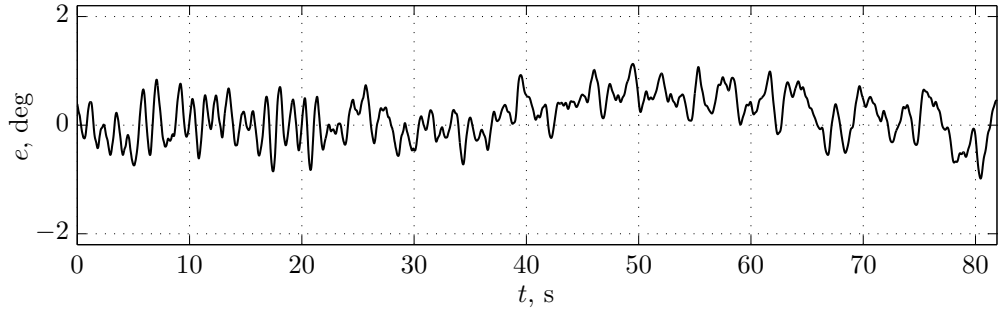
For the single-integrator dynamics, the error varies rapidly around zero degrees. After the sigmoids have induced the change from single-integrator to double-integrator dynamics at 60.96 seconds, the error is less often reduced to zero degrees, indicating that it is harder for the human operator to reduce the error. Although it might seem that the error for double-integrator dynamics does not vary around zero as much as the error for the single-integrator dynamics, this is not the case. During tests with longer runs for the double-integrator dynamics the error indeed seems to vary around the zero mean. In addition, the drift in the error from it's zero mean between 40 to 60 seconds is caused by the sigmoid. In Figure 8-7c the control signal u (including remnant) is plotted together with the remnant signal n. The difference between the dynamics before and after the change induced by the sigmoids is clearly visible here. The control signal is relatively small during the single-integrator dynamics and becomes bigger after the change due to the sigmoids. With a variance percentage of approximately 10% over the complete signal, the remnant signal n stays small in relation to the control signal u for both single- and double-integrator dynamics.

For further analysis of the final system, the system is 'split up' into two systems: System I describes the system with single-integrator dynamics before the sigmoids induce a change, System II describes the system with double-integrator dynamics after the sigmoids have induced a change. The corresponding parameter values per system are stated in Table 8-5. Note that the vehicle dynamics H_c in System I are under the influence of minor

(a) The target forcing function f_t plotted with the system output signal y.



(b) The error signal e.



(c) The control signal u plotted with the remnant signal n.

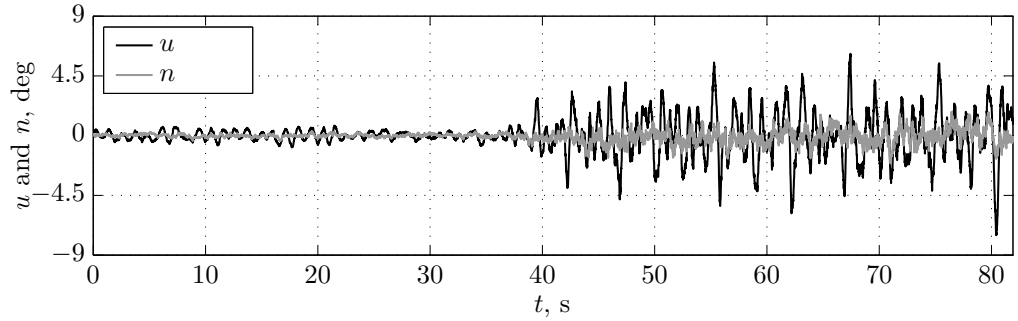


Figure 8-7: Time traces of the simulated signals.

double-integrator dynamics because of the value of T_c, which is very small: 0.01 seconds. The reason to have a non-zero value for T_c is that the state-space matrices for the vehicle dynamics consist of division by T_c. For the pilot dynamics H_p the zero value of T_{lead} does not result in any problems.

Analysis of the two systems–which together compose the final system–can now be done. In Figure 8-8 the PSD for the most important signals are plotted for System I and II. The target forcing function f_t clearly shows the distinct peaks for which the multisine

was carefully constructed (explained in Section 7-1-1). The presence of the target forcing function in the closed-loop system is also visible in the PSDs of the output signal y, the error e and the control signal u.

Table 8-5: System I & II Parameter Values.

Parameter	System I	System II
K_v	0.70 -	0.40 -
T_{lead}	0.00 s	4.00 s
K_c	5.00 -	6.00 -
T_c	0.01 s	4.00 s
K_n	0.68 -	4.13 -
T_n	0.20 s	0.20 s
τ_v	0.20 s	0.20 s
ω_{nm}	10.00 rad/s^2	10.00 rad/s^2
ζ_{nm}	0.20 -	0.20 -

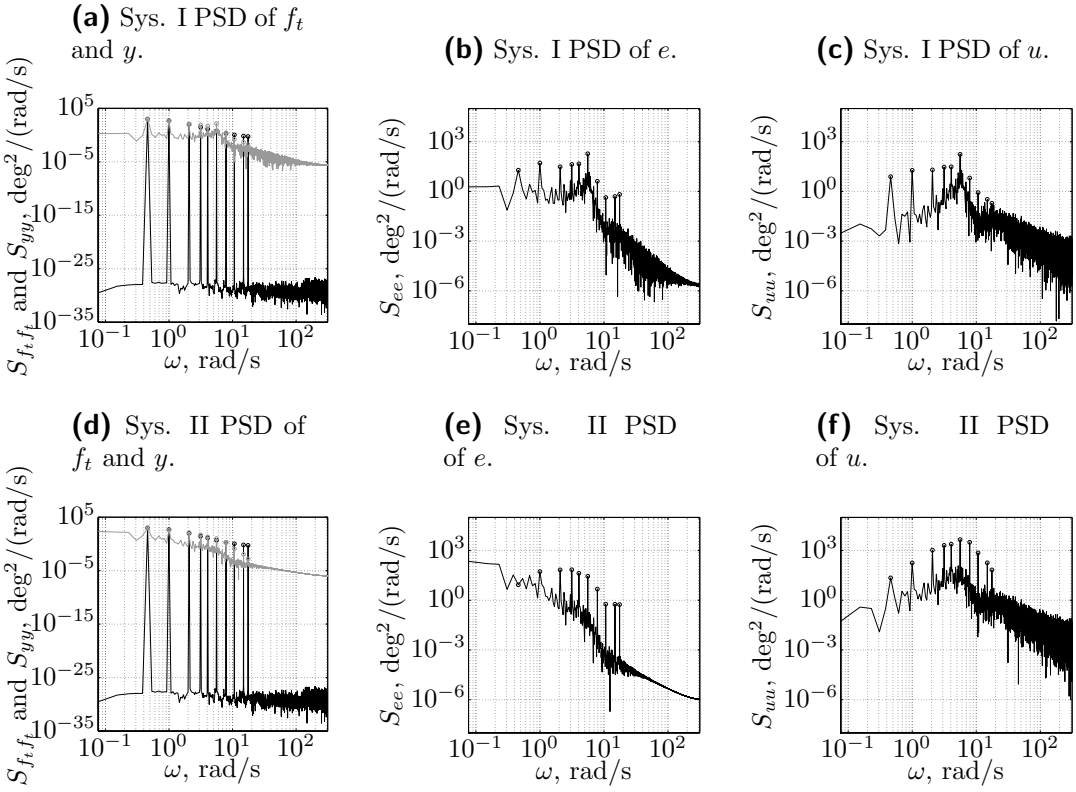


Figure 8-8: Power Spectral Densities of target forcing function f_t (black), output signal y (gray), error signal e and control signal u.

The difference between the single-integrator dynamics of System I and the double-integrator dynamics of System II become visible from the frequency responses of the

two systems, which are plotted in Figure 8-9. For the single-integrator dynamics $|H_p(j\omega)|$ is relatively flat compared to the double-integrator $|H_p(j\omega)|$, which has a steeper slope. The difference between the single-integrator dynamics of System I and the double-integrator dynamics of System II is most clearly visible from the steepness of the slopes of $|H_c(j\omega)|$. The frequency responses of the pilot dynamics $H_p(j\omega)$, vehicle dynamics $H_c(j\omega)$ and the open-loop dynamics $H_{ol}(j\omega)$ from the simulated data are plotted against the analytical frequency response. The responses from the simulated data follow the analytical responses quite neatly, which indicates that the simulated data is correct.

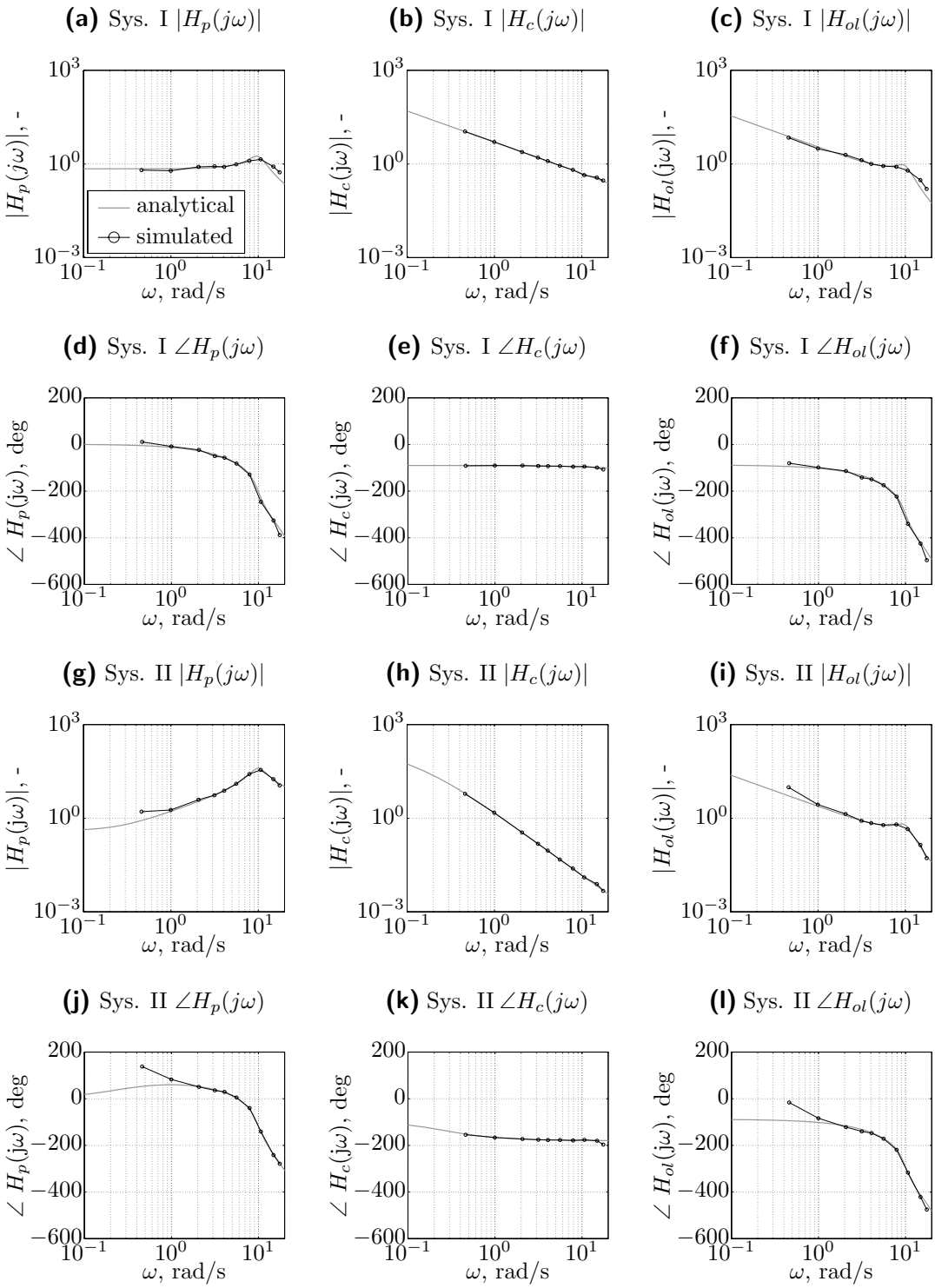


Figure 8-9: Frequency responses of $H_p(j\omega)$, $H_c(j\omega)$, and $H_{ol}(j\omega)$ for System I and System II as defined by Table 8-5.

Chapter 9

Parameter Estimation Results

With the simulated data from the control task in Section 8, parameter estimation can be performed. For comparison, the proposed MLE method is compared with a sliding time window MLE method. Both methods will be discussed briefly in Section 9-1. The results of the parameter estimations will be discussed in Section 9-2. Simulations were performed for three different noise realizations with different parameter variance percentages θ_p and different remnant variance percentage n_p. Following these results, the outcome of further investigation on the influence of the remnant n and the initial parameter values will be discussed in Section 9-3, and the results of an evaluation on the effect of local minima will be examined in Section 9-4.

9-1 Comparing Two Time-Varying Maximum Likelihood Estimation Methods

In this thesis, a time-domain identification procedure based on MLE is proposed for the estimation of time-varying pilot vehicle system parameters. To see how well this time-varying MLE method compares to other methods, it will be compared to the sliding time window MLE method used in Zaal & Sweet [2011]. In this section both methods will be briefly introduced. It is possible to compare the methods relative to each other with the use of the Variance Accounted For (VAF). The VAF is used to compare the estimated control signal $\hat{u}$ of one of the methods to the corresponding control signal u. If the estimated control signal equals the simulated control signal, the VAF will be 100%. Note that the line-search algorithm as proposed in Section 7-1-2 is used for both methods.

9-1-1 Method 1: Time-Varying Model Identification

For the time-varying model identification procedure the entire data set of a single simulation run of 81.92 s will be used. As discussed in Section 7-1, for the control task in this thesis, data from the second simulation run is used to avoid transient behavior in the first simulation run. The parameters that will be estimated are stated in the parameter vector in (9-1).

$$\Theta = \begin{bmatrix} K^*_{v_0} & K_{v_1} & K_{v_2} & K_{v_3} & T^*_{lead_0} & T_{l_1} & T_{l_2} & T_{l_3} & \tau_v & \omega_{nm} & \zeta_{nm} \end{bmatrix}^\top \tag{9-1}$$

Before the identification procedure is initiated, initial values have to be given to the parameters, which are used for the first parameter estimation iteration. Although a randomization of these initial values is usually applied, the choice is made to offset each parameter with an equal percentage from the true values that were used to create the simulated data. This is done in order to make it easier to compare the results. The true values are labeled as 100%, so when all parameters are subjected to a positive offset of 10% this equals a parameter variance θ_p of 110%. Note that T_{lead_1} has a value of zero. To be able to give T_{lead_1} an offset, it is set to 0.01 for the instances that it is given an offset by θ.

9-1-2 Method 2: Sliding Time Window Identification

In [Zaal & Sweet, 2011] the sliding time window MLE method is described as follows:

To estimate time-varying pilot model parameters in the current study, the MLE optimization is performed at every time step t_i using a sliding time window of length Δt. Choosing a Δt that is too small will decrease the accuracy of estimated parameters related to low-frequency dynamics. A Δt that is too large will reduce the method's ability to detect small variations in pilot model parameters. In the current study, the length of the time window is chosen to be 20 s.

The sliding time window MLE method in [Zaal & Sweet, 2011] is performed every 2 seconds, and because of the width of the window it is initiated at $t = 12.0$ seconds and stops at $t = 80.0$ seconds. To compare the sliding time window MLE method with the proposed method in this research, the sliding time window method will be performed according to the definition in [Zaal & Sweet, 2011], but on the simulated data generated by control task defined in this chapter. In Figure 9-1 a representation of the sliding time window MLE method is given as defined by Zaal & Sweet [2011]. Because of the sliding time window, this method only observes a certain part of the data and the parameters are assumed to have constant values for the data in Δt. The parameter vector in Equation 9-2, therefore, only consists of five parameters.

$$\Theta = \begin{bmatrix} K_v & T_l & \tau_v & \omega_{nm} & \zeta_{nm} \end{bmatrix}^\top \tag{9-2}$$

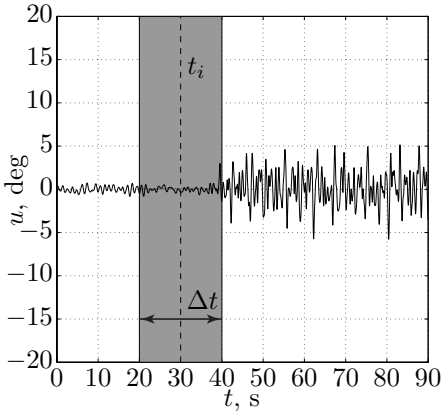


Figure 9-1: Sliding time window MLE method.

In [Zaal & Sweet, 2011] the initial parameter set was constructed from the simulated pilot model parameters at each time step. Because only $K_v(t)$ and $T_{lead}(t)$ are considered in this parameter estimation problem and because the proposed time-varying model MLE method only uses one initial parameter which is set at the beginning instead of at each time step, it is chosen to set the initial values of the sliding time window method to the average of the *top* and *bottom* values of these sigmoids multiplied by the same parameter variance θ_p that is chosen for the proposed MLE method.

9-2 Results

The results of parameter estimation are plotted in Figure 9-2. The plots show the results obtained by setting the initial parameter values to a parameter variance θ_p of 110%. For the estimation, simulated data is used, which is generated by the control task in Section 8 without a remnant ($n_p = 0\%$). The four estimated sigmoid parameters for both $K_v(t)$ and $T_{lead}(t)$ were used to plot the sigmoids in Figure 9-2a and Figure 9-2b. For the proposed MLE method (indicated by MLE_1) the sigmoidal shape is clearly visible. Without any human operator remnant signal, the parameter estimation with the sigmoids is perfect. The sliding time window MLE method (indicated by MLE_2) also produces estimates which come close to the values provided by the simulated data, but it shows a lot of peaks and fluctuation around these values. An evaluation of MLE_1 and MLE_2 for the estimation of the fixed, time-invariant parameters in Figure 9-2c, Figure 9-2d, and Figure 9-2e shows similar results. With a VAF_1 of 100% and a VAF_2 of 82.3% the proposed method is significantly better than the sliding time window method. Note that VAF_1 is only calculated for the measurement time that the sliding time window method acts (from 12 seconds to 80 seconds). VAF_2 is calculated by averaging the total sum of the VAFs for each time window estimation instance.

When the remnant variance percentage n_p is increased, however, a clear difference with the system without remnant can be seen. To show this, $K_v(t)$ and $T_{lead}(t)$ are plotted in Figure 9-3 for estimation performed on simulated data with remnant variance percentages of 0%, 5%, and 10%. The plots show that both MLE methods return worse estimation results when the remnant contribution is increased. For the results with an n_p of 5% and 10%, the *top*-value of $K_v(t)$ and the *bottom*-value of $T_{lead}(t)$ seem to be estimated correctly, but their respective *bottom*- and *top*-values do not follow the simulated sigmoids. The sliding time window MLE method seems to have similar problems when estimating the changing behavior (between 20.96 seconds and 60.96 seconds), but rapidly stabilizes and starts fluctuating around the simulated values after 60.96 seconds. The plots of the fixed values τ_v, ω_{nm}, and ζ_{nm} are not shown here, but indicate good results for MLE_1. Despite the fact that MLE_2 shows a lot of peaks and fluctuation around the fixed values, the estimation for the fixed values is also adequate. With a VAF_1 of 95.5% and a VAF_2 of 76.6% for a n_p of 5%, the proposed method still produces better results. For a n_p of 10% this is also the case, with a value of 90.8% for VAF_1 and a value of 71.0% for VAF_2. In conclusion, the addition

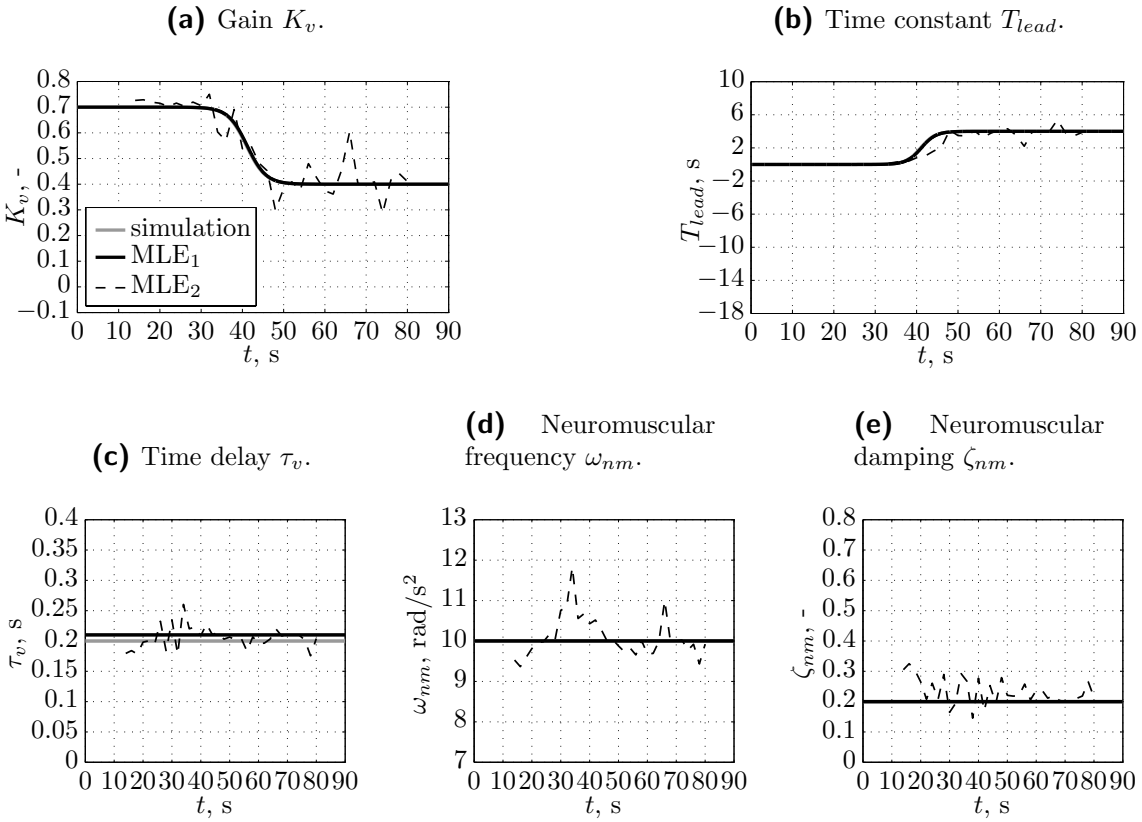


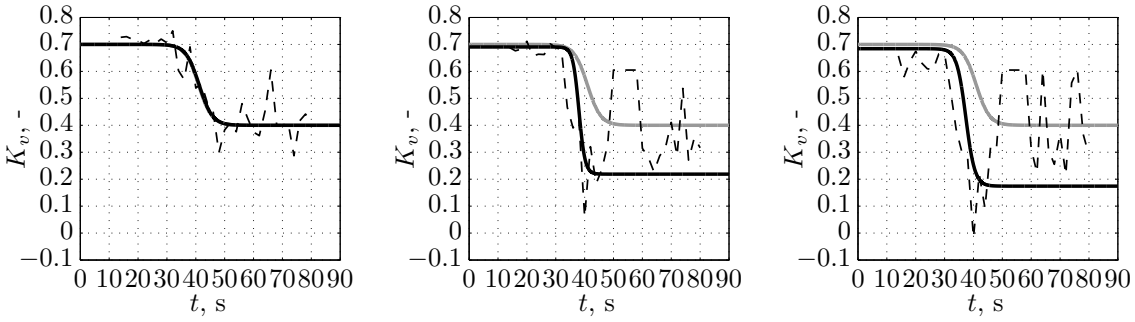
Figure 9-2: Parameter estimation results for the proposed MLE method (MLE_1) and the sliding time window MLE method (MLE_2). For the estimation, the initial parameter values were all subjected to a θ_p of 110%.

of a remnant to the system has adverse effects on the parameter estimation. There is a clear difference in performance between the two methods and MLE_1 consistently performs better. It is notable that with an increase in human operator remnant, the ΔVAF between the two methods also increases.

(a) K_v, 0% remnant variance percentage.

(b) K_v, 5% remnant variance percentage.

(c) K_v, 10% remnant variance percentage.



(d) T_{lead}, 0% remnant variance percentage.

(e) T_{lead}, 5% remnant variance percentage.

(f) T_{lead}, 10% remnant variance percentage.

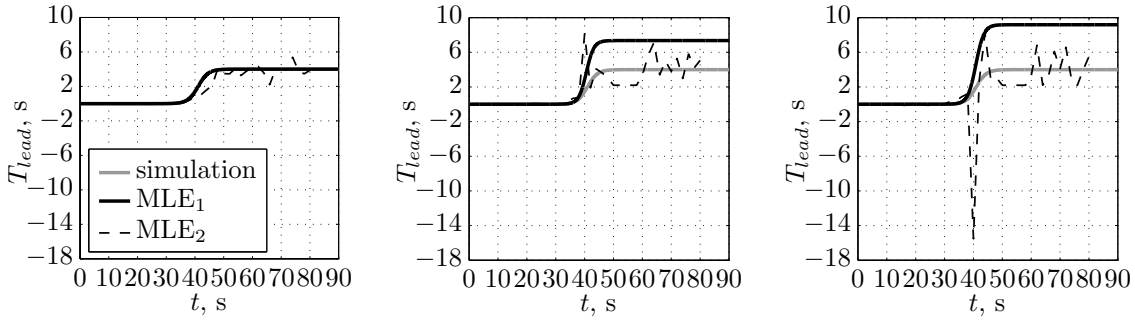


Figure 9-3: Parameter estimation results for sigmoids K_v and T_{lead} under the influence of different amounts of remnant variance percentage n_p. For the estimation, the initial parameter values were all subjected to a θ_p of 110%.

The results obtained for parameter estimation up until now are based on simulation data in which the time-varying parameters are modeled as Boltzmann sigmoids. It is important to understand that for the parameter estimation in this case, the time-varying parameters are estimated as Boltzmann sigmoids. The time-varying dynamics that can be modeled with a sigmoid are expected to be representative for a transition in human control behavior between two dynamic systems. However, it is interesting to see in what way the proposed estimation method with sigmoids will react on simulation data in which the time-varying parameters are not modeled as a Boltzmann sigmoid. To check this, the Boltzmann sigmoid in the simulation data is replaced with a linear transition as depicted by Figure **??** and

estimated with the proposed estimation method. The parameter estimation results of both the proposed method MLE_1 and the sliding time window method MLE_2 are shown in Figure 9-4 for K_v and T_{lead} with remnant variance percentages n_p of 0%, 5% and 10%. The results are comparable to those of Figure 9-3, except for the results of K_v with an n_p of 0% in which the sigmoid seems to be influenced by biases. It is also worth mentioning that the sliding time window method MLE_2 seems to perform better than MLE_1 for T_{lead}. In general, however, the proposed method still performs better in terms of the VAF. For an n_p of 0%, VAF_1 equals 96.4% and VAF_2 equals 84.8%. For an n_p of 5%, VAF_1 has a value of 95.5% and VAF_2 has a value of 79.8%. And, consequently, for an n_p of 10%, VAF_1 amounts to 91.6% and VAF_2 to 75.4%. The big differences between the VAFs are not directly visible from Figure 9-4, but are more evident from the estimation of the fixed values τ_v, ζ_{nm}, and ω_{nm} where MLE_2 performs drastically worse for this simulation data set.

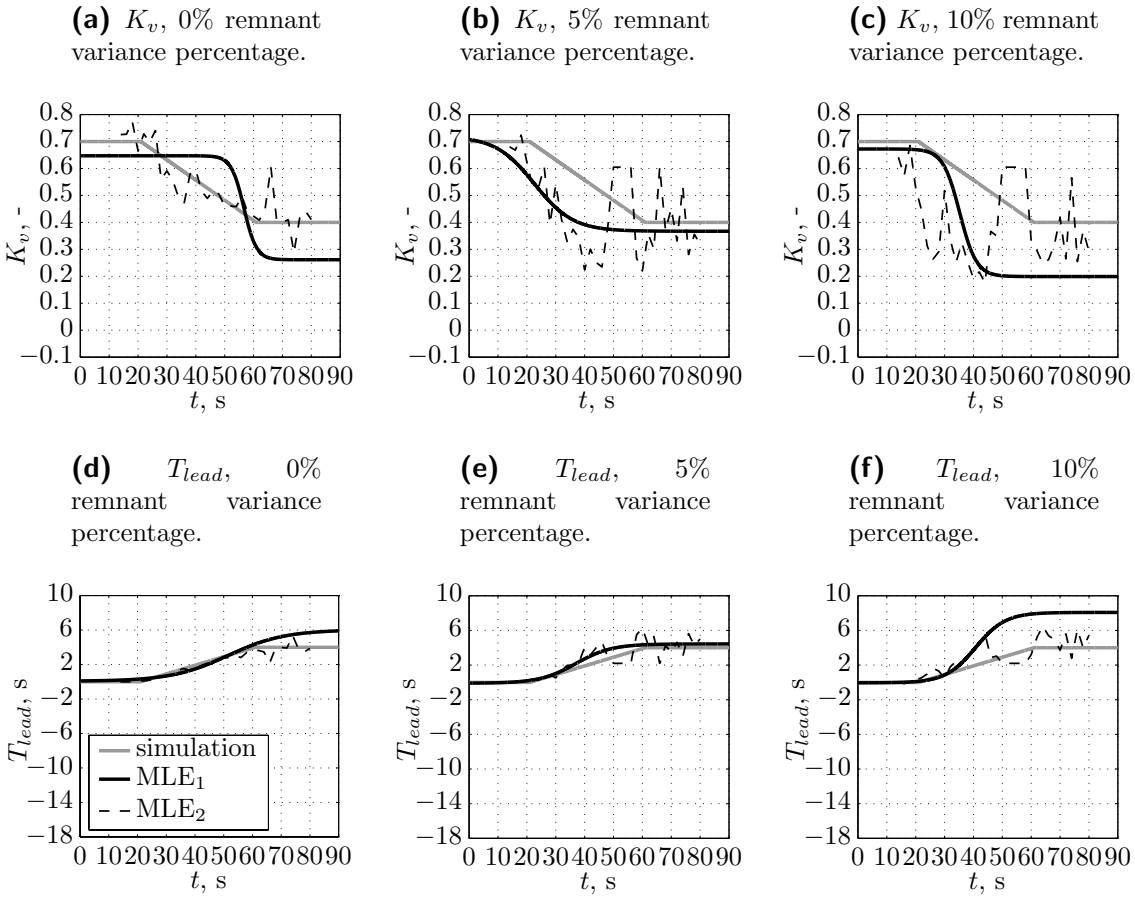


Figure 9-4: Parameter estimation results for ramp signals K_v and T_{lead} under the influence of different amounts of remnant variance percentage n_p. For the estimation, the initial parameter values were all subjected to a θ_p of 110%.

9-3 Influence of Remnant and Initial Parameter Values

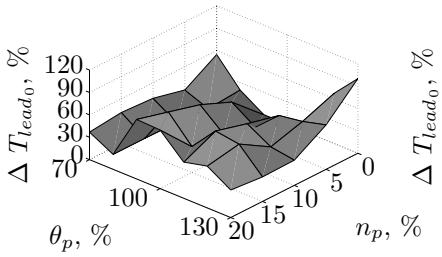
In Section 9-2 it became clear that there are two important factors that have an influence on the accuracy of the parameter estimation results: the amount of remnant signal in the simulated data (the remnant variance percentage n_p) and the offset given to the initial parameter values for parameter estimation (the parameter variance percentage θ_p). Although a larger remnant has a detrimental effect on the results, the parameter variance percentage θ_p showed to have a small effect on parameter estimation. To verify this, and to understand how the remnant variance percentage and the parameter variance percentage influence the proposed MLE method, twelve sets of simulation data for three different white noise realizations are generated. Table 9-1 gives an overview of the characteristics of these twelve simulation data sets. For the complete control task as defined in Section 8, six sets are generated: three closed-loop system sets and three open-loop system sets. The difference in these sets of three is the way in which noise is added to the system: as filtered noise through the shaping filter (i.e., the remnant), as a zero-mean Gaussian white noise signal multiplied by a time-varying sigmoid K_n to keep the remnant variance percentage of the single-integrator dynamics equal to that of the double-integrator dynamics, or as a time-invariant zero-mean Gaussian white noise signal. Two other sets of three are generated to see if MLE is influenced by the number of parameters, and to see if the single-integrator (System I*) and double-integrator (System II*) dynamics behave differently from the complete system. Compared to Section 9-2, the number of parameters is reduced as T_{lead} is now set to a fixed value. The system parameters for System I* and System II* are given in Table 9-2. Note that System I* and System II* differ from System I and System II, as for the former systems, $K_v(t)$ is modeled as a sigmoid. Furthermore, the values of $K_n(t)$ are not given in the table, as the values are dependent on the kind of noise and the remnant variance percentage n_p for each set in Table 9-1.

Table 9-1: Overview of the characteristics of twelve packages of simulation data. Closed-loop is abbreviated to 'CL' and open-loop to 'OL'.

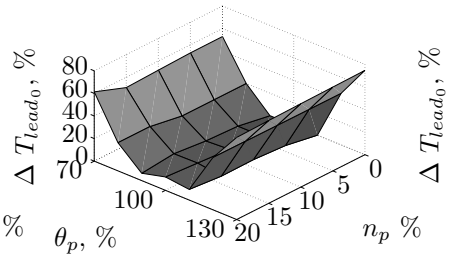
Complete System		System I*	System II*
CL filtered noise	OL filtered noise	CL filtered noise	CL filtered noise
CL varying white noise	OL varying white noise	CL varying white noise	CL varying white noise
CL constant white noise	OL constant white noise	CL constant white noise	CL constant white noise

The twelve simulation data sets are evaluated by estimating parameters for a θ_p between 70% and 130%, and for a n_p ranging from 0% to 20%. For each parameter a three-dimensional plot is made, where θ_p and n_p are set out against the percentual difference $\Delta\theta$ of estimated parameter $\hat{\theta}$ in comparison with the true parameter value θ_0. The plots for all twelve sets can be found in Appendix C. Note that from Figure C-13 to Figure C-13 in this appendix, another set of plots can be found. For each of these plots, all parameters, except one, are kept at a θ_p of 100%. From these plots it can be deduced that changing one parameter also has its influence on other parameters. To illustrate the main findings

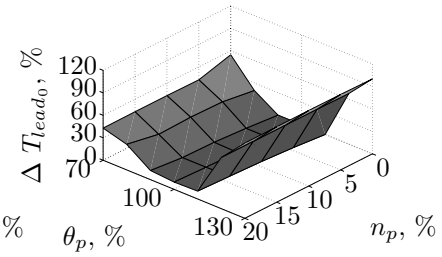
(a) Closed-loop compl. system filtered noise.



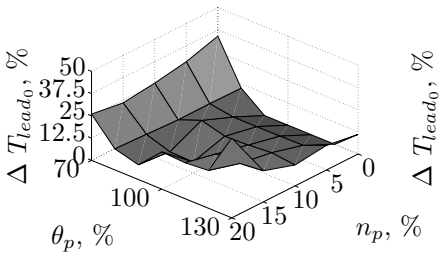
(b) Closed-loop compl. system varying white noise.



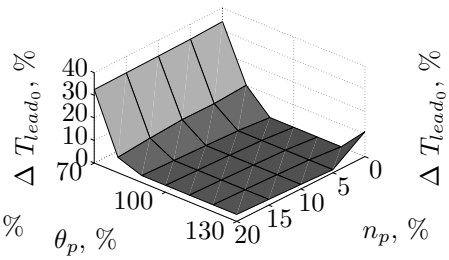
(c) Closed-loop compl. system const. white noise.



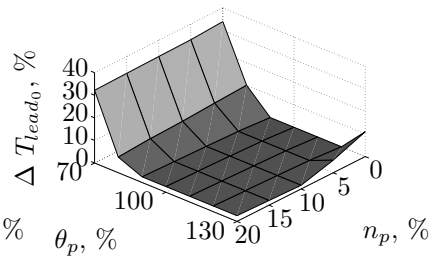
(d) Open-loop compl. system filtered noise.



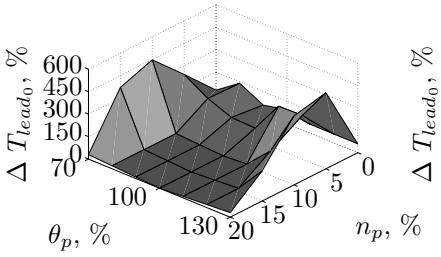
(e) Open-loop compl. system varying white noise.



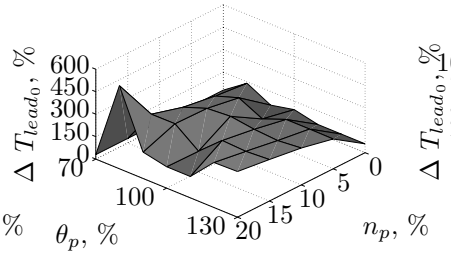
(f) Open-loop compl. system const. white noise.



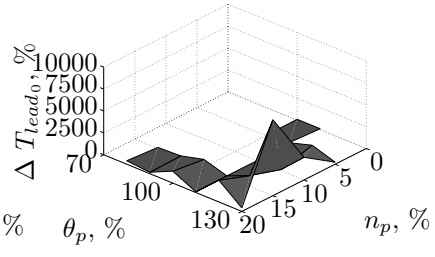
(g) Closed-loop Sys. I* filtered noise.



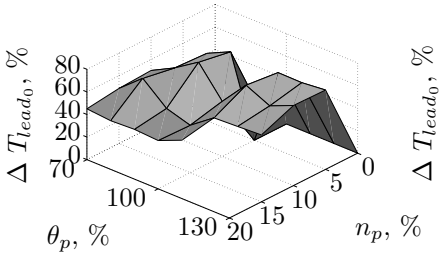
(h) Closed-loop Sys. I* varying white noise.



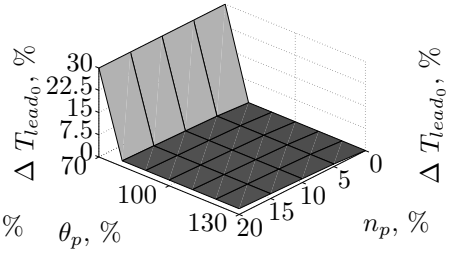
(i) Closed-loop Sys. I* const. white noise.



(j) Closed-loop Sys. II* filtered noise.



(k) Closed-loop Sys. II* varying white noise.



(l) Closed-loop Sys. II* const. white noise.

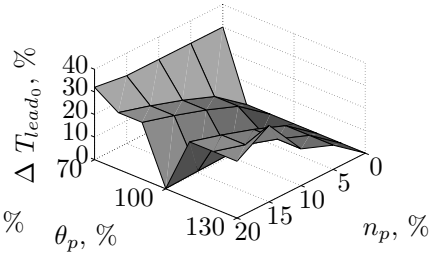


Figure 9-5: Percentual parameter difference $\Delta\theta$ of T_{lead_0} performed on twelve packages of different simulation data according to Table 9-1 plotted for a range of parameter variance percentages θ_p and remnant variance percentages n_p.

Table 9-2: System I* & II* parameter values.

Parameter	System I*	System II*
K_{v_0}	0.70 -	0.70 -
K_{v_1}	0.40 -	0.40 -
K_{v_2}	122.88 s	122.88 s
K_{v_3}	-2.30 -	-2.30 -
T_{lead}	0.00 s	4.00 s
K_c	5.00 -	6.00 -
T_c	0.01 s	4.00 s
T_n	0.20 s	0.20 s
τ_v	0.20 s	0.20 s
ω_{nm}	10.00 rad/s^2	10.00 rad/s^2
ζ_{nm}	0.20 -	0.20 -

from the estimation results of the twelve sets of Table 9-1, the parameter estimation results for the lead time constant parameter T_{lead_0} are shown in Figure 9-5. When Figure 9-5a is compared with Figure 9-5b and Figure 9-5c, a clear difference can be seen between the filtered noise closed-loop system and the white noise systems: while the white noise systems show U-shaped plots in the direction of the n_p-axis, the filtered noise system results in a disordered plot. Filtering the noise thus has an adverse effect on parameter estimation. From the results in Section 9-2 it was expected that a higher n_p would have a significant influence on the estimation result. This is true, as for all plots (except for Figure 9-5g) a gradual increase in $\Delta\theta$ can be seen with an increase in n_p (with some exceptions at low values of θ_p). Unexpected, however, are the large differences that are introduced to the system by θ_p. In general, it seems that a θ_p between 90% and 110% gives the best results. Below and above these values, the percentual difference in parameter value $\Delta\theta$ rapidly increases towards very large values. When the open-loop system plots in Figure 9-5d, Figure 9-5e, and Figure 9-5f are compared to the closed-loop system plots, the open-loop systems show better results, especially for a θ_p above 120%. The results for System I* in Figure 9-5g, Figure 9-5h, and Figure 9-5i illustrate that $\Delta\theta$ becomes very large for the single-integrator dynamics. Because parameter estimation returned so-called 'not a number' (NaN) values–in this instance unrepresentably large values–in the case of constant white noise, the plot in Figure 9-5i is not complete. At the location where NaNs were found, empty spots are visible in this plot. This indicates that the constant white noise band has a contribution that is too big for the single-integrator dynamics. When the double-integrator dynamics are evaluated, Figure 9-5j shows that $\Delta\theta$ grows rapidly when θ_p is not 100%. This is even more clear from Figure 9-5l. For System II*, $\Delta\theta$ rapidly increases with an increasing n_p. The varying white noise results for System II* are unexpected: the estimation of the single-integrator dynamics in Figure 9-5h is much harder than the estimation of the double-integrator dynamics in Figure 9-5k. In addition, the plots of all parameters for closed-loop System II* with varying white noise are very similar to Figure 9-5k, while normally some differences can be seen between the parameters. This

indicates that for a varying white noise, parameters of the double-integrator dynamics are easily estimated above a θ_p of 80% even though an increasing n_p has a slight negative effect on the estimation. Although the influence of System I* on $\Delta\theta$ seems big in comparison to that of System II*, the contribution in the complete closed-loop and complete open-loop systems seems mostly to originate from System II* due to the similar shapes of the plots. Note that for System I* & II* parameter T_{lead_0} is a fixed, time-invariant value. However, when the plots of T_{lead_0} are compared to the results for the sigmoid constants of $K_v(t)$ of System I* & II*, no real difference can be found.

In addition to the twelve simulation data sets in Table 9-1, another eleven data sets are created. In each of these data sets one specific parameter is varied for θ_p, whereas the other parameters are held at their true values ($\theta_p = 100\%$) as explained in Section 9-1. From the parameter estimation results of these data sets it becomes clear that the variation of a single parameter with θ_p has an effect on all parameters. This indicates a correlation between the parameters. Except for the individual θ_p variation data sets ofT_{lead_1}, T_{lead_2}, and ω_{nm}, all data sets show similar results in terms of $\Delta\theta$. As the results for the T_{lead_1} data set can be attributed to giving T_{lead_1} a value of 0.01 instead of zero (as explained in Section 9-1), the only significant effects on $\Delta\theta$ can be assigned to the variation of θ_p for T_{lead_2} and ω_{nm}. Considering that the only effects for these two parameters are observed at θ_ps of 70% and 130%, it can be concluded that within a range of 80% to 120% for θ_p these two parameters show estimation results similar to the other parameters. This means that within this range no single parameter has a significantly larger contribution to the parameter estimation compared to other parameters. Simultaneously, it can be concluded that an increase in human operator variance percentage n_p results in a similar increase in $\Delta\theta$ for each of the eleven data sets. This once again leads to the conclusion that a higher amount of human operator remnant has a negative effect on the parameter estimation results in general.

9-4 Local Minima

As mentioned in Section 3-3, the nonlinear estimation problem under consideration in this study yields a high number of local minima. When parameter estimation is performed, the GN algorithm repeatedly fails to find the global minimum, because it ends up at a local minimum. To show that local minima form a problem, the negative log-likelihood and a comparison of remnant n with prediction error $\epsilon(k)$ are plotted in Figure 9-6. Figure 9-6a, Figure 9-6c, and Figure 9-6e show the results of parameter estimation, done with a θ_p of 110%, for the closed-loop control task as defined in Section 8 with a n_p of 10%. The negative log-likelihood in Figure 9-6a steadily decreases to a value of 5905. The first and last iteration of the GN algorithm are plotted in Figure 9-6c and Figure 9-6e. The difference in prediction error and remnant at the first iteration is neatly reduced at the seventh iteration, indicating that there is a high probability that the global minimum is found. The small differences that are still visible in Figure 9-6e between the prediction error

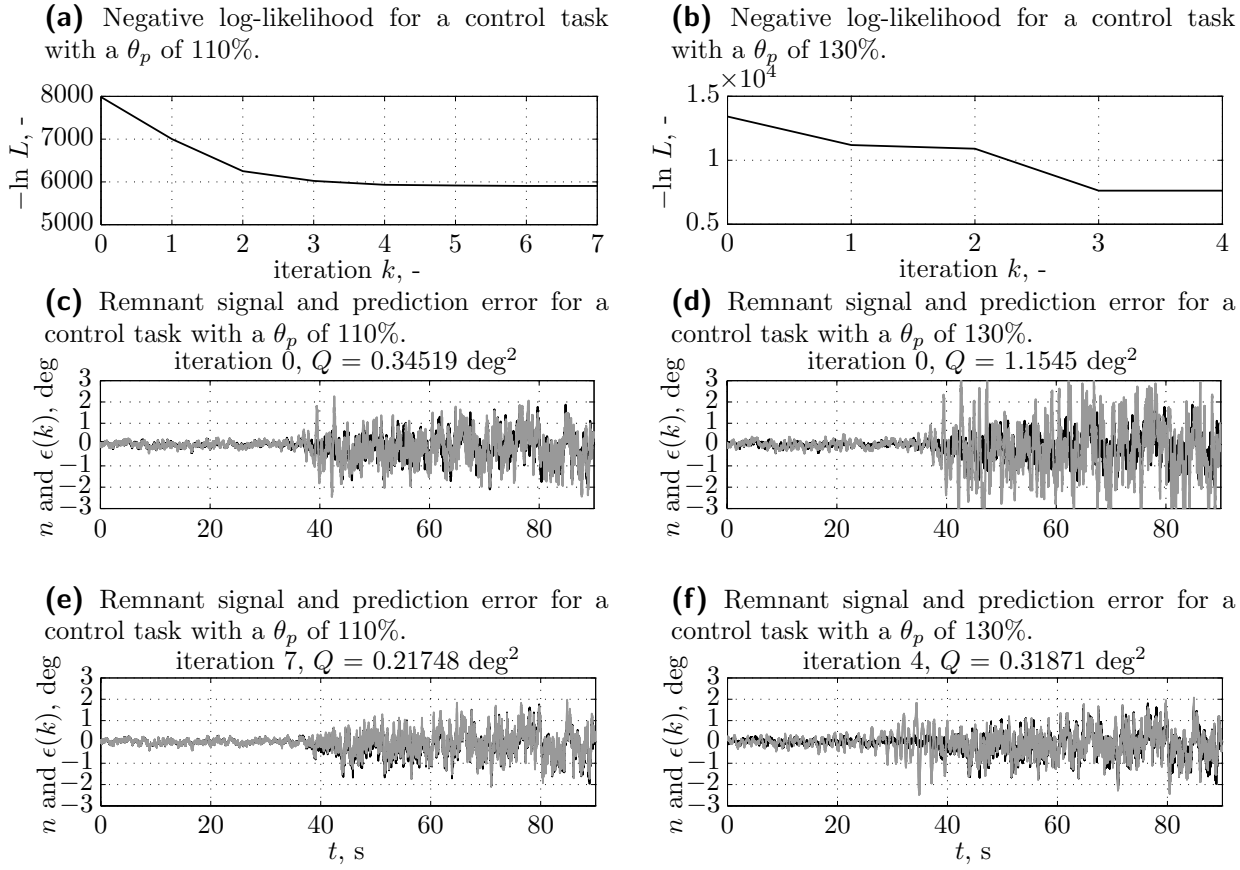


Figure 9-6: Negative log-likelihood and prediction error results of two parameter estimation runs with a n_p of 10% on simulated data of a closed-loop control task with θ_ps of respectively 110% and 130%.

and the remnant are caused by minor differences in the simulation of the data in Simulink® and the parameter estimation in MATLAB® as explained in Section 8-3. In Figure 9-6b, Figure 9-6d, and Figure 9-6f the results of parameter estimation, done with a n_p of 10%, on data of a closed-loop control task with a θ_p of 130% are shown. The negative log-likelihood for this parameter estimation, plotted in Figure 9-6b, erratically reduces to a value of 7625. As this is the same closed-loop control task data set, the negative log-likelihood is expected to drop to a value of 5905 as well. The final value of 7625 indicates that the GN algorithm converged to a local minimum. This is also visible from Figure 9-6d and Figure 9-6f, as at iteration 4 the prediction error still depicts a distinct difference from the remnant signal.

To understand how the local minima problem relates to the parameter variance percentage θ_p and the remnant variance percentage n_p, the negative log-likelihood values of the final GN iterations for the closed-loop control task are averaged for the three noise realizations that were used before in Section 9-3. The results are plotted in Figure 9-7. Each of the lines represents a system with a certain amount of remnant n_p. Optimal parameter estimation for each system would result in convergence to the global minimum. From these plots it

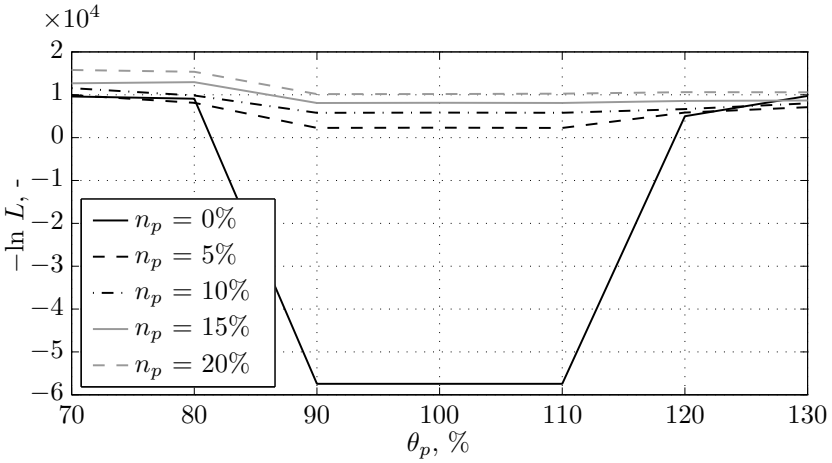


Figure 9-7: The negative log-likelihood for different parameter variance percentages θ_p and different remnant variance percentages n_p.

becomes clear that the global minimum is usually found when θ_p has a value between 90% and 110%. Below and above these values, the negative log-likelihood is higher than the value of the global minimum, which leads to the conclusion that the GN algorithm reduces to a local minimum. It thus seems that a clear relation exists between the initial parameter values and finding the global minimum. When the influence of the remnant is evaluated, it becomes clear that an increasing remnant makes the U-shaped plots flatter, indicating that the difference between the global minimum and local minima becomes smaller. This is especially true for higher values of θ_p. Note that there is a big difference between the system without remnant ($n_p = 0\%$) and the system with a remnant variance percentage n_p of 5%. This suggests that for systems with remnant variance percentages below 5% the difference between the global minimum and the local minima rapidly reduces.

In Figure 9-8 similar results are shown for the estimation in which only one initial parameter is varied with θ_p, whereas all other initial parameter values are held at their true values ($\theta_p = 100\%$). All parameters, except for T_{lead_2} and ω_{nm}, show near-constant lines for the negative log-likelihood as shown for K_{v_0} in Figure 9-8a (plotted for all parameters in Appendix D). This indicates that if only one of these parameters is varied, the probability that the global minimum is found is likely, even at high and low parameter variance percentages. A cause for this might be that ten out of eleven parameters start with the true value as their initial value. In Figure 9-8b and Figure 9-8c, the results for T_{lead_2} and ω_{nm} are shown. For T_{lead_2} local minima are found below a θ_p of 90% and above a θ_p of 120%. For ω_{nm} local minima are found below a value of 80% for θ_p. A higher n_p especially has positive effects for ω_{nm} as its lines become flatter with higher remnant variance percentages. These results for T_{lead_2} and ω_{nm} are consistent with the results in Section 9-3, where T_{lead_2} and ω_{nm} also showed to have detrimental effects on the parameter estimation results. Note, that in Section 9-3 the boundaries, in which parameter estimation still seems to return proper results, are identified to range from for 80% to 120% for θ_p. Here, these limits seem

to be even smaller with values of 90% and 120%. From these results it can be concluded that for the estimation problem considered in this study, the parameters T_{lead_2} and ω_{nm} contribute in a negative way to the local minima problem, especially for parameter variance percentages that are further removed from the true values at 100%.

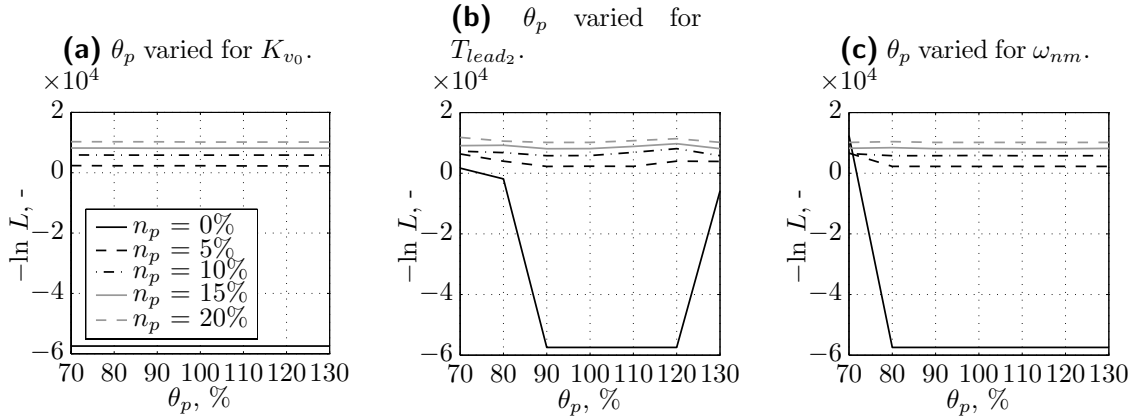


Figure 9-8: The negative log-likelihood for different parameter variance percentages θ_p and different remnant variance percentages n_p. For the estimations in the figures only the respective parameter is varied for θ_p, all other parameters are held at a θ_p of 100%.

Chapter 10

Discussion

Parameter estimation with a proposed time-varying time-domain MLE identification procedure was done on simulated data of a closed-loop manual control task. The results of the proposed MLE method were compared to the results of the sliding time window MLE method proposed by Zaal & Sweet [2011]. A comparison of the respective VAF values of the methods leads to the conclusion that the proposed method performs better than the sliding time window method. Because the MLE method proposed in this thesis uses Boltzmann sigmoids to model the time-varying dynamics, the estimation results in a much smoother signal for the estimated time-varying parameters in comparison to the sliding time window MLE method, which often returns rapidly changing results. When regarding computational effort, the proposed method is much faster than the sliding time window method.

The smooth shape of the Boltzmann sigmoids also has its drawbacks in the sense that the Boltzmann sigmoid must be rather steep for the sigmoid to start at the same values of the single-integrator at 20.96 seconds and to end at the double-integrator dynamics values at 60.96 seconds. Although the entire Boltzmann sigmoid is modeled over 40 seconds, a rapid change occurs in a very short period of time. This is not a big problem, as it is expected that a human operator adapts his or her control behavior rapidly from one set of dynamics to another set. However, if a more gradual transition between the dynamics needs to be modeled, the Boltzmann sigmoid might not be suitable. Furthermore, the choice of a Boltzmann sigmoid imposes restrictions on the shape of the transition between different sets of dynamics, if parameter estimation on polynomials would pose less problems, this might be a better choice due to the innate freedom of a polynomial's shape. Finally, the Boltzmann sigmoid introduces four parameters to the parameter estimation problem for each time-varying parameter that needs to be estimated. It thus rapidly increases the complexity of the estimation problem.

The results of the estimation problem are heavily influenced by the human operator

remnant, by the choice of the initial parameter values, and by local minima. Due to the nonlinear behavior of the human operator, the addition of a remnant is unavoidable to model a realistic control task. It is therefore difficult to reduce the negative contribution the remnant has on the proposed MLE method. It should be noted that the research in this study was done by giving all parameters an equal offset with θ_p. This was done to be able to easily compare the results to each other. It might be useful to study what the effects are of taking randomized values of the initial parameter values. The problem related to the initial parameter values can be reduced by, for example, using a genetic algorithm as proposed in [Zaal et al., 2009]. A major drawback of such a genetic algorithm is the amount of computational time needed and the fact that there is no guarantee that the result reduces to initial parameter values that enhance the parameter estimation results. A clear correlation was found between the initial parameter values and finding the global minimum. Increasing the chance of finding acceptable initial parameter values thus decreases the local minima problem. The local minima problem is also related to the gradient-based GN output error method used in this study. Other methods such as convexification [Ljung, 2010], a random search, interval analysis [Kampen, 2010], or simulated annealing could help resolve the local minima problem. The results in this study clearly show that the presence of expected levels of human operator remnant signal has a severe negative effect on the parameter estimation results. Therefore, the estimation method needs to be adjusted to be able to cope with these remnant signal levels. For future research it is thus recommended to study a filter error method [Raol et al., 2004] which combines the output error method with a Kalman filter. Although a filter error method is mathematical more complex and computationally intensive, it seems to be a promising method for solving the local minima problem encountered in this study. The filter error method can also help with estimating part of the remnant, which might result in better parameter estimates.

Finally, simulated data was used for the reason that it is easy to know what the true values–and thus the optimal parameter estimates–of the original dynamic system are. Future research should also focus on using experimental data to understand the effect of the proposed MLE method on the difference between a real-life and a simulated system.

Chapter 11

Conclusions

In this thesis a Maximum Likelihood Estimation method is proposed for the estimation of linear time-varying pilot-vehicle system parameters in the time-domain. The parameter estimation is performed on simulated data of a closed-loop manual control task, which contains time-varying human operator dynamics as well as time-varying vehicle dynamics. The time-varying system behavior is modeled utilizing Boltzmann sigmoids. The proposed method is compared to a sliding time window Maximum Likelihood Estimation method. Although the proposed method performs better than the sliding time window method, it is heavily influenced by the human operator remnant, by the choice of the initial parameter values, and by local minima. Because improvements can be made for each of these influential factors, future research is recommended.

Part IV

Back Matter

Appendix A

Orthogonal Polynomial Overview

Orthogonal functions are often used in LTV and nonlinear parameter estimation problems. A subclass of these orthogonal functions are the *orthogonal polynomials*, which have traditionally been divided into *classical* and *nonclassical* orthogonal polynomials. Especially for nonlinear system identification problems the product property of the polynomials is advantageous [Rémond et al., 2008]. Some frequently used classical orthogonal polynomials are the Hermite, Laguerre, and Jacobi polynomials. Gegenbauer polynomials, a special subclass of the Jacobi polynomials, include Legendre and Chebyshev polynomials. This classification of polynomials was deemed narrow-minded by [Andrews & Askey, 1985] and resulted in the development of the so-called *Askey Scheme*. The extended scheme in Figure A-1 gives an overview of the relations between different groups of hypergeometric orthogonal polynomials. This extended scheme was taken from [Koekoek et al., 2010] with written permission from the author and only shows the set of classical orthogonal polynomials, which are special or limiting cases of either the Askey-Wilson polynomials or the q-Racah polynomials. For the the q-Askey Scheme with the q-analogues included and an extensive explanation of the relations between the hypergeometric orthogonal polynomials, please refer to [Koekoek et al., 2010]. Figure A-1 shows different levels of which the notation will be clarified by the definition of hypergeometric functions. A hypergeometric function ${}_rF_s$ is defined by Equation (A-1).

$$ {}_rF_s(a_1,\ldots,a_r;b_1,\ldots,b_s;z) := \sum_{k=0}^{\infty} \frac{(a_1,\ldots,a_r)_k}{(b_1,\ldots,b_s)_k} \frac{z^k}{k!} \tag{A-1} $$

where

$$ (a_1,\ldots,a_r)_k := (a_1)_k \ldots (a_r)_k. \tag{A-2} $$

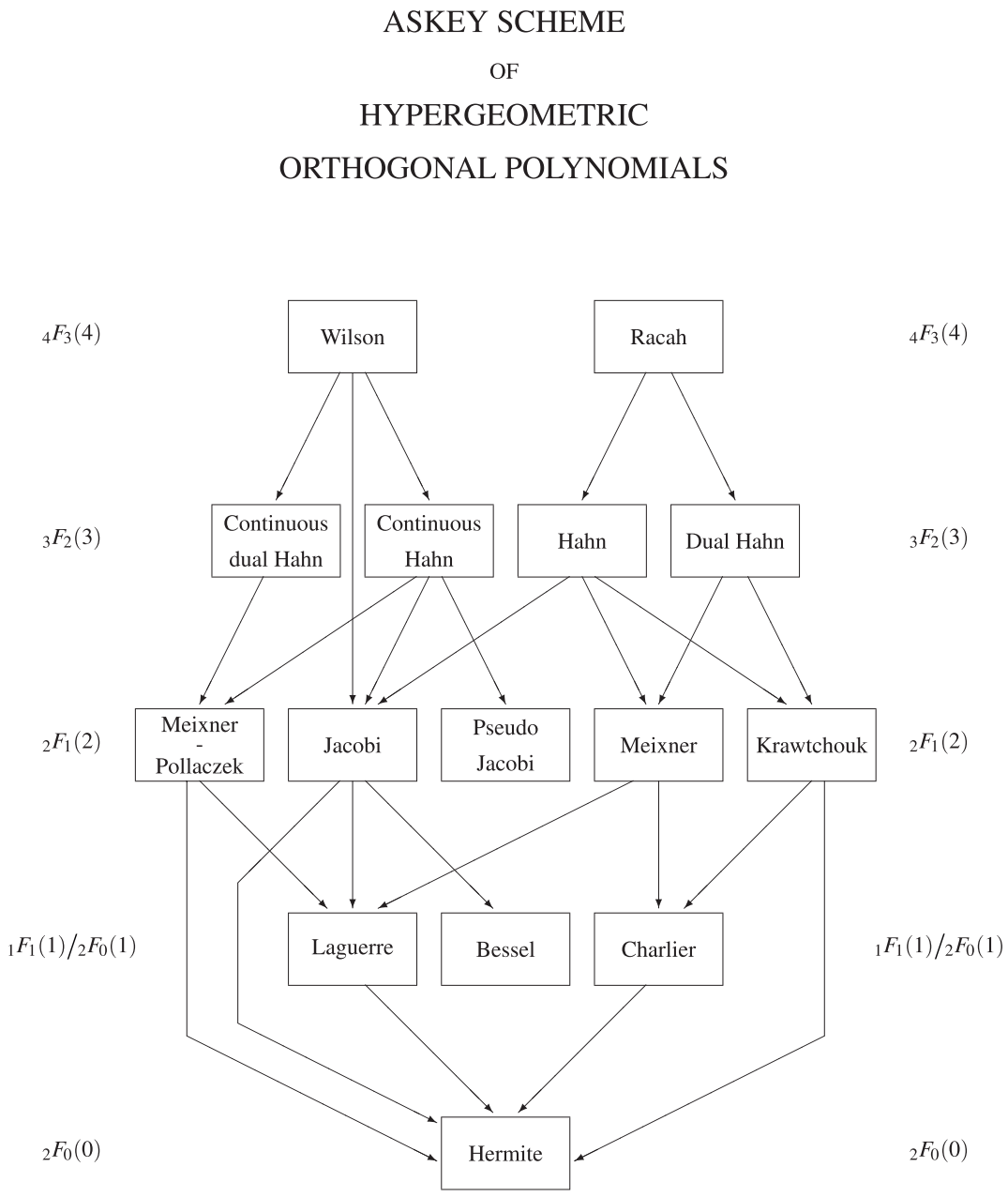


Figure A-1: Askey Scheme of Hypergeometric Orthogonal Polynomials. [Koekoek et al., 2010].

Appendix B

Simulink Models

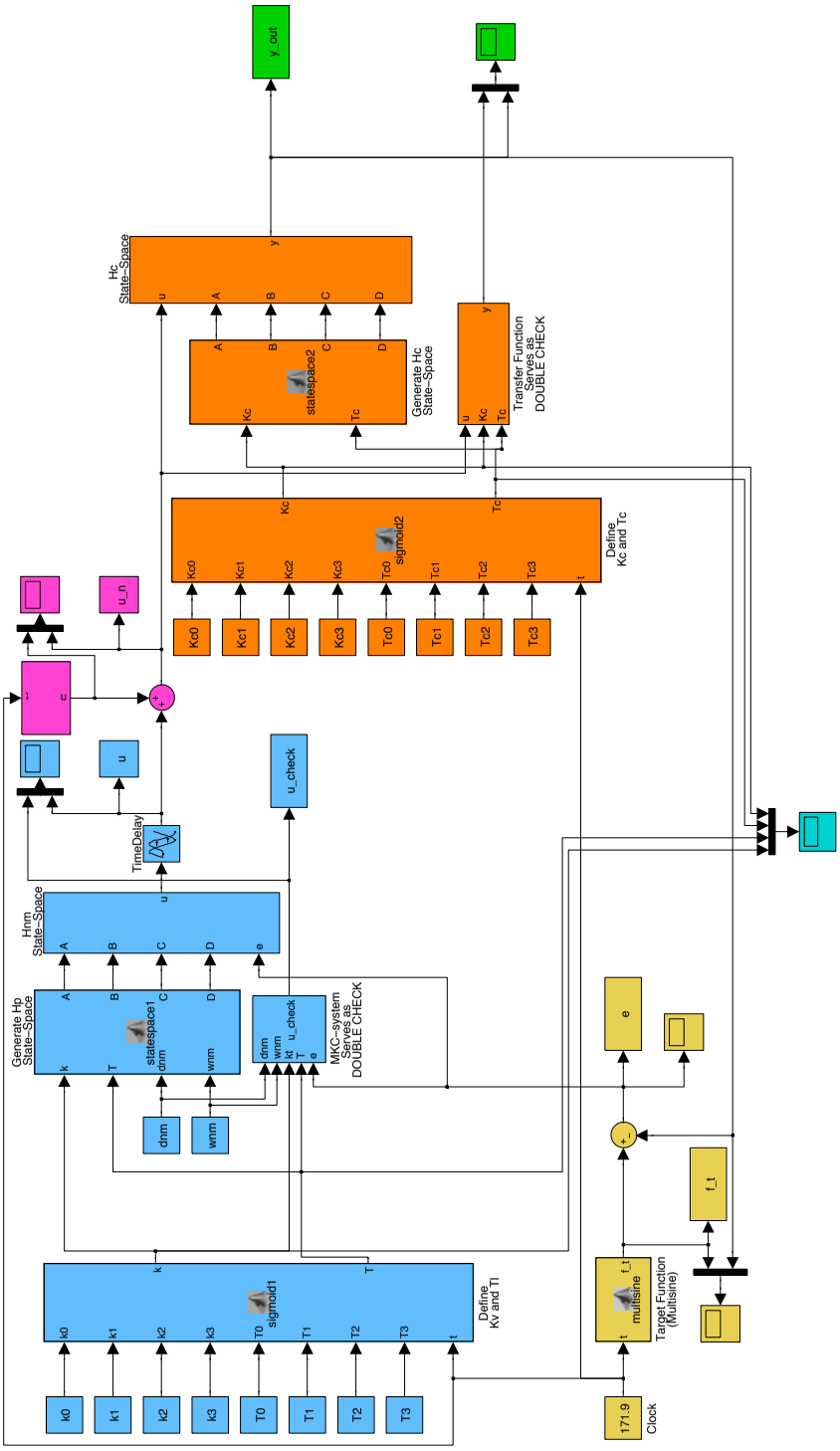


Figure B-1: The completed control system in Simulink® for the control task as mentioned in Chapter 8.

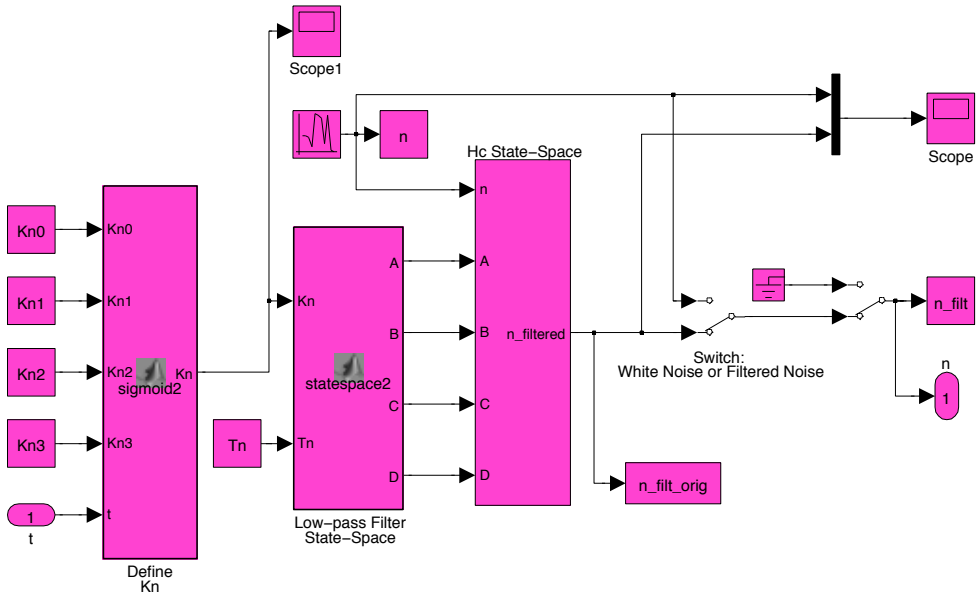


Figure B-2: The subsystem for the noise from Figure B-1 with a switch to choose for white noise or filtered noise with H_n.

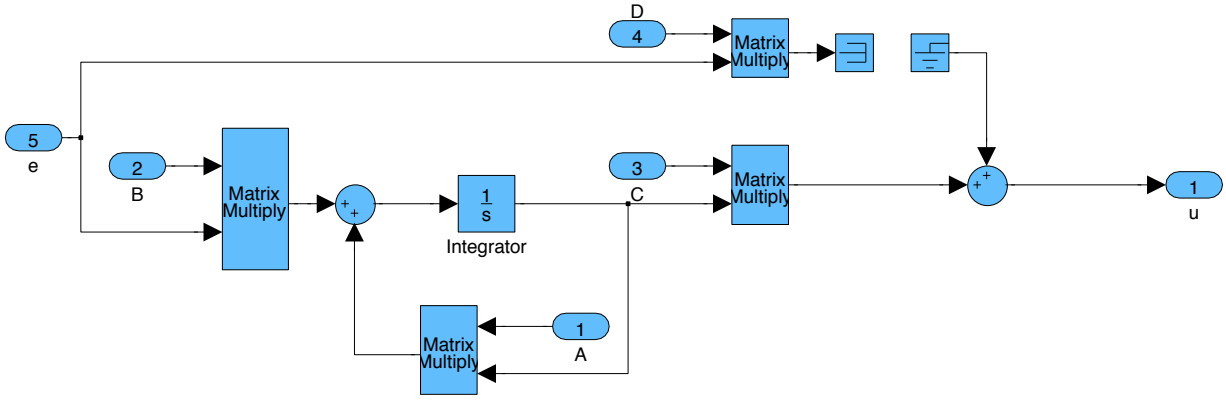


Figure B-3: The H_{nm} state-space subsystem from Figure B-1 generates control signal u from error signal e. The same system is used for H_n and for H_c.

Appendix C

Parameter-Remnant Plots

C-1 Closed-Loop Filtered Noise Complete System

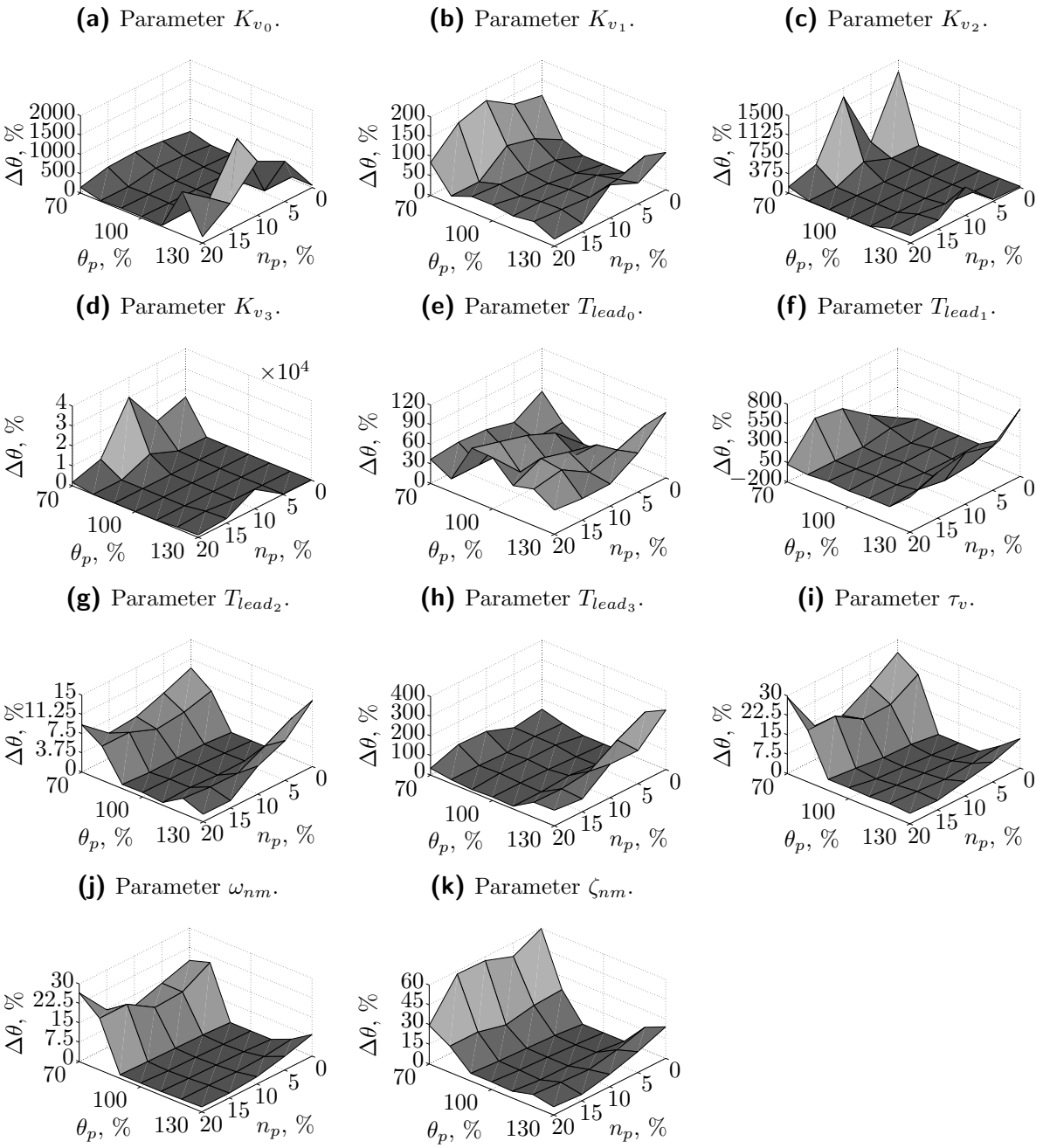


Figure C-1: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of the complete, closed-loop, filtered noise system.

C-2 Closed-Loop Sigm. White Noise Complete System

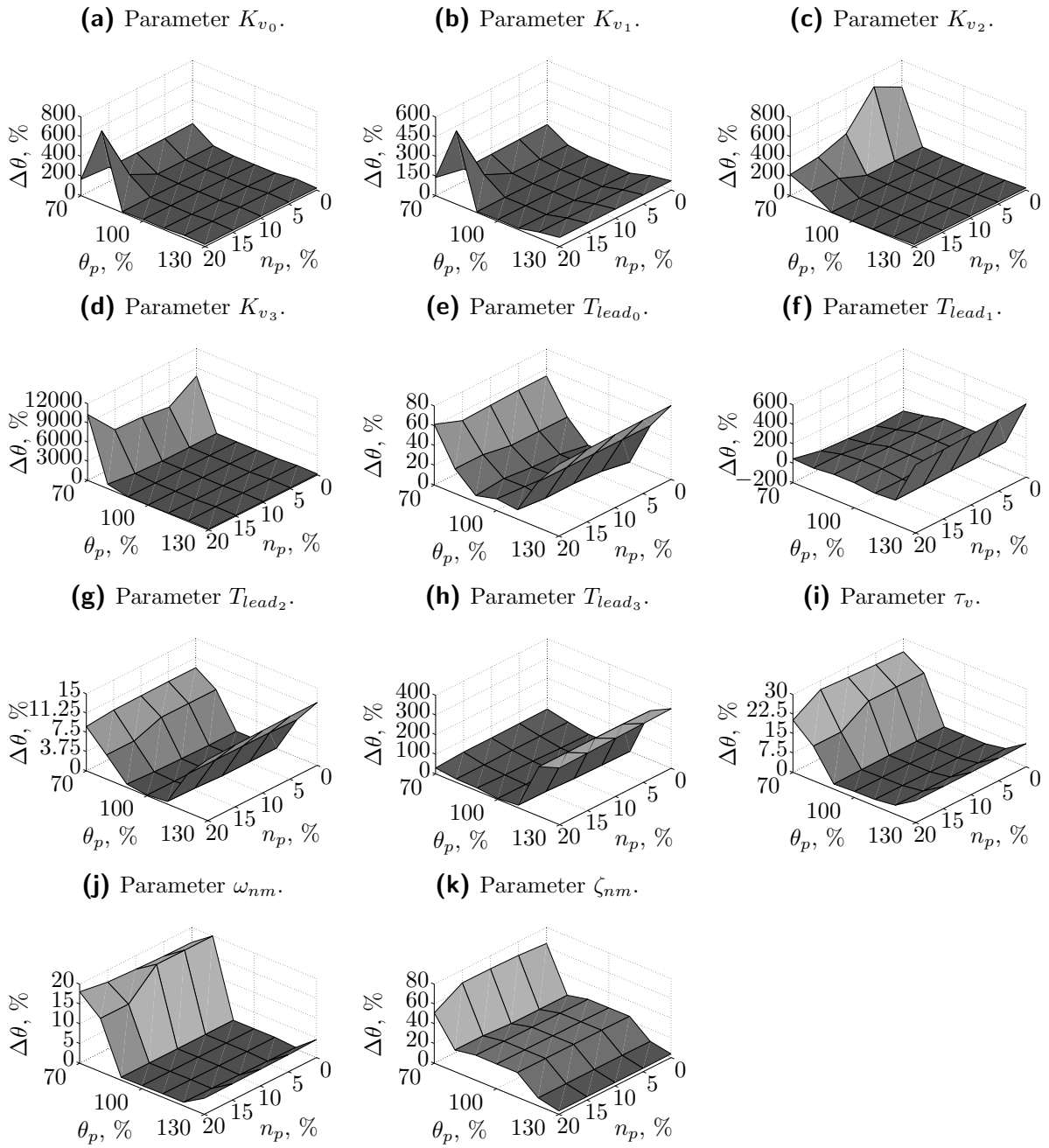


Figure C-2: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of the complete, closed-loop, sigmoidal white noise system.

C-3 Closed-Loop Const. White Noise Complete System

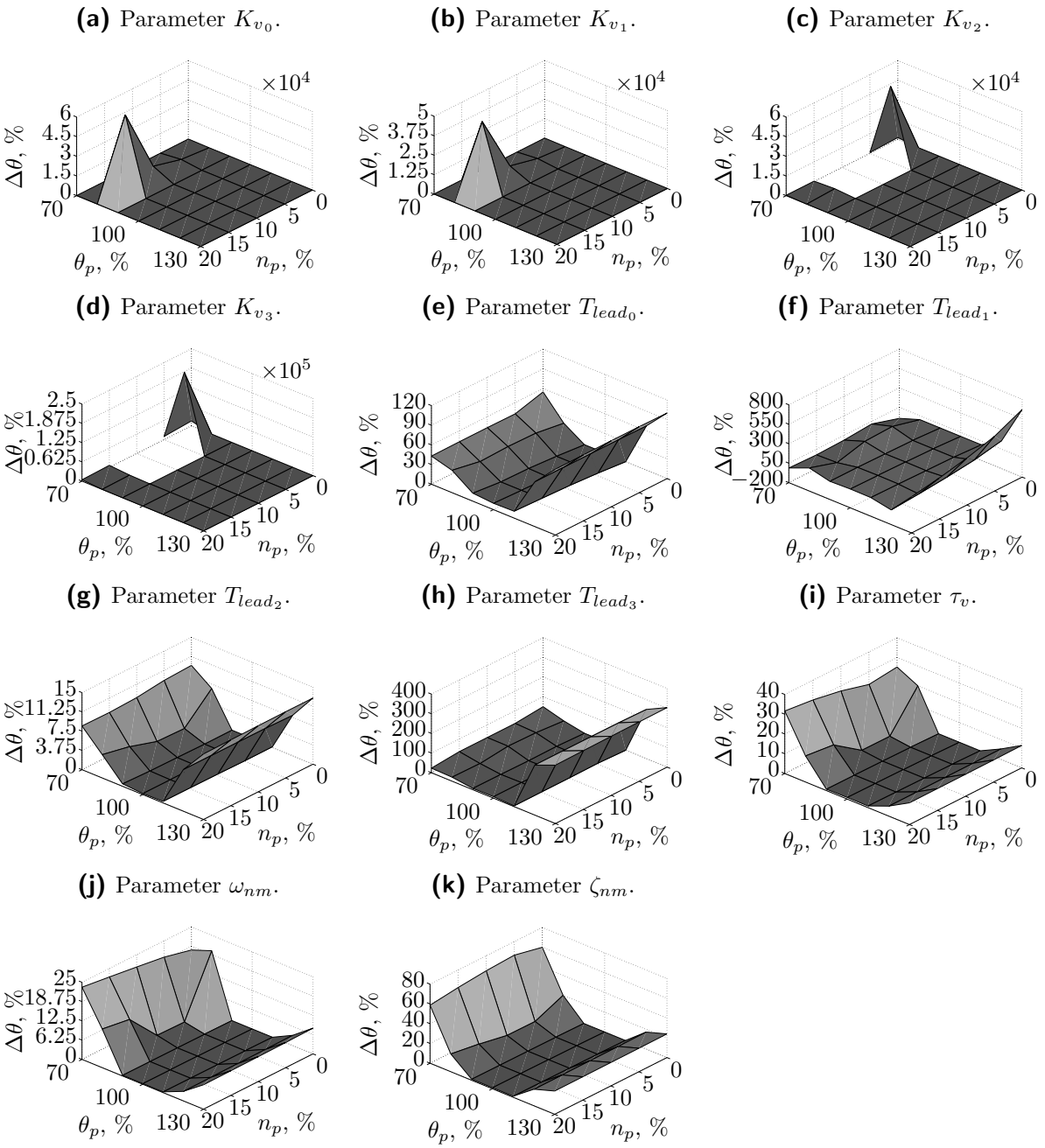


Figure C-3: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of the complete, closed-loop, constant white noise system.

C-4 Open-Loop Filtered Noise Complete System

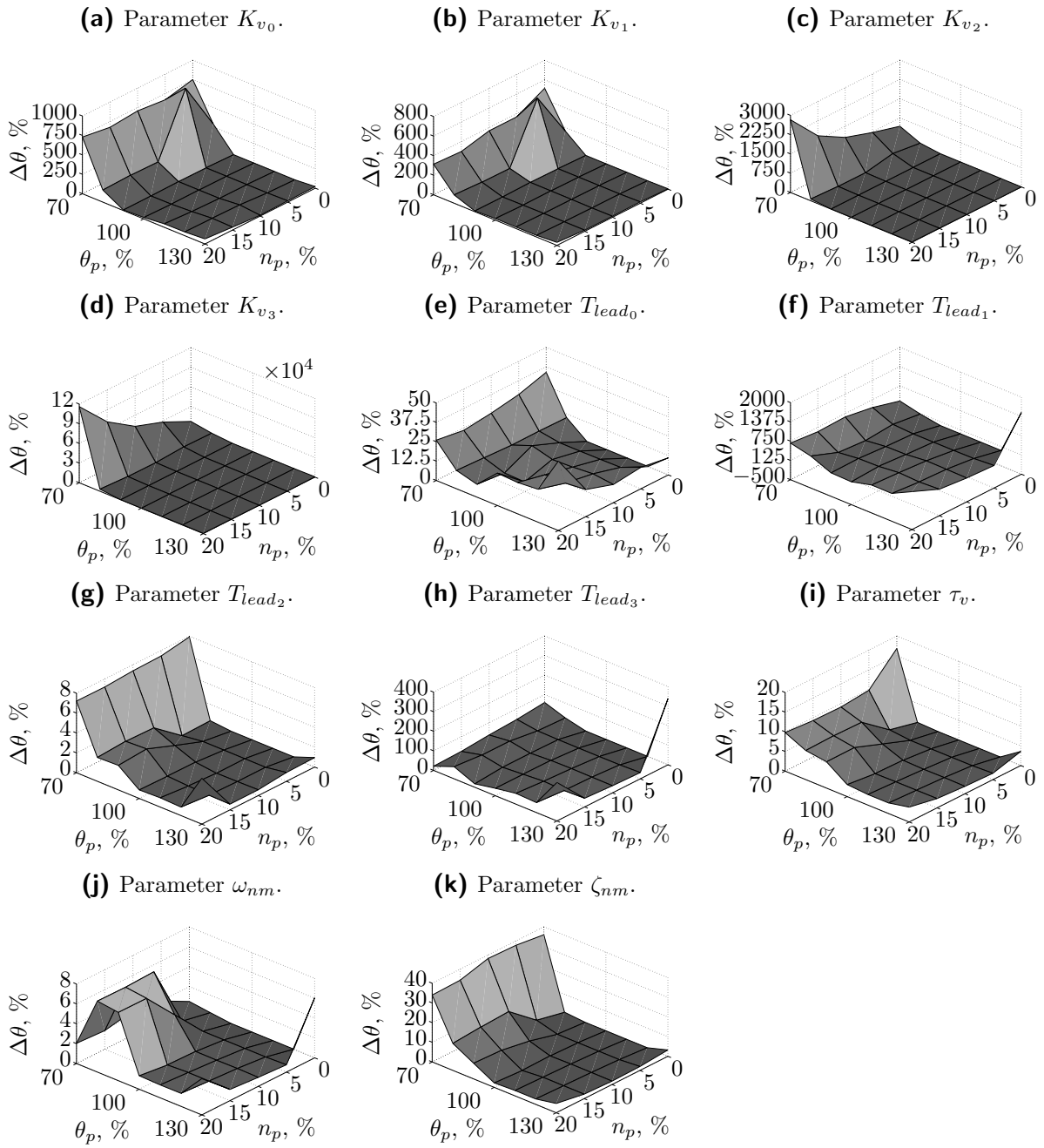


Figure C-4: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of the complete, open-loop, filtered noise system.

C-5 Open-Loop Sigm. White Noise Complete System

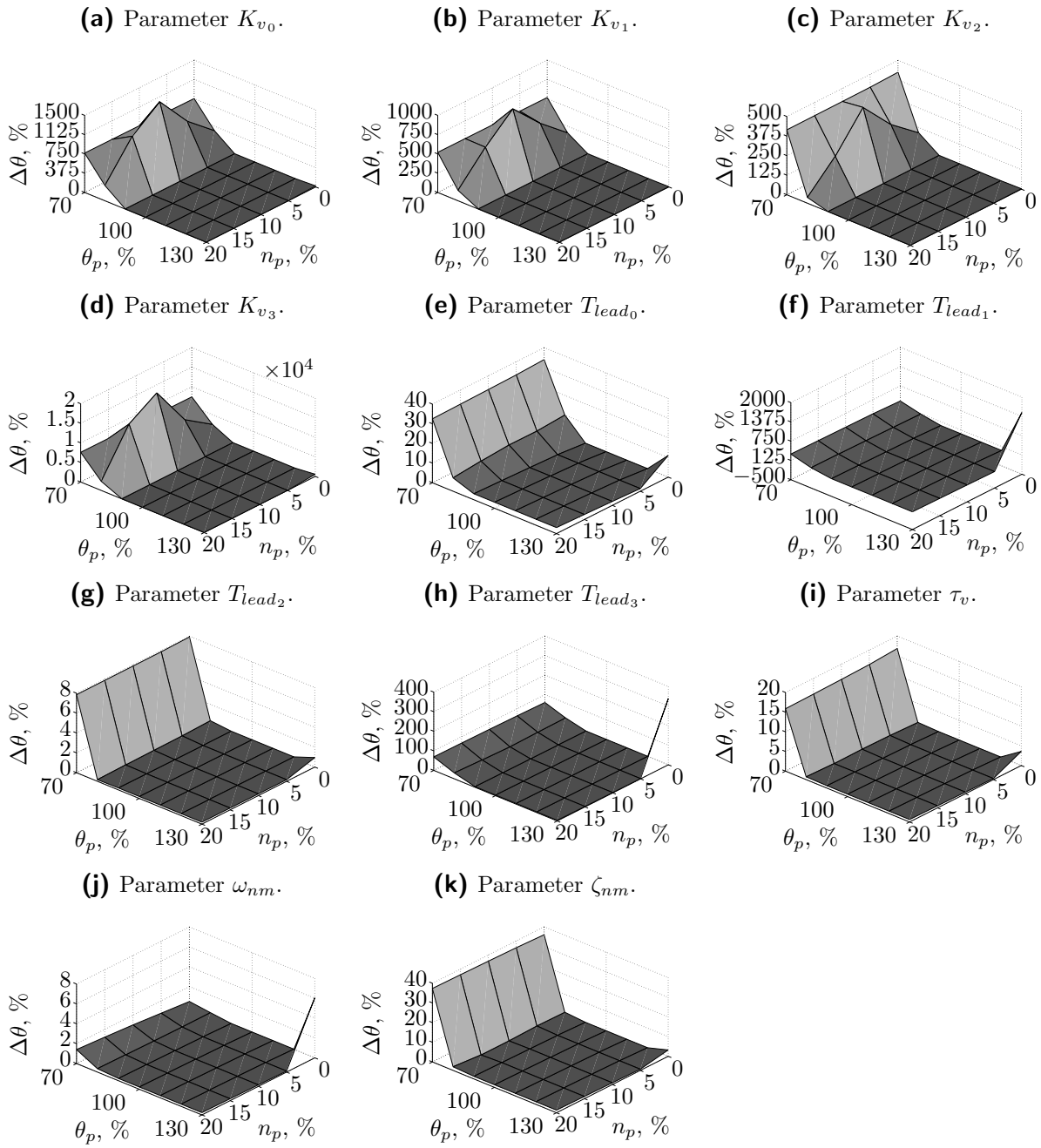


Figure C-5: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of the complete, open-loop, sigmoidal white noise system.

C-6 Open-Loop Const. White Noise Complete System

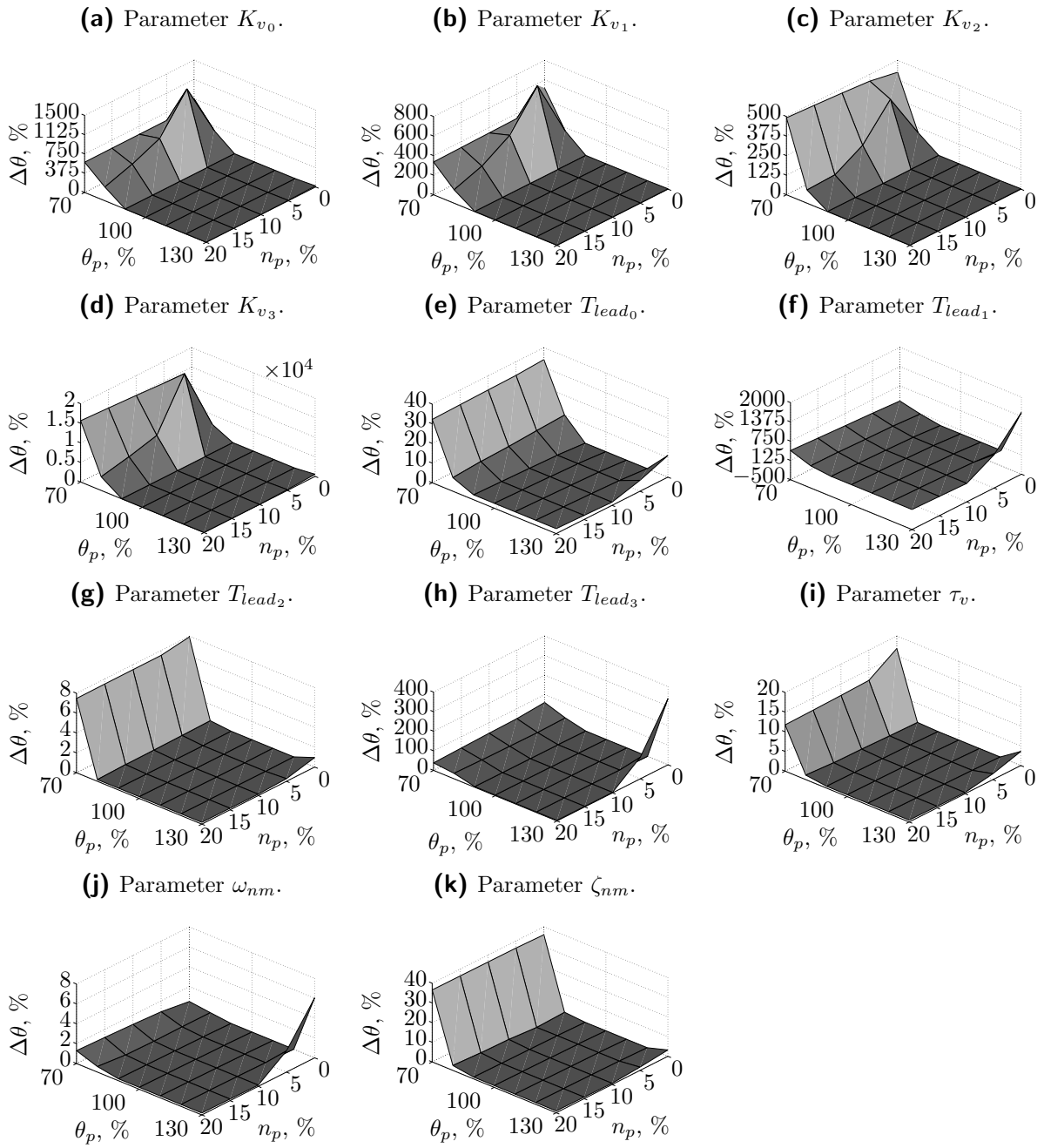


Figure C-6: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of the complete, open-loop, constant white noise system.

C-7 Closed-Loop Filtered Noise System I

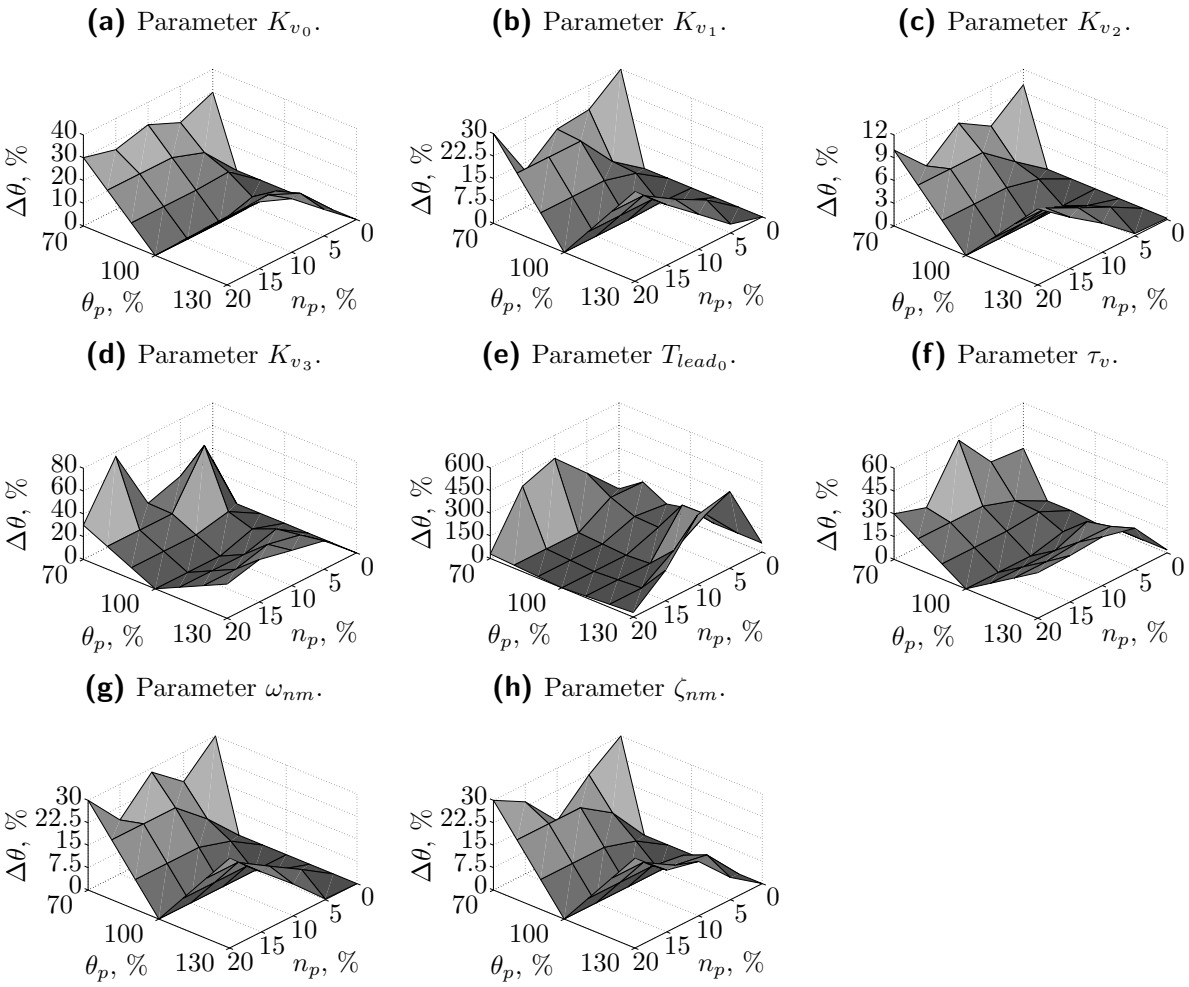


Figure C-7: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of complete, closed-loop, filtered noise System I.

C-8 Closed-Loop Sigm. White Noise System I

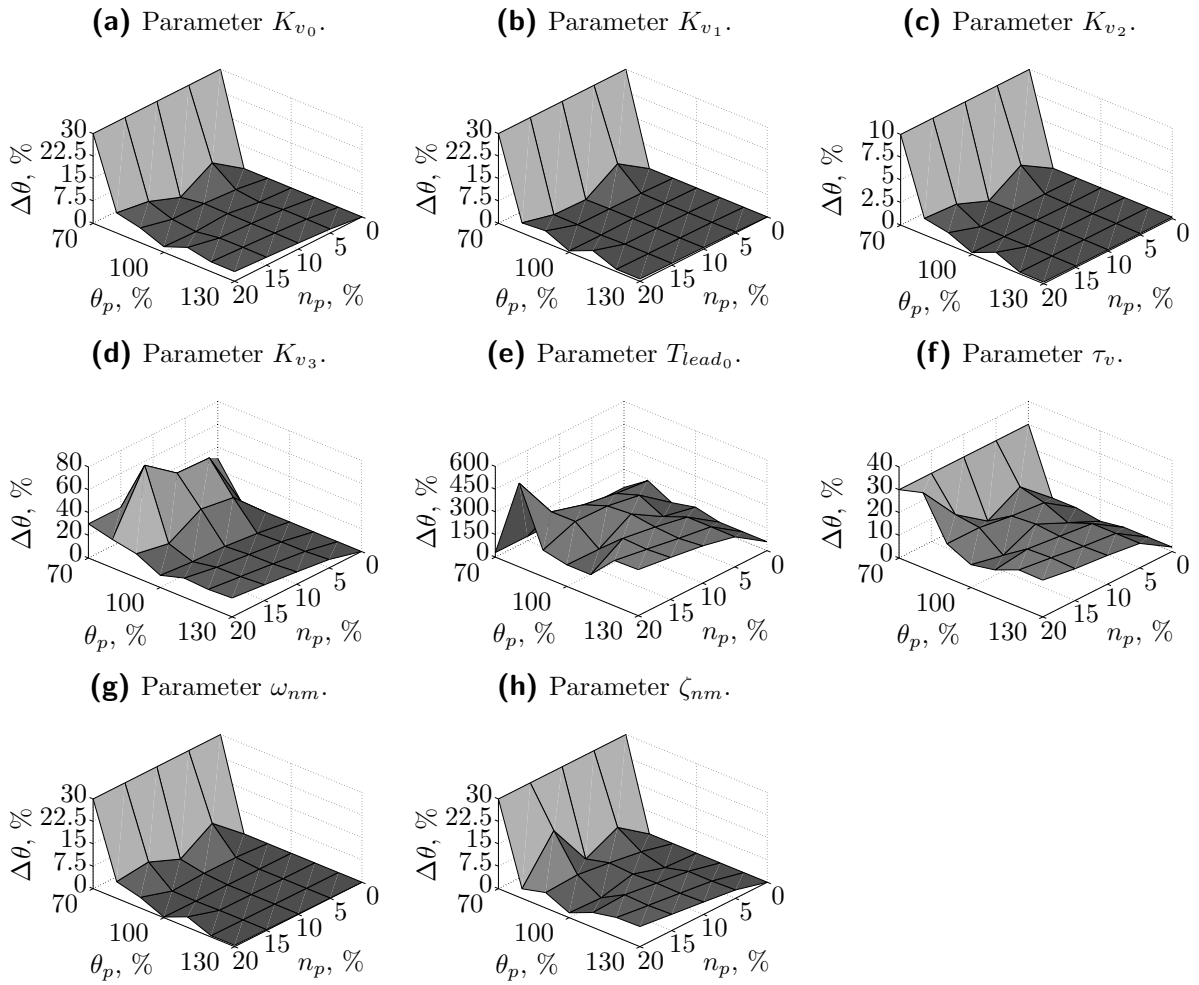
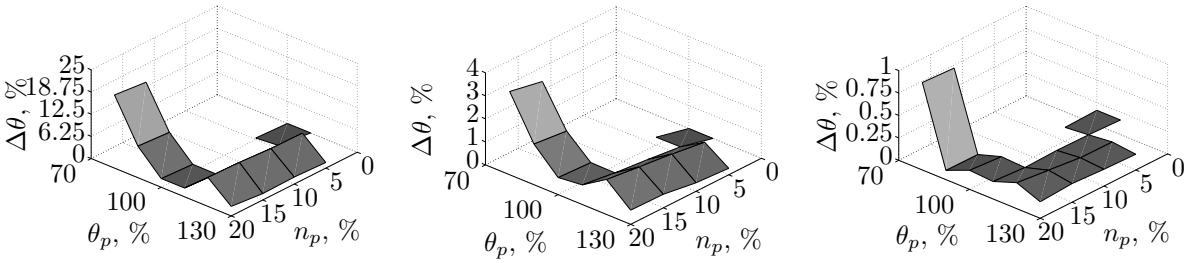


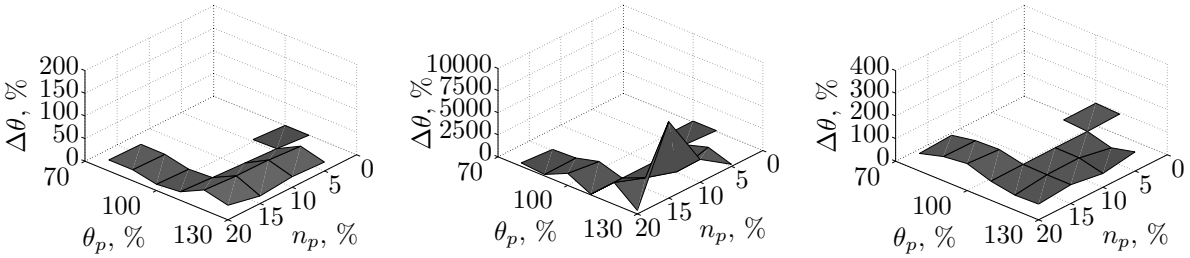
Figure C-8: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of closed-loop, sigmoidal white noise System I.

C-9 Closed-Loop Const. White Noise System I

(a) Parameter K_{v_0}. **(b)** Parameter K_{v_1}. **(c)** Parameter K_{v_2}.



(d) Parameter K_{v_3}. **(e)** Parameter T_{lead_0}. **(f)** Parameter τ_v.



(g) Parameter ω_{nm}. **(h)** Parameter ζ_{nm}.

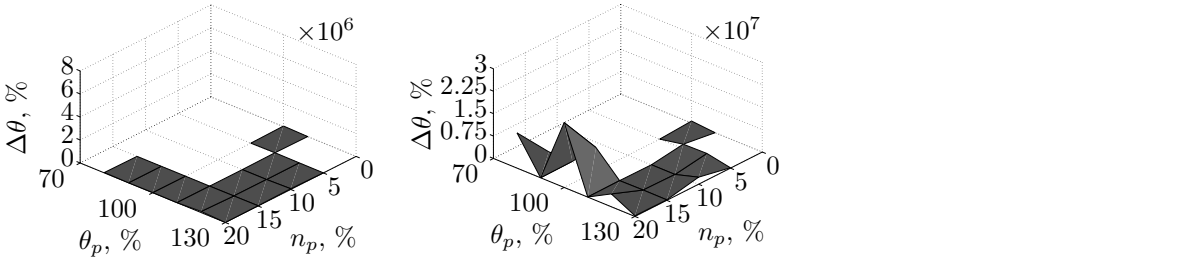


Figure C-9: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of closed-loop, constant white noise System I.

C-10 Closed-Loop Filtered Noise System II

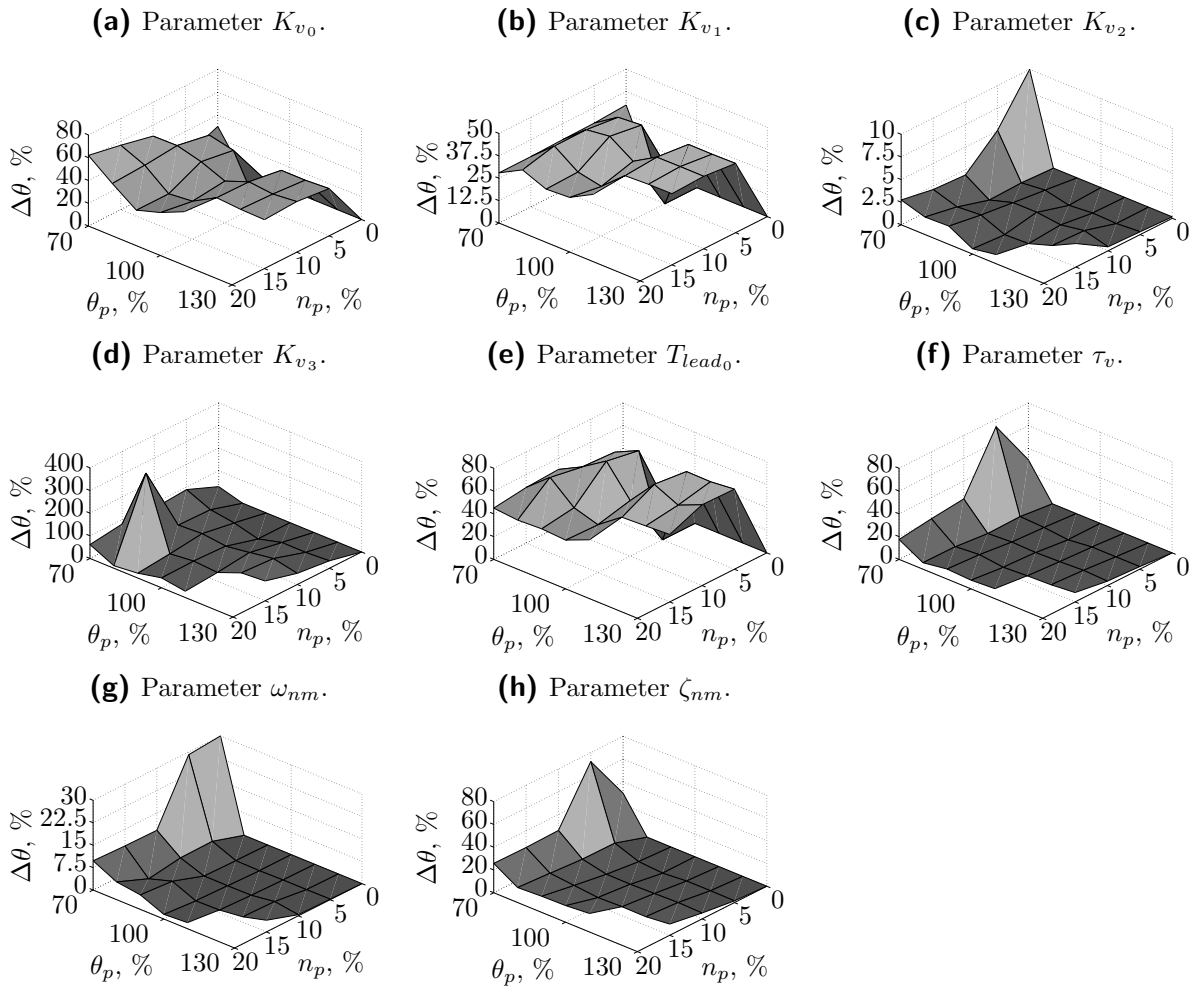


Figure C-10: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of closed-loop, filtered noise System II.

C-11 Closed-Loop Sigm. White Noise System II

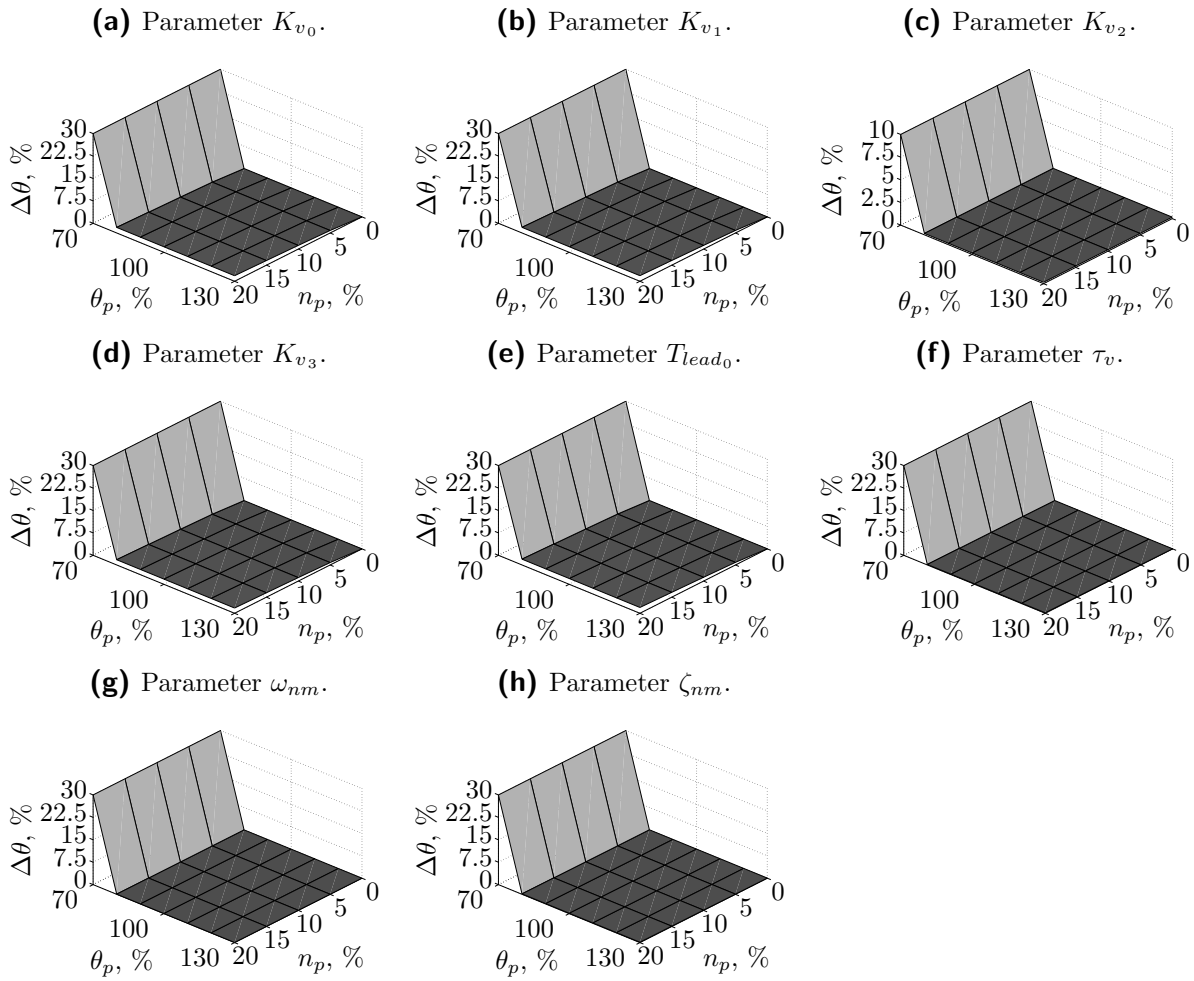


Figure C-11: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of closed-loop, sigmoidal white noise System II.

C-12 Closed-Loop Const. White Noise System II

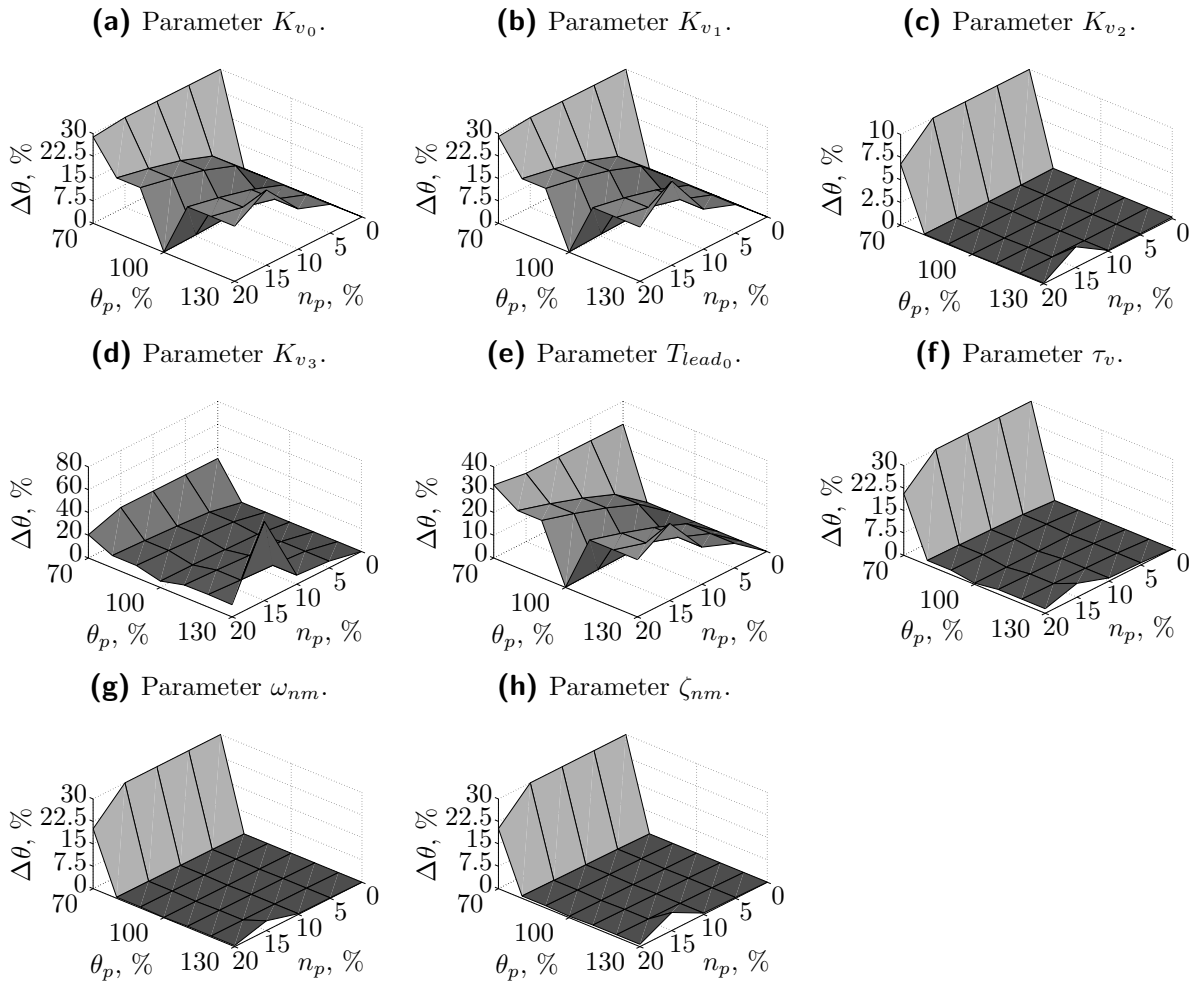


Figure C-12: Parameter difference $\Delta\Theta$ from true values θ_0 over a range of pilot remnant variance percentages n_p and a range of different initial parameter offsets θ_p. MLE executed on simulated data of closed-loop, constant white noise System II.

C-13 Varied K_{v_0} Closed-Loop Filtered Noise System.

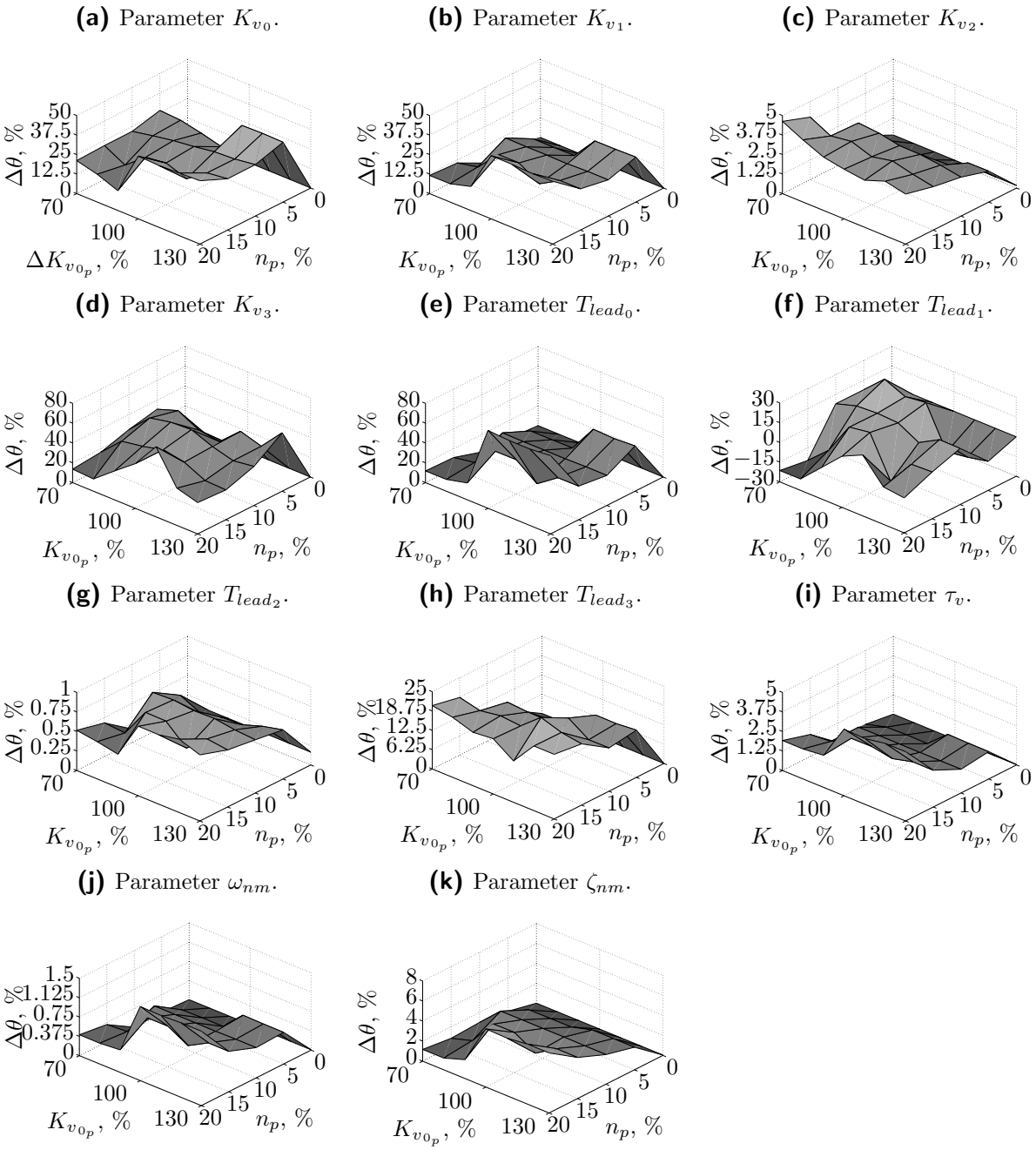


Figure C-13: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except K_{v_0} which is varied over a range of parameter offsets ΔK_{v_0} and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-14 Varied K_{v_1} Closed-Loop Filtered Noise System.

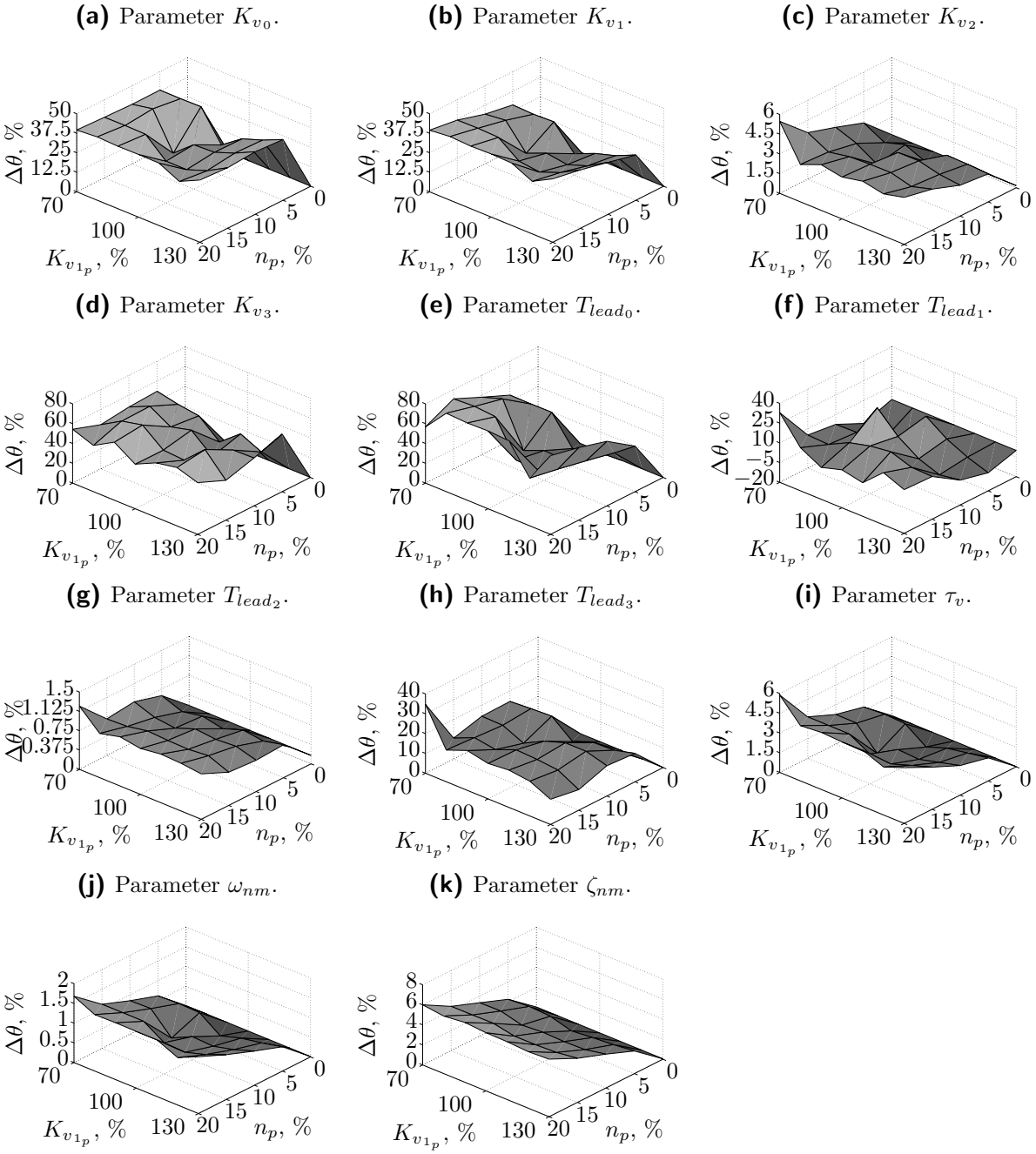


Figure C-14: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except K_{v_1} which is varied over a range of parameter offsets ΔK_{v_1} and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-15 Varied K_{v_2} Closed-Loop Filtered Noise System.

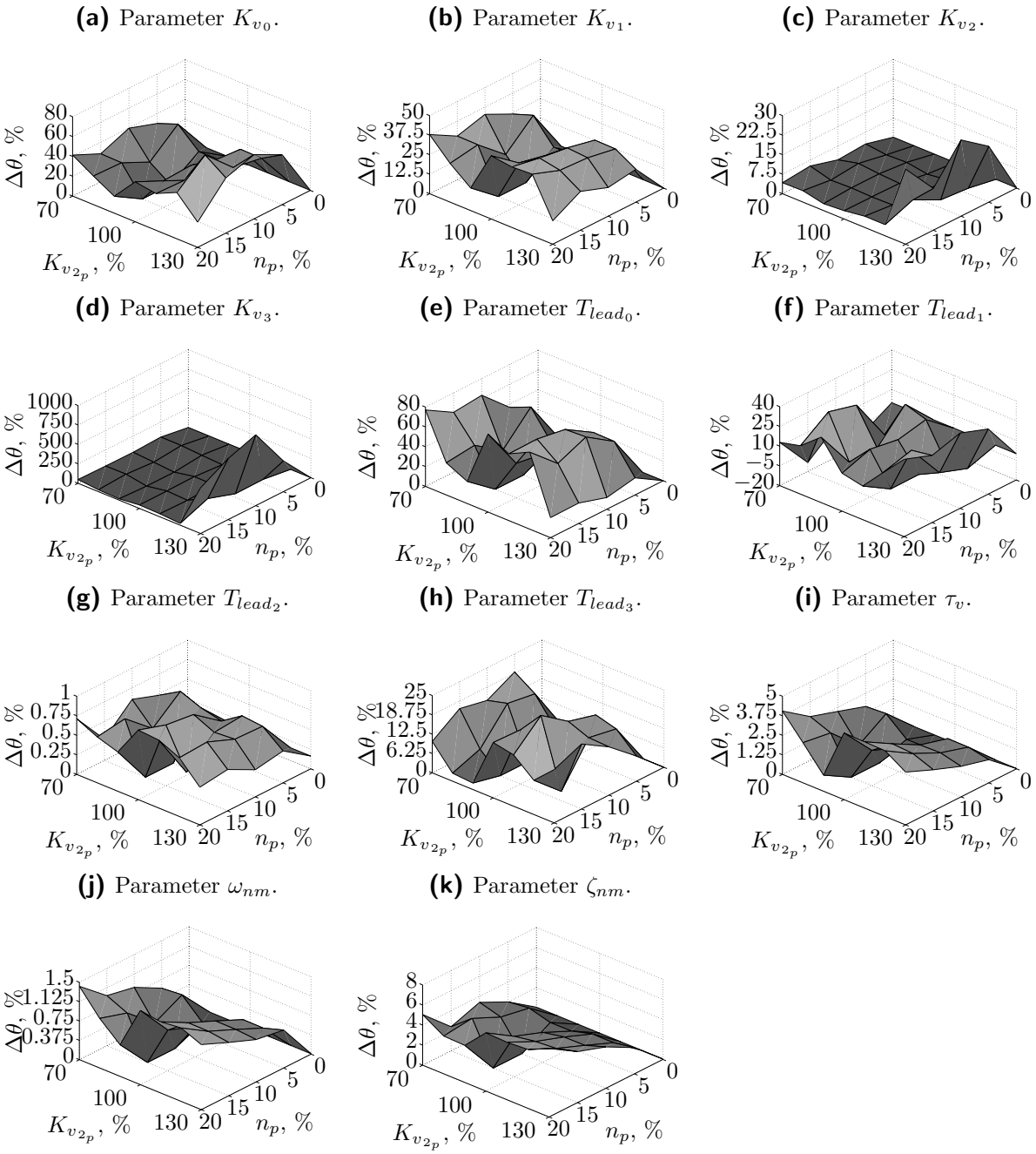


Figure C-15: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except K_{v_2} which is varied over a range of parameter offsets ΔK_{v_2} and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-16 Varied K_{v_3} Closed-Loop Filtered Noise System.

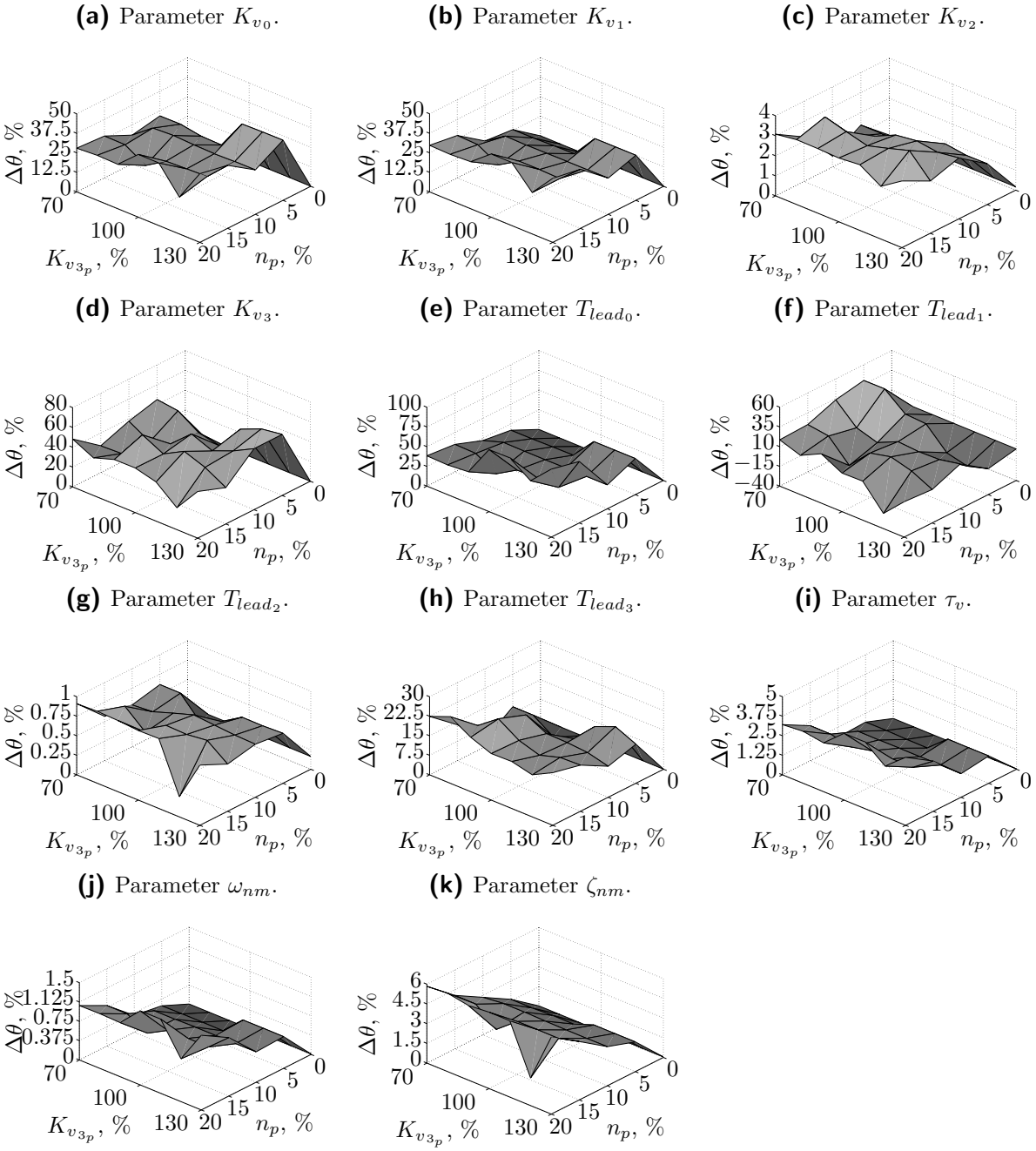
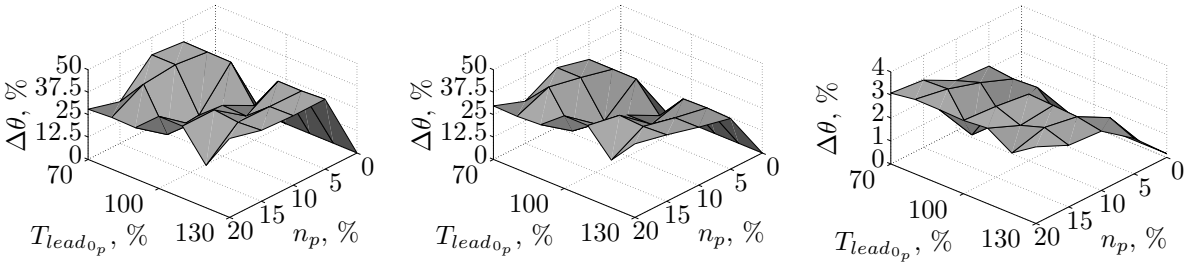


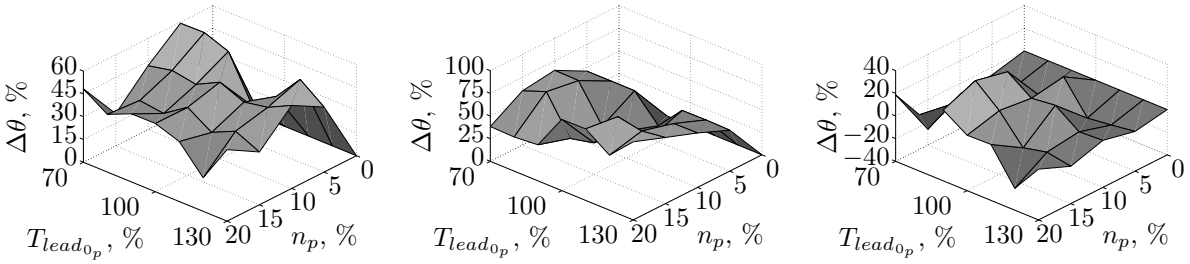
Figure C-16: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except K_{v_3} which is varied over a range of parameter offsets ΔK_{v_3} and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-17 Varied T_{lead_0} Closed-Loop Filtered Noise System.

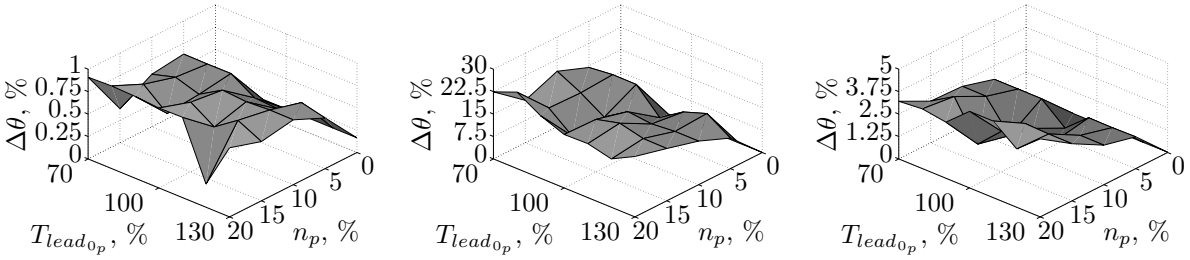
(a) Parameter K_{v_0}. **(b)** Parameter K_{v_1}. **(c)** Parameter K_{v_2}.



(d) Parameter K_{v_3}. **(e)** Parameter T_{lead_0}. **(f)** Parameter T_{lead_1}.



(g) Parameter T_{lead_2}. **(h)** Parameter T_{lead_3}. **(i)** Parameter τ_v.



(j) Parameter ω_{nm}. **(k)** Parameter ζ_{nm}.

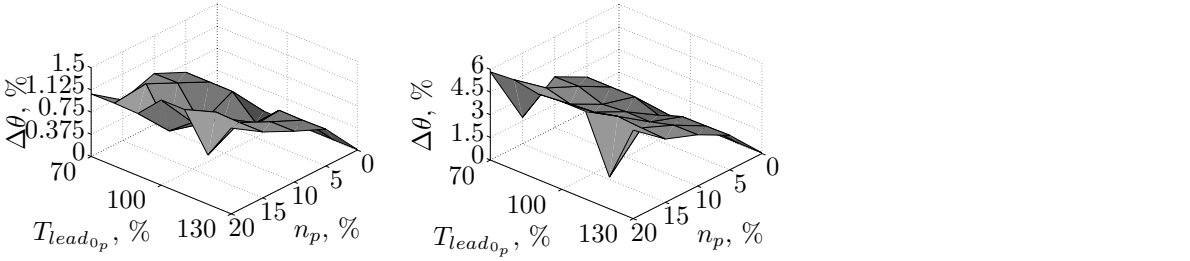


Figure C-17: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except T_{lead_0} which is varied over a range of parameter offsets ΔT_{lead_0} and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-18 Varied T_{lead_1} Closed-Loop Filtered Noise System.

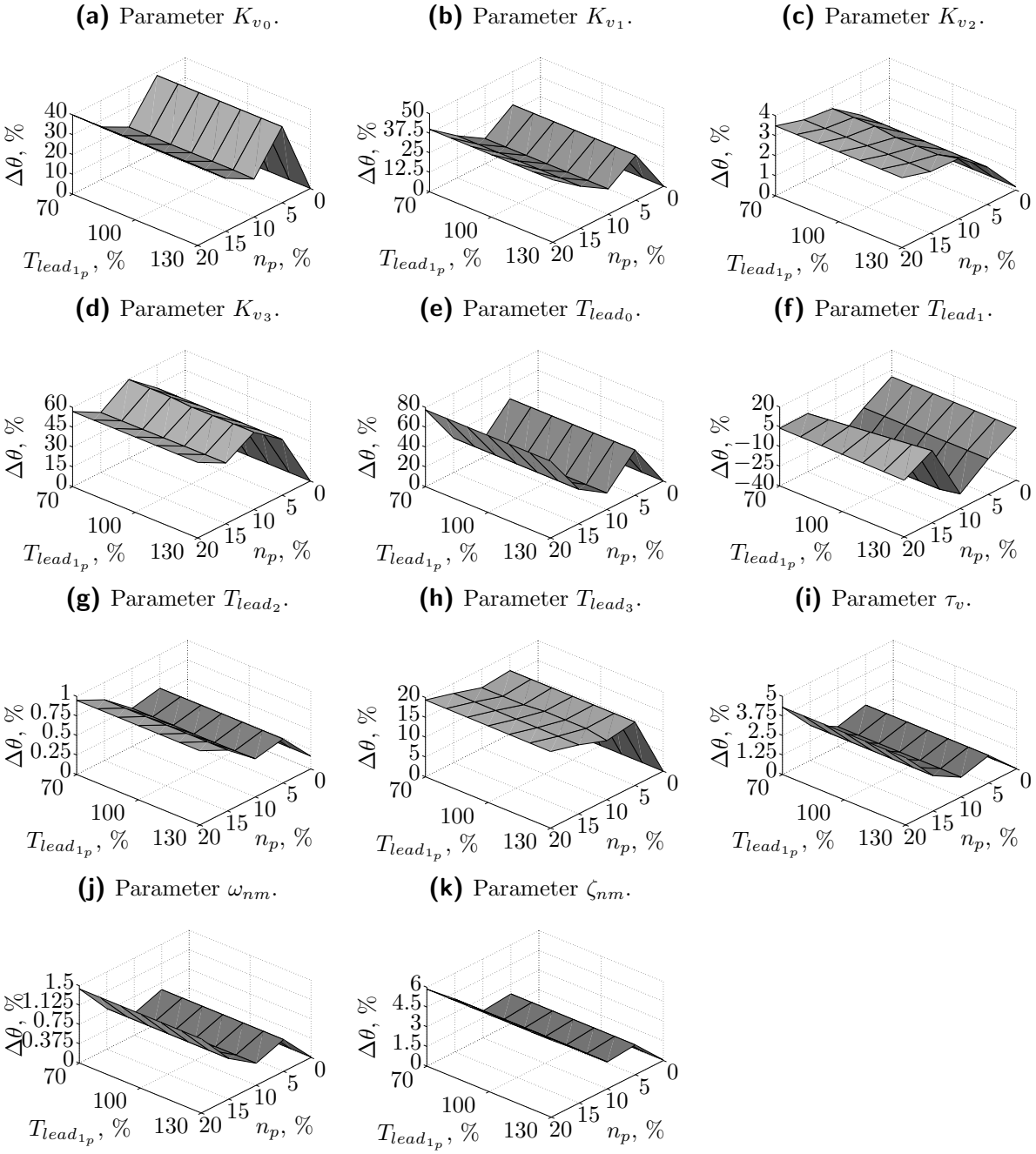
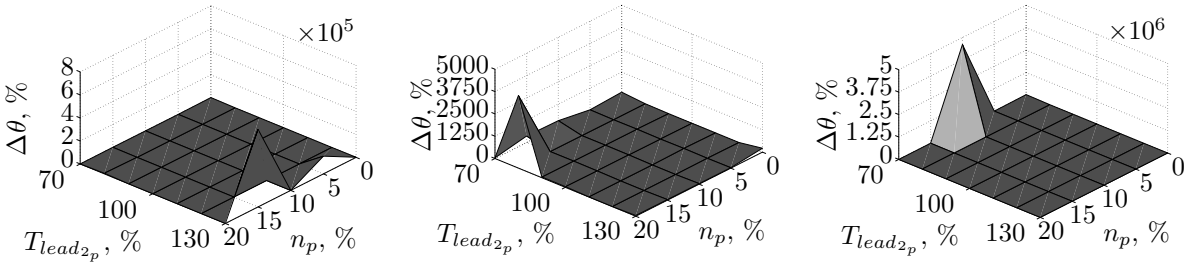


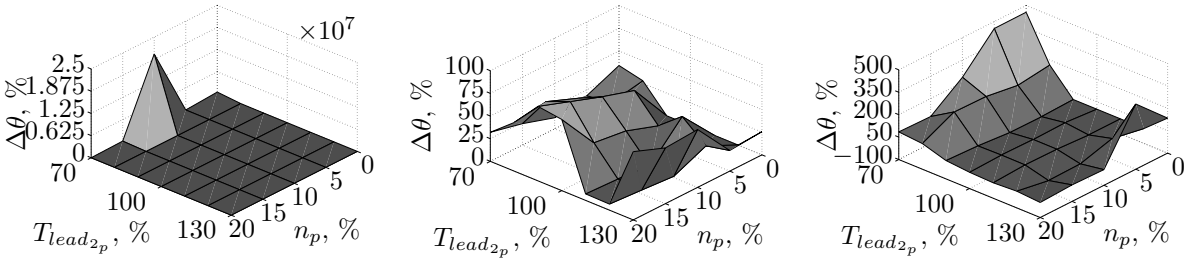
Figure C-18: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except T_{lead_1} which is varied over a range of parameter offsets ΔT_{lead_1} and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-19 Varied T_{lead_2} Closed-Loop Filtered Noise System.

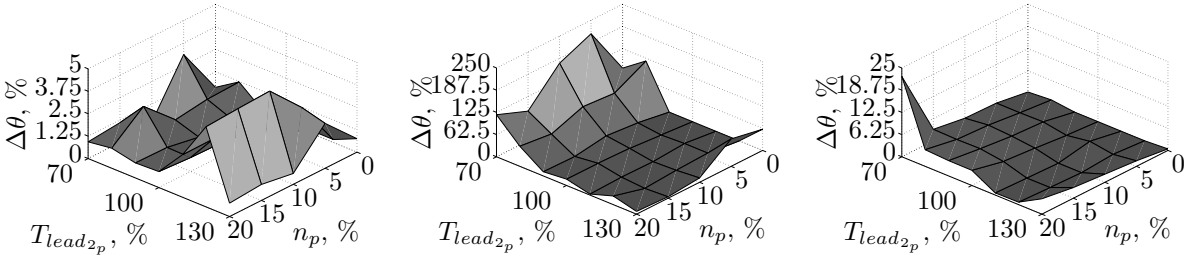
(a) Parameter K_{v_0}. **(b)** Parameter K_{v_1}. **(c)** Parameter K_{v_2}.



(d) Parameter K_{v_3}. **(e)** Parameter T_{lead_0}. **(f)** Parameter T_{lead_1}.



(g) Parameter T_{lead_2}. **(h)** Parameter T_{lead_3}. **(i)** Parameter τ_v.



(j) Parameter ω_{nm}. **(k)** Parameter ζ_{nm}.

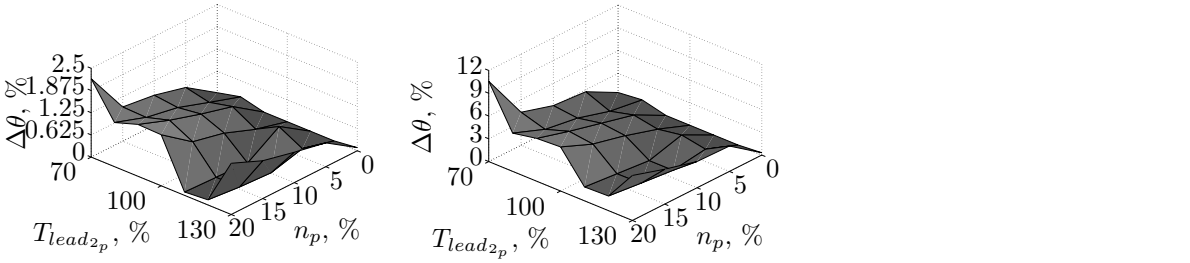


Figure C-19: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except T_{lead_2} which is varied over a range of parameter offsets ΔT_{lead_2} and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-20 Varied T_{lead_3} Closed-Loop Filtered Noise System.

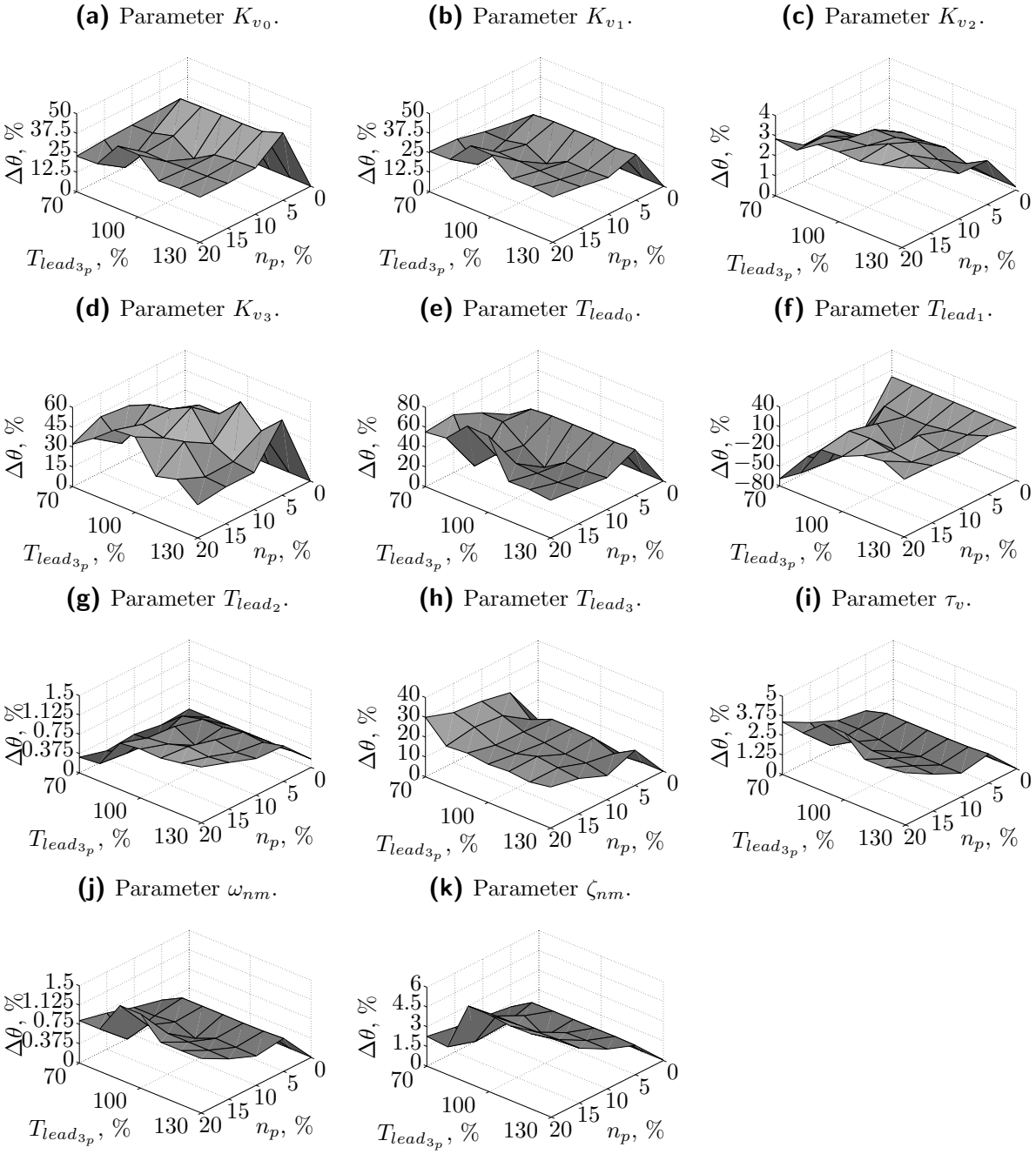


Figure C-20: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except T_{lead_3} which is varied over a range of parameter offsets ΔT_{lead_3} and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-21 Varied τ_v Closed-Loop Filtered Noise System.

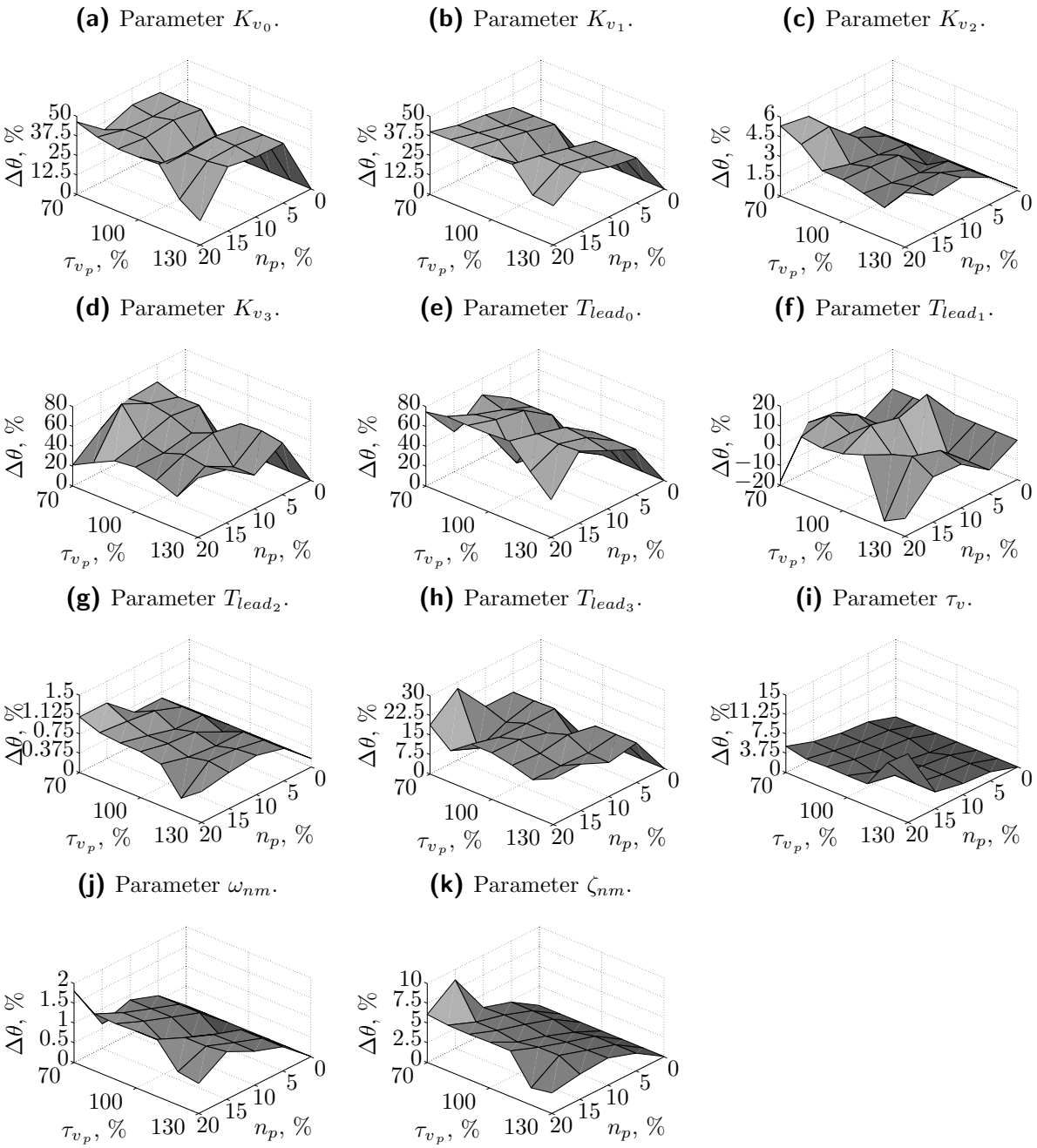


Figure C-21: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except τ_v which is varied over a range of parameter offsets $\Delta\tau_{v_0}$ and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-22 Varied ω_{nm} Closed-Loop Filtered Noise System.

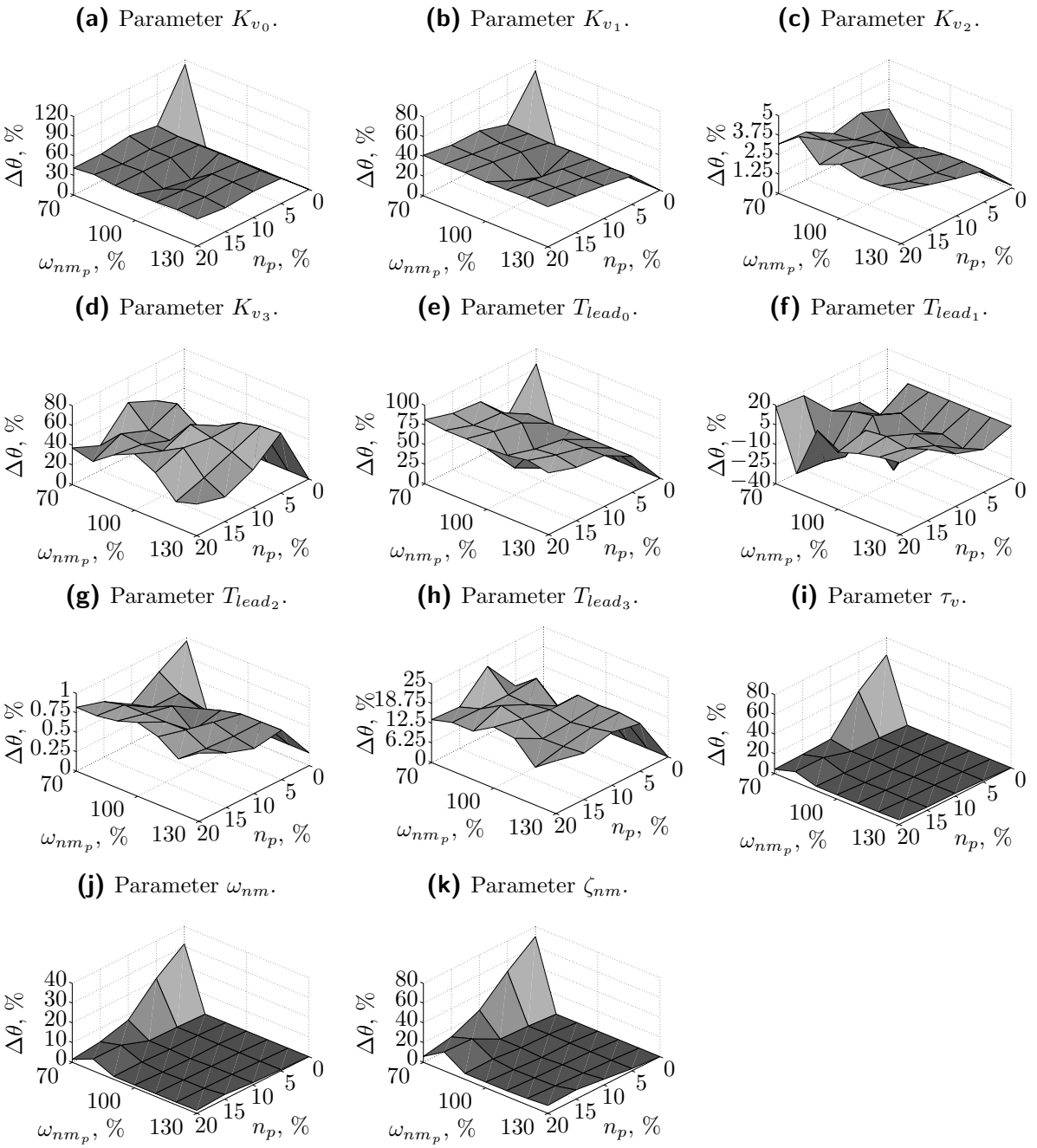


Figure C-22: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except ω_{nm} which is varied over a range of parameter offsets $\Delta\omega_{nm_0}$ and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

C-23 Varied ζ_{nm} Closed-Loop Filtered Noise System.

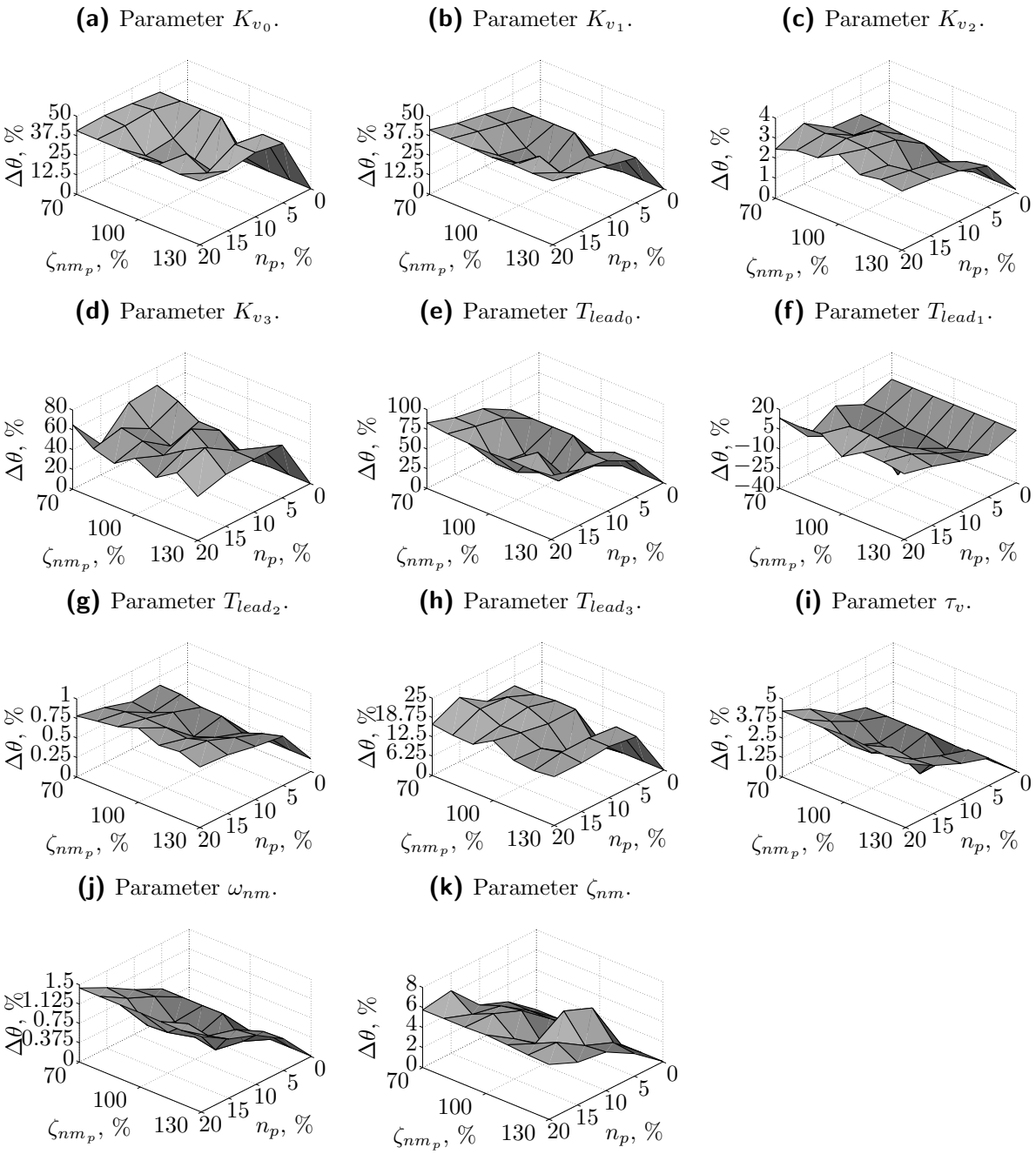
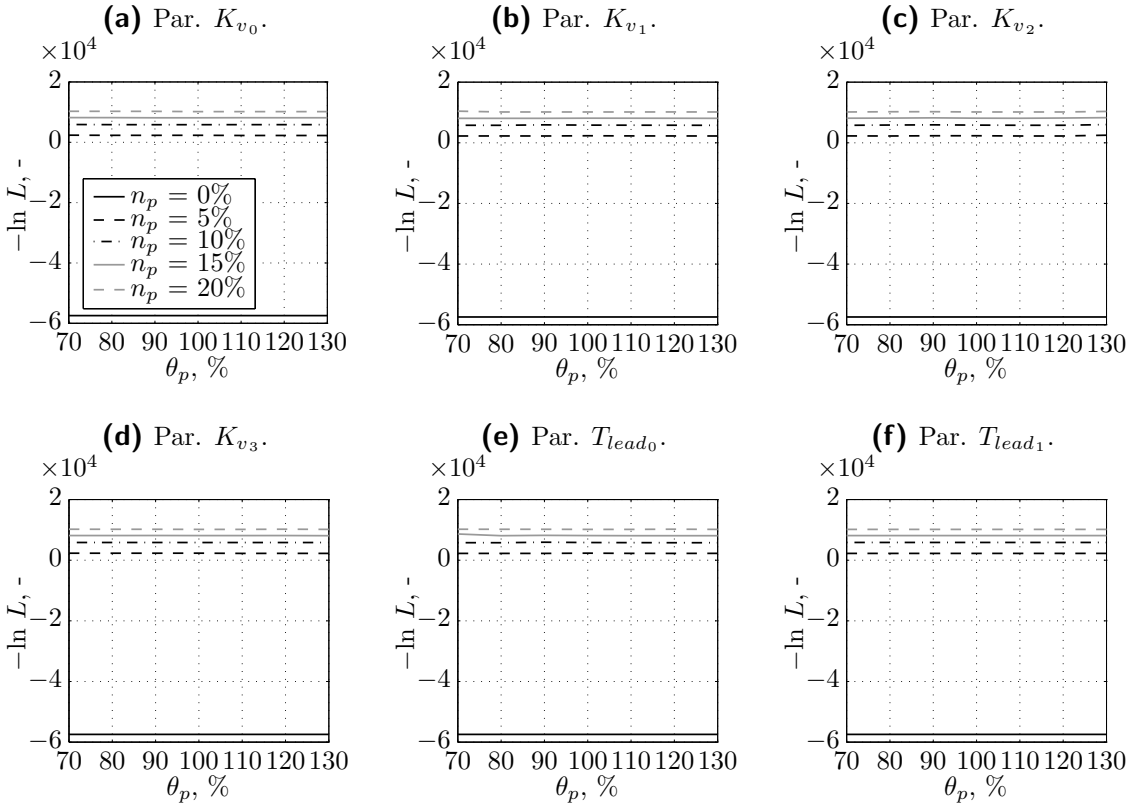


Figure C-23: Parameter difference $\Delta\Theta$: all parameter values are kept on the true values except ζ_{nm} which is varied over a range of parameter offsets $\Delta\zeta_{nm_0}$ and a range of remnant variance percentages n_p. MLE executed on complete, closed-loop, filtered noise system data.

Appendix D

Negative Log-Likelihood and Variance Percentages Plots



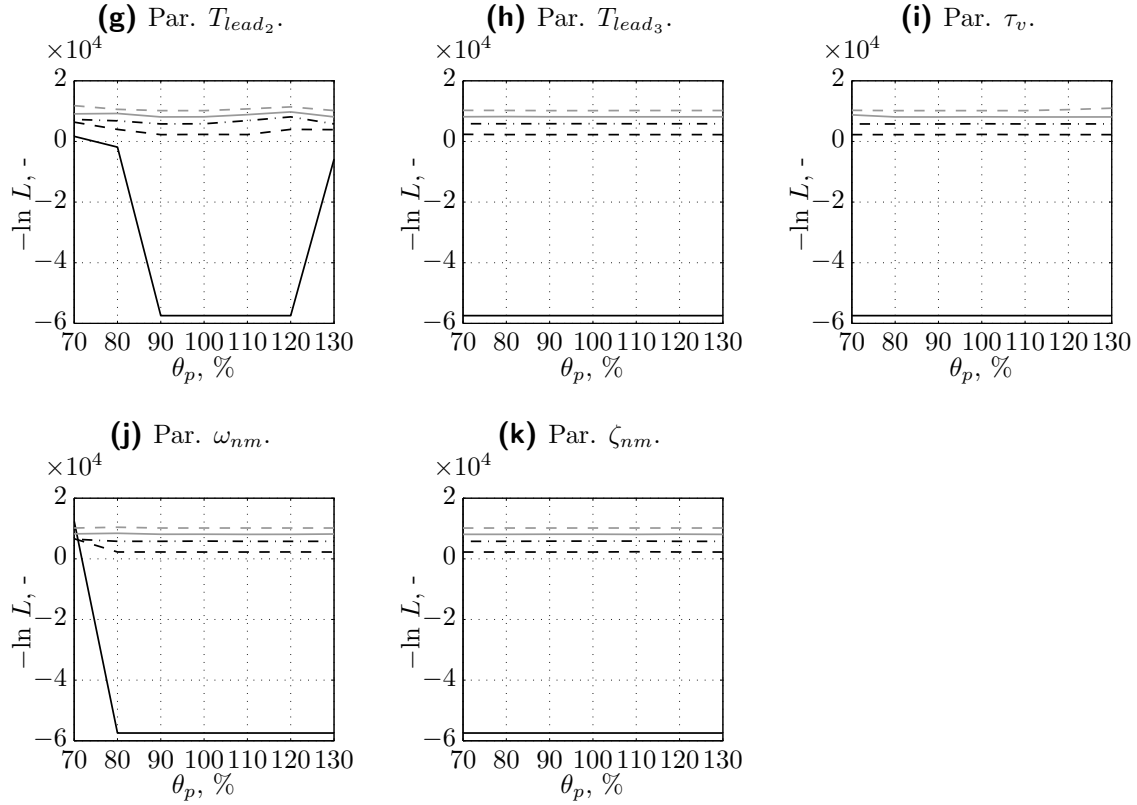


Figure D-0: The negative log-likelihood for different parameter variance percentages θ_p and different remnant variance percentages n_p. For the estimations in the figures only the respective parameter is varied for θ_p, all other parameters are held at a θ_p of 100%.

Bibliography

Abu-Naser, M., & Williamson, G. A. (2006, September). Convergence Properties of Adaptive Estimators of Time-Varying Linear Systems using Basis Functions. In *Digital Signal Processing Workshop, 12th - Signal Processing Education Workshop, 4th* (p. 336-341).

Abu-Naser, M., & Williamson, G. A. (2007, April). Convergence of Adaptive Estimators of Time-Varying Linear Systems using Basis Functions: Continuous Time Results. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2007.* (Vol. 3, p. III-1361-III-1364).

Altschul, R. E., Nagel, P. M., & Oliver, F. (1984, June). Statistical Time Series Models of Pilot Control with Applications to Instrument Discrimination. *Proceedings of the Twentieth Annual Conference of Manual Control*, 41-76.

Andrews, G., & Askey, R. (1985). Classical Orthogonal Polynomials. In C. Brezinski, A. Draux, A. Magnus, P. Maroni, & A. Ronveaux (Eds.), (Vol. 1171, p. 36-62). Springer Berlin / Heidelberg.

Apkarian, P., Biannic, J.-M., & Gahinet, P. (1993). *Gain-Scheduled $\mathcal{H}_\infty$ Control of a Missile via Linear Matrix Inequalities.*

Asutkar, V. G., Patre, B. M., & Basu, T. K. (2009a). Identification of Time-Varying Systems with Fast Changing Parameters using Forgetting Factor Approach. In *Proceedings of the International Conference on Advances in Computing, Communication and Control* (p. 715-719). ACM.

Asutkar, V. G., Patre, B. M., & Basu, T. K. (2009b, June). Kalman Filter Approach for Identification of Linear Fast Time-Varying Processes. In *International Conference on Control, Automation, Communication and Energy Conservation* (p. 1-5).

Biezad, D., & Schmidt, D. K. (1984, June). Time Series Modeling of Human Operator Dynamics in Manual Control. *Proceedings of the Twentieth Annual Conference of Manual Control*, 1-40.

Blanchard, E. D. (2010). *Polynomial Chaos Approaches to Parameter Estimation and Control Design for Mechanical Systems with Uncertain Parameters*. Ph.D. Dissertation, Virginia Polytechnic Institute and State University.

Brown, R. G. (1983). *Introduction to Random Signal Analysis and Kalman Filtering*. Wiley.

Campi, M. C. (1994, November). Exponentially Weighted Least Squares Identification of Time-Varying Systems with White Disturbances. *IEEE Transactions on Signal Processing*, *42*(11), 2906-2914.

Chan, S. C., & Zhang, Z. G. (2011). Local Polynomial Modeling and Variable Bandwidth Selection for Time-Varying Linear Systems. *IEEE Transactions on Instrumentation and Measurement*, *60*(3), 110-2-1117.

Chang, R. Y., & Wang, M. L. (1983). Application of Shifted Legendre Polynomials in Analysis of Dynamics Systems. *Journal of the Chinese Institute of Engineers*, *6*(4), 235-243.

Chen, C. F., & Hsiao, C. H. (1975, May). Time-domain Synthesis via Walsh Functions. *Proceedings of the Institution of Electrical Engineers*, *122*(5), 565-570.

Chu, Q. P. (1987). *Maximum Likelihood Parameter Identification of Flexible Spacecraft*. Ph.D. Dissertation, Faculty of Aerospace Engineering, Delft University of Technology.

Clement, P. R. (1982). Laguerre Functions in Signal Analysis and Parameter Identification. *Journal of the Franklin Institute*, *313*(2), 85-95.

Evensen, G. (2009). *Data Assimilation: The Ensemble Kalman Filter*. Springer.

Fan, J., & Yao, Q. (2005). *Nonlinear Time Series: Nonparametric and Parametric Methods*. Springer.

Fletcher, R. (2000). *Practical Methods of Optimization* (2nd ed.). Wiley.

Gelb, A. (1999). *Applied Optimal Estimation*. MIT Press.

Grouffaud, J., Larzabal, P., & Clergeot, H. (1997, April). A Maximum Likelihood Approach for the Passive Identification of Time-Varying Multipath Channels. In *IEEE International Conference on Acoustics, Speech, and Signal Processing* (Vol. 5, p. 3585 -3588).

Guo, J., & Dong, E. (2011, April). Parameter Identification for Time-Varying Systems by Evolutionary Neural Network. In *International Conference on Electric Information and Control Engineering, 2011* (p. 6009-6012).

Guo, L. (1990, February). Estimating Time-Varying Parameters by the Kalman Filter Based Algorithm: Stability and Convergence. *Automatic Control, IEEE Transactions on*, *35*(2), 141-147.

Hwang, C., & Guo, T. Y. (1984a). Identification of Lumped Linear Time-Varying Systems via Block-Pulse Functions. *International Journal of Control*, *40*(3), 571-583.

Hwang, C., & Guo, T. Y. (1984b). Parameter Identification of a Class of Time-Varying Systems via Orthogonal Shifted Legendre Polynomials. *Journal of the Franklin Institute*, *318*(1), 56-69.

Hwang, C., & Shih, Y. P. (1982). Parameter Identification via Laguerre Polynomials. *International Journal of Systems Science*, *13*(2), 209-217.

Jones, R. W. (2004, August). Recursive Estimation of Time-Delay using a High-order Pade Approximant. In *SICE 2004 Annual Conference* (Vol. 2, p. 1328-1333).

Kampen, E. van. (2010). *Global Optimization using Interval Analysis. Interval Optimization for Aerospace Applications.* Ph.D. Dissertation, Delft University of Technology.

Koekoek, R., Lesky, P. A., & Swarttouw, R. F. (2010). *Hypergeometric Orthogonal Polynomials and Their Q-Analogues*. Springer.

Kwong, C. P., & Chen, C. F. (1981). The Convergence Properties of Block-Pulse Series. *International Journal of Systems Science*, *12*(6), 745-751.

Ledin, J. (2003). *Embedded Control Systems in C/C++*. Elsevier Science.

Ljung, L. (1999). *System Identification - Theory For the User* (2nd ed.). Upper Saddle River, NJ: Prentice Hall PTR.

Ljung, L. (2007). Frequency Domain Versus Time Domain Methods in System Identification — Re-visited.

Ljung, L. (2010, March). Perspectives on System Identification. *Annual Reviews in Control*, *34*(1), 1-12.

Ljung, L., & Glover, K. (1981). Frequency Domain Versus Time Domain Methods in System Identification. *Automatica*, *17*(1), 71-86.

MacNeil, J. B., Kearney, R. E., & Hunter, I. W. (1992, December). Identification of Time-Varying Biological Systems from Ensemble Data (Joint Dynamics Application). *IEEE Transactions on Biomedical Engineering*, *39*(12), 1213-1225.

McRuer, D. T., Graham, D., Krendel, E., & Reisener Jr., W. (1965). *Human Pilot Dynamics in Compensatory Systems. Theory, Models and Experiments with Controlled Element and Forcing Function Variations.* Air Force Flight Dynamics Laboratory, Research and Technology Division, Air Force Systems Command.

McRuer, D. T., & Jex, H. R. (1967). A Review of Quasi-Linear Pilot Models. *IEEE Transactions on Human Factors in Electronics*, *8*(3), 231-249.

Morais, T. S., Steffen Jr., V., & Bachschmid, N. (2008). Time-Varying Parameter Identification using Orthogonal Functions. *Journal of Physics: Conference Series*, *135*(1).

Morrison, N. (1969). *Introduction to Sequential Smoothing and Prediction.* McGraw-Hill.

Motulsky, H., & Christopoulos, A. (2004). *Fitting Models to Biological Data Using Linear and Nonlinear Regression: A Practical Guide to Curve Fitting.* Oxford University Press.

Nahi, N. E. (1969). *Estimation Theory and Applications.* Wiley.

Nieuwenhuizen, F. M., Zaal, P. M. T., Mulder, M., Van Paassen, M. M., & Mulder, J. A. (2008, July-August). Modeling Human Multichannel Perception and Control Using Linear Time-Invariant Models. *Journal of Guidance, Control, and Dynamics*, *31*(4), 999-1013.

Pacheco, R. P., & Steffen Jr., V. (2002, July). Using Orthogonal Functions for Identification and Sensitivity Analysis of Mechanical Systems. *Journal of Vibration and Control*, *8*(7), 993-1021.

Palanisamy, K. R., & Bhattacharya, D. K. (1981). System Identification via Block-Pulse Functions. *International Journal of Systems Science*, *12*(5), 643-647.

Paraskevopoulos, P. N. (1983). Chebyshev Series Approach to System Identification, Analysis and Optimal Control. *Journal of the Franklin Institute*, *316*(2), 135-157.

Paraskevopoulos, P. N. (1985, June). Legendre Series Approach to Identification and Analysis of Linear Systems. *Automatic Control, IEEE Transactions on*, *30*(6), 585-589.

Paraskevopoulos, P. N., & Kekkeris, G. T. (1983). Identification of Time-Invariant and Time-Varying Distributed Parameter Systems using Chebyshev Series. *AMSE' 83 Conference, Nice, France*, *1*, 51-69.

Paraskevopoulos, P. N., & Kekkeris, G. T. (1988). Hermite Series Approach to Optimal Control. *International Journal of Control*, *47*(2), 557-567.

Paraskevopoulos, P. N., & King, R. E. (1976). *Parametric Identification using Discrete Laguerre Polynomials.* Greek Atomic Energy Commission, Nuclear Research Center 'Democritus'.

Paraskevopoulos, P. N., Sparis, P. D., & Mouroutsos, S. G. (1985). The Fourier Series Operational Matrix of Integration. *International Journal of Systems Science*, *16*(2), 171-176.

Pintelon, R., & Schoukens, J. (2001). *System Identification: A Frequency Domain Approach*. New York, NY: IEEE Press.

Rao, G. P., & Sivakumar, L. (1975, October). System Identification via Walsh Functions. *Electrical Engineers, Proceedings of the Institution of*, *122*(10), 1160-1161.

Raol, J., Girija, G., & Singh, J. (2004). *Modelling and Parameter Estimation of Dynamic Systems*. Institution of Electrical Engineers.

Rémond, D., Neyrand, J., Aridon, G., & Dufour, R. (2008, July-August). On the Improved Use of Chebyshev Expansion for Mechanical System Identification. *Mechanical Systems and Signal Processing*, *22*, 390-407.

Sanyal, S., Kukreja, S. L., Perreault, E. J., & Westwick, D. T. (2005, September). Identification of Linear Time Varying Systems using Basic Pursuit. *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai, China*.

Schölkopf, B., & Smola, A. J. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press.

Shih, Y. M. (1983). Application of Chebyshev Polynomials in Analysis and Identification of Linear Systems. *Journal of the Chinese Institute of Engineers*, *6*(2), 135-140.

Tóth, R. (2010). *Modeling and Identification of Linear Parameter-Varying Systems*. Springer.

Tzafestas, S. G. (1978). Walsh Series Approach to Lumped and Distributed System Identification. *Journal of the Franklin Institute*, *305*(4), 199-220.

Vapnik, V. N. (1998). *Statistical Learning Theory*. Wiley.

Verspecht, T. (2011). *Identification of Time Variant Neuromuscular Admittance using Wavelets*. MSc. Thesis, Faculty of Aerospace Engineering, Delft University of Technology.

Vuik, C., Van Beek, P., Vermolen, F., & Van Kan, J. (2007). *Numerical Methods for Ordinary Differential Equations*. VSSD.

Westwick, D. T., & Kearney, R. E. (2003). *Identification of Nonlinear Physiological Systems*. IEEE Press.

Wittenmark, B. (1979). A Two-Level Estimator for Time Varying Parameters. *Automatica*, *15*(1), 85-89.

Zaal, P. M. T. (2011). *Pilot Control Behavior Discrepancies Between Real and Simulated Flight Caused by Limited Motion Stimuli.* Ph.D. Dissertation, Faculty of Aerospace Engineering, Delft University of Technology.

Zaal, P. M. T., Pool, D. M., Chu, Q. P., Van Paassen, M. M., Mulder, M., & Mulder, J. A. (2009, July-August). Modeling Human Multimodal Perception and Control Using Genetic Maximum Likelihood Estimation. *Journal of Guidance, Control, and Dynamics*, *32*(4), 1089-1099.

Zaal, P. M. T., Pool, D. M., Mulder, M., & Van Paassen, M. M. (2008). New Types of Target Inputs for Multi-Modal Pilot Model Identification. In *Proceedings of the AIAA Modeling and Simulation Technologies Conference and Exhibit, Honolulu (HI).*

Zaal, P. M. T., & Sweet, B. T. (2011, August). Estimation of Time-Varying Pilot Model Parameters. In *Proceedings of the AIAA Modeling and Simulation Technologies Conference, Portland, OR* (Vol. 1, p. 598-614).

Zhu, H., & Wu, H. (2007, December). Estimation of Smooth Time-Varying Parameters in State Space Models. *Journal of Computational and Graphical Statistics*, *16*(4), 813-832.

Zhu, Y., & Pagilla, P. R. (2003, June). Adaptive Estimation of Time-Varying Parameters in Linear Systems. In *Proceedings of the American Control Conference, 2003.* (Vol. 5, p. 4167-4172).