M.Sc. Thesis

Non-Linear Bayesian System Identification of Cortical Responses Using Volterra Series

Mike de Pont

Technische Universiteit Delft

TU Delft

DCSC

M.Sc. Thesis

Non-linear Bayesian System Identification of Cortical Responses Using Volterra Series

by

Mike de Pont

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday October 16, 2020 at 14:00.

Student number:	4323955	
Project duration:	January 1 2020 – September 1, 2020	
Thesis committee:	Prof. dr. ir. Jan-Willem van Wingerden,	TU Delft
	dr. ir. Alfred C. Schouten,	TU Delft
	dr. ir. Kim Batselier,	TU Delft, supervisor

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

Acknowledgements

In front of you lies the final piece of seven wonderful years of studying at TU Delft. Looking back, I would like to express my gratitude to my former and current roommates for all the fun during my college days. In good times and bad, they have always been open to socializing. In the long term, this has contributed greatly to my development and motivation to complete everything successfully. Furthermore, I would like to thank Vera de Pont for designing the cover image. In addition, I would like to thank my parents for all their inexhaustible support during the past seven years.

Finally, I would like to thank my daily supervisor dr. ir. Kim Batselier for his help during the last nine months. The problems encountered during this research were often complex, but we struggled through them during the weekly meetings. Without his help, this study could have taken a lot longer.

Abstract

The human sensorimotor system can be seen as a complex network in which the brain plays an important role, resulting in a difficult-to-understand relation between proprioceptive stimuli and cortical responses. However, understanding this relationship is of added value for understanding various diseases which cause dysfunctionality. In recent years, a variety of studies have been conducted towards finding the non-linear relationship between the cortical responses and wrist joint manipulation. This research is dedicated to providing an initial set-up to create models that are able to provide accurate predictions despite noisy data. The relationship between wrist joint manipulation and the cortical response is assumed to be non-linear and the corresponding identification method is categorized in a two-step process, namely the model structure, i.e. Volterra series, and stochastic identification method, i.e. Bayesian Inference. To understand the working principle of the proposed algorithm, the method is first applied to a set of computer models. Finally, an attempt is made to model the cortical responses evoked by wrist joint manipulations.

List of Figures

1.1 Common symptoms for Parkinson's disease . . . 1
1.2 DARPA Challenge . . . 1
1.3 The dominant cortical response of six different participants evoked by the same input signal (top-left). The figure substantiates the non-linear characteristics of EEG signals and shows the oscillatory behaviour which may be caused by noise corruption. . . . 2
1.4 The experimental setup conducted by Vlaar [35, 36]. The participants were instructed to gaze at a static screen, while the rightforearm and hand are fixated such that the right wrist joint is aligned with the robotic manipulator, as shown in the lower-right image.The manipulator excites a sequence of three different multisine realizations subsequently, as shown in the top-right image. Each real-ization consists only odd harmonic frequencies ranging from 1 to 23 Hz. One of the example realizations is depicted in the lower-leftimage. . . . 3

2.1 The number of unique parameters up to the fourth order Volterra series . . . 7
2.2 An example of a second-order Volterra kernel [5] including the rotated coordinate system. . . . 10

3.1 Ground truth Volterra model. Left: $h_1(\tau_1)$, right: $h_2(\tau_1,\tau_2)$. . . . 18
3.2 The ground truth Neural Network model structure . . . 19
3.3 The singular values of the regression matrix $\mathbf{U}$ for six different model structures constructed with both a multisine input and a GWN input. . . . 20
3.4 The performance of the competitive model class set while modeling the final 250 time steps of the Volterra ground truth system . . . 21
3.5 The prior MCD of the competitive model obtained with an uninformative prior . . . 21
3.6 Relative Entropy vs. log-datafit of the second degree model classes . . . 22
3.7 The posterior parameter PDF compared with the ground truth parameters obtained with an uninformative prior . . . 23
3.8 Prior MCD obtained with an informative prior . . . 23
3.9 The posterior parameter PDF compared with the ground truth parameters obtained with an informative prior . . . 24
3.10 Relative Entropy of the twelve candidate model classes compared between the informative and uninformative prior . . . 24
3.11 Predictive posterior MCD obtained with an uninformative prior . . . 25
3.12 Predictive posterior MCD obtained with an informative prior . . . 25
3.13 Posterior parameters of the second degree model classes . . . 26
3.14 Hyper Robust Predictions . . . 26
3.15 Prior MCD obtained with an uninformative prior . . . 27
3.16 Predictive posterior MCD obtained with an uninformative prior . . . 28
3.17 Prior MCD obtained with the altered prior $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.4 \cdot \mathbf{I}_{n_H}\right)$ . . . 28
3.18 Predictive posterior MCD obtained with the altered prior $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.4 \cdot \mathbf{I}_{n_H}\right)$ . . . 29
3.19 The Relative Entropy comparison for two different prior variances . . . 29
3.20 Two normal distributions . . . 30
3.21 Relation between σ_{HRP} and $\mathbf{y}_{\text{intersect}}$ . . . 30
3.22 Prior MCD in noisy conditions obtained with an uninformative prior . . . 31
3.23 Predictive posterior MCD in noisy conditions obtained with an uninformative prior . . . 32
3.24 Prior MCD in noisy conditions obtained with an informative prior . . . 32
3.25 Predictive posterior MCD in noisy conditions obtained with an informative prior . . . 33

4.1 Schematic representation of the nervous system . . . 35
4.2 Prior MCD obtained with an uninformative prior . . . 37
4.3 Predictive posterior MCD obtained with an uninformative prior . . . 37

4.4 Perforamce of $\mathcal{M}_{2,60}$ . . . 38
4.5 Prior MCD obtained with an informative prior . . . 38
4.6 Predictive posterior MCD during validation . . . 38
4.7 Singular values of the regression matrix of $\mathcal{M}_{2,60}$ . . . 39
4.8 Validation sequence of the best performing models of participants 1 and 7 . . . 41
4.9 Autocorrelation of the residuals . . . 42
4.10 Autocorrelation of the residuals . . . 42
4.11 Cross correlation of the residuals with the input vector $\mathbf{U}_N$ . . . 43

5.1 Alternative random walk input sequence . . . 46

List of Tables

3.1 Volterra ground truth: Hyperparameters used to construct the prior variance matrix 17
3.2 Volterra ground truth: evaluation of the Log-Evidence for the second degree model classes . . 22
3.3 Volterra ground truth: performance of candidate models obtained with an uninformative prior 25
3.4 Volterra ground truth: performance of candidate models with an informative prior 26
3.5 NN ground truth: performance of candidate models with an uninformative prior 28
3.6 NN ground truth: performance of candidate models obtained with the altered prior $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.4 \cdot \mathbf{I}_{n_H}\right)$ 29
3.7 NN ground truth: performance of candidate models in noisy conditions obtained with an uninformative prior 31
3.8 NN ground truth: performance of the candidate models in noisy conditions obtained with an informative prior 32

4.1 Rank of $\mathbf{U}$ per model class 39
4.2 Cortical responses: performance of candidate models obtained with an informative prior . . . 40
4.3 Cortical responses: performance of candidate models obtained with an informative prior . . . 40

Contents

Acknowledgements iii

Abstract v

List of Figures vii

List of Tables ix

1 Introduction 1
- 1.1 Thesis Objective . . . 3
- 1.2 Thesis Outline . . . 4

2 Volterra Series System Identification: A Bayesian Approach 5
- 2.1 Volterra Series . . . 5
- 2.2 Bayesian Inference . . . 7
  - 2.2.1 Parameter Estimation . . . 8
  - 2.2.2 Model Comparison . . . 12
  - 2.2.3 Predictive Analysis . . . 12
- 2.3 Model Evaluation . . . 14

3 Computer Simulations 17
- 3.1 Experimental Setups . . . 17
  - 3.1.1 Volterra . . . 17
  - 3.1.2 Noisy Neural Network . . . 18
- 3.2 Modeling approach . . . 18
  - 3.2.1 Competitive Model Class Set . . . 18
  - 3.2.2 Excitation Signal . . . 19
- 3.3 Results . . . 20
  - 3.3.1 Volterra . . . 20
  - 3.3.2 Noisy Neural Network . . . 27
- 3.4 Conclusion . . . 32

4 Cortical Responses 35
- 4.1 Experimental Setup . . . 35
- 4.2 Modeling Approach . . . 36
  - 4.2.1 Single participant . . . 36
  - 4.2.2 All participants . . . 36
- 4.3 Results . . . 37
  - 4.3.1 Single participant . . . 37
  - 4.3.2 All participants . . . 39
- 4.4 Conclusion . . . 42

5 Discussion and Recommendations 45
- 5.1 Discussion . . . 45
  - 5.1.1 Independency of the Model Classes . . . 45
  - 5.1.2 Choice of Input Sequence . . . 45
  - 5.1.3 Reflection on Bayesian Model Averaging . . . 46
  - 5.1.4 Reflection on Modeling Approach . . . 46
  - 5.1.5 Relation with Previous Performed Studies . . . 47
- 5.2 Recommendations . . . 47

5.2.1 Tensor Decomposition for Higher Order Volterra Series 47
5.2.2 Bayesian Inference for Frequency Analysis 48
5.2.3 Estimating the Model Class Distribution 48

6 Conclusion **49**

Bibliography **51**

1 Introduction

Many diseases which cause dysfunctionality are nowadays primarily diagnosed by common symptoms and have no objective method that can either deny or confirm the disease. Like many others, Parkinson's disease (Fig. 1.1) causes a variety of disorders such as tremor of extremities and reducing arm movement, which is caused by the break down of nerve cells in the brain. However, even though it is suspected that the disease is due to gene mutations, the exact origin of it is still unknown. Understanding the human sensorimotor system can aid in comprehending several movement disorders and may lead to earlier detection of these disorders. Furthermore, the mechanism in the brain underlying the control of joints is complicated and is so far too unknown to be applied to control robotic systems. Understanding this mechanism offers opportunity to incorporate more intelligent control algorithms, in order to achieve tasks which are considered to be very difficult to date. The Defence Advance Research Projects Agency (DARPA) hosts a variety of challenges to field human-like robotics in practical applications. One of the tasks requires the robotic systems to open and close doors (Fig. 1.2), which unfortunately often does not yield to satisfactory results. Researching the human sensorimotor system advances the field of human supervised control of robotic systems.

Studying the dynamic relation between the wrist joint manipulation and the electrical activity of the brain requires a proprioceptive stimulus of the joint and a cortical measurement technique, such as electroencephalography (EEG). The corresponding device consists of an arbitrary number of electrodes, which measures the voltage fluctuations on the scalp caused by ion-activity of the neurons in the brain. Thus, it is a non-invasive medical method that is relatively accessible and cheap compared to alternatives such as functional Magnetic Resonance Imaging (fMRI). Additionally, EEG can accurately detect activity in a high resolution, making it suitable for signal analysis used for studying the brain.

However, difficulties arise when processing EEG data. The electrical activity produced by the brain is of the order $10\mu V$, making it sensitive to ambient external electric sources either present in the human body (e.g. facial muscle artifacts) or in the room (e.g. electrical wiring). In addition, the brain is continuously involved in multiple processes, resulting in extra noise signals irrelevant for one's experiment. Hence, the acquired

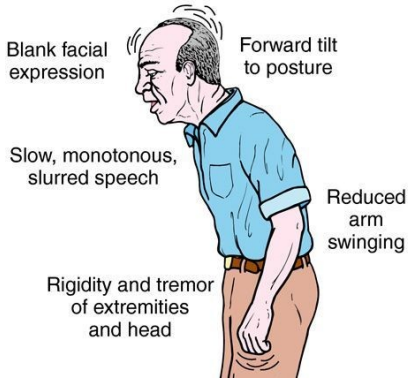


Figure 1.1: Common symptoms for Parkinson's disease

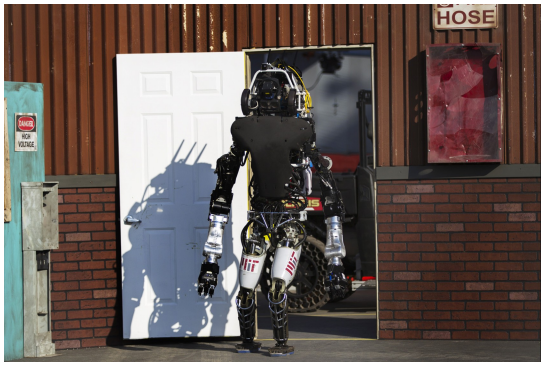


Figure 1.2: DARPA Challenge

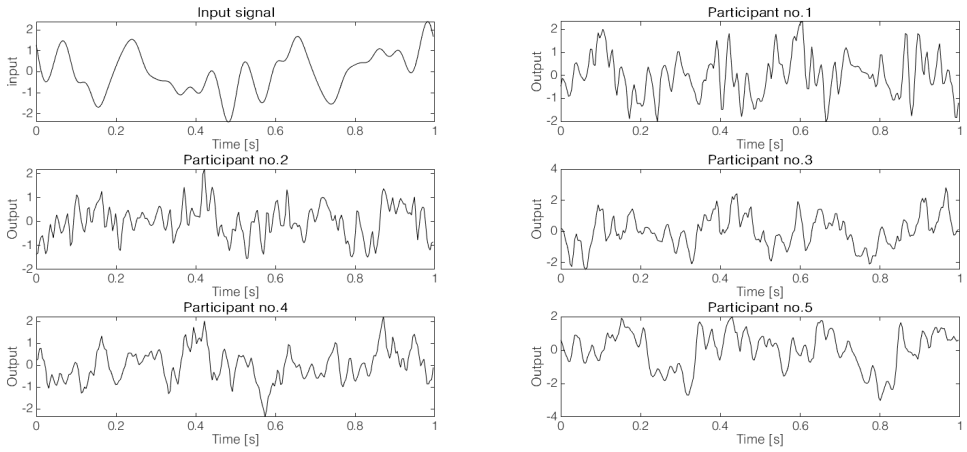


Figure 1.3: The dominant cortical response of six different participants evoked by the same input signal (top-left). The figure substantiates the non-linear characteristics of EEG signals and shows the oscillatory behaviour which may be caused by noise corruption.

cortical response's signal-to-noise ratio (SNR) is not satisfactory. Moreover, a recent study has shown that only 10% of the brain activity could be explained using a best linear approximation, meaning that over 80% of the response is caused by non-linear behaviour [35]. Non-linear system identification techniques are often less well developed and it requires a more complex approach, making it computationally challenging. Fig. 1.3 illustrates the most dominant signal of the brain's activity (highest SNR) of six different participants evoked by the same input signal.

The non-linear relationship between the somatosensory system, i.e. the part of the brain that processes proprioceptive stimuli, and wrist joint manipulation has been a topic of interest in system identification. Tian et al. [32] tried to use a Non-linear Auto-Regressive Moving Average with eXogenous input (NARMAX) model in combination with a Hierarchical Neural Network (HNN) to capture the dynamics of the human sensory system. The NARMAX-HNN model generates satisfactory results, giving a mean Variance Accounted For (VAF) of 92.33% over ten subjects with standard deviation of 1.57%. This research was based on an earlier study by Vlaar et al. [36], who conducted the experiments as shown in Fig. 1.4 where he obtained the input and output data needed to describe the relationship between wrist joint manipulation and cortical response. Vlaar et al. [36] identified a regularized zeroth and second order Volterra series model combined with a Best Linear Approximation model and managed to find a mean VAF of 46.20% with a standard deviation of 8.32% over the participants.

Although this implies that the NARMAX-HNN model is preferable to the Volterra model, there are some important comments to report. To begin with, the Volterra model is linear with regard to the parameters, meaning that the resulting minimization problem is convex and can be solved using linear optimization techniques. The NARMAX-HNN however is a combination of two non-linear functions, resulting in a non-convex optimization problem that requires more sophisticated algorithms. Moreover, the NARMAX function includes sampled auto-regressive terms, accounting for possible closed loop behaviour. The Volterra model does not take that into account. Secondly, the NARMAX-HNN model is presented as a dual-input-single-output model. Based on neuroanatomical connections, Tian et al. [32] used both the perturbation signal as well as its first derivative as input signal, providing the algorithm with more information. Consequently, this complicates the model structure. On the contrary, Vlaar et al. [36] only considers the perturbation signal, which contributes to the simplicity of the model. Besides, one of the main advantages of Volterra series is that no prior knowledge is required regarding the true underlying structure of the system. Finally, in the NARMAX-HNN model a conscious decision was made to include a neural network to accommodate for higher order dynamics, whereas the Volterra model only captures zeroth, first and second order dynamics. Low order dynamic models are unable to explain high frequency behaviour generated by higher order dynamics.

A commonality among the two approaches is that both ought to find a single mathematical relationship to explain the EEG data and to make subsequent predictions. This model selection procedure is based on the VAF, where after the best performing model is selected for validation. However, in both studies, there is no clear substantiation as to why a model is given full preference, while there remains uncertainty which model outperforms the other models. Additionally, VAF is unable to detect a bias between the true and modeled

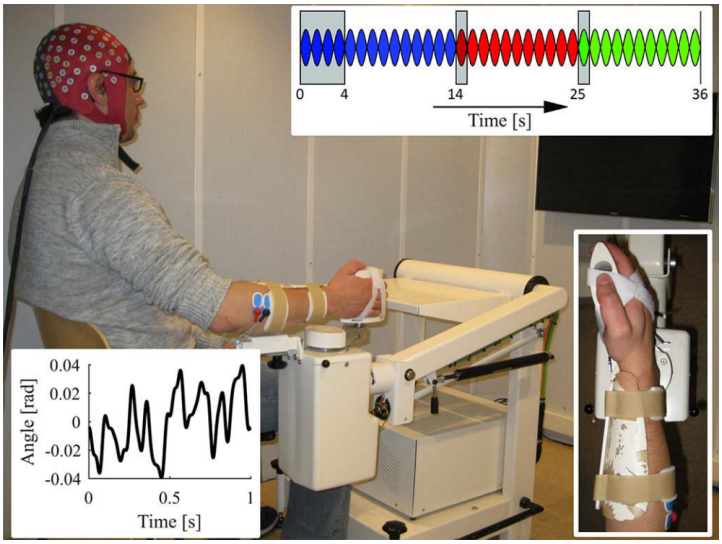


Figure 1.4: The experimental setup conducted by Vlaar [35, 36]. The participants were instructed to gaze at a static screen, while the rightforearm and hand are fixated such that the right wrist joint is aligned with the robotic manipulator, as shown in the lower-right image.The manipulator excites a sequence of three different multisine realizations subsequently, as shown in the top-right image. Each real-ization consists only odd harmonic frequencies ranging from 1 to 23 Hz. One of the example realizations is depicted in the lower-leftimage.

output. Besides, EEG is usually highly corrupted with noise, which affects the level of uncertainty of the involved parameters. Conventional methods, as applied in [32, 36], do not include this uncertainty in the model selection procedure, which may lead to false conclusions which model performs best.

This motivates the use of Bayesian Inference, which allows one to incorporate uncertainty information while comparing different models [4]. This concept relies on conventional probabilistic theory and it is suitable for estimating parameters from posterior distributions. This distribution is a result of Bayes' rule, i.e. combining prior knowledge with obtained data. The technique has been investigated thoroughly in EEG applications, although all are focused on solving inverse problems. This class of problems applied on EEG data is from a mathematical point of view ill-posed, since the number of neurons present in the brain causing the electric activity is greatly larger than the number of sensors attached on one's scalp [33]. Having said that, past studies primarily distinguish themselves from each other by taking different assumptions on the prior distribution, resulting in different regularization techniques. Among others, well known prior distributions are the Gaussian [33], Bernoulli-Laplacian [8] and Gibbs [19] distributions.

Furthermore, Bayesian Inference allows one to combine the predictions of a variety of candidate model classes, which is known as Bayesian Model Averaging (BMA). Where conventional system identification methods select a single model class based on a arbitrary performance index, BMA combines the model classes by summing predictions weighted by its relative level of significance. It is shown by Beck [3] and Trujillo-Barreto et al. [33] that averaging over the competitive model class set may lead to a better predictive ability by reducing the expected squared error with respect to single chosen model, which would be accepted using conventional methods. The Bayesian Inference approach is summarized in a three-level process [20, 33], namely:

1. Given assumptions $\mathcal{H}$ and the available data $\mathcal{D}$, the posterior probability of the parameters $\boldsymbol{\theta}$ is calculated, according to Bayes' rule.

2. Again, according to Bayes' rule and given the data $\mathcal{D}$, different alternative assumptions $\mathcal{H}$ are compared.

3. The uncertainty is taken into account when subsequent predictions are made. The prediction about some quantity $\mathbf{t}$ is obtained by summing over different assumptions $\mathcal{H}$.

1.1. Thesis Objective

The experiments in this report are intended to provide a better understanding of non-linear Bayesian system identification using Volterra series and its effects in making subsequent predictions. This has been done

for the following two reasons. First, existing studies provide a method to explain EEG data evoked by wrist joint manipulations, however, Tian et al. [32], Vlaar et al. [36] did not incorporate any uncertainties while performing parameter estimation, which affects the model selection process. By taking this into account through Bayesian Inference, it is expected to make a difference in the relative performance between models. Second, the current studies seek to find a single best model for making predictions, however, it has been demonstrated (i.e. by Hoeting et al. [15], Raftery et al. [25]) that this approach may lead to overconfident forecasts. Bayesian Model Averaging mitigates this overconfidence by averaging over the competitive model set when making predictions.

Understanding the performance of Bayesian Inference using Volterra Series on the system identification of cortical responses evoked by wrist joint manipulations is a difficult process, as the actual brain waves cannot be measured. For this reason, this study first applies the algorithms on a set of computer models. This provides a more objective picture of the advantages and disadvantages of the proposed approach, as the modeled outputs can be compared with the underlying ground truth signal, which are unknown investigating brain waves.

Having said this, the main objective of this thesis is:

The Development of Non-Linear Bayesian System Identification of the Cortical Response Evoked by Wrist Joint Manipulation Using Volterra Series

To achieve this, the following sub-objectives are formulated that are applied to both the computer models as well as the cortical response data.

Sub-objective 1. Understanding the effect of incorporating uncertainty on the parameter estimation and the model selection process.

Sub-objective 2. Examining whether it is beneficial to perform Bayesian Model Averaging compared to conventional methods while modeling.

Sub-objective 3. Understanding the effect of imposing different prior Gaussian distributions on the performance of the algorithm.

This research is innovative for the following two reasons. First of all, Bayesian Inference has not previously been used for the system identification of cortical responses evoked by wrist joint manipulations. Second, Bayesian Inference in conjunction with Volterra Series has not been studied before.

1.2. Thesis Outline

The body of this study is subdivided in three parts. First, chapter 2 introduces the theory needed to understand the work that has been done. Second, the algorithm is applied on two types of computer models in chapter 3, namely a Volterra System and a Neural Network. In chapter 4, the method is applied to model the cortical responses evoked by wrist joint manipulations. Then, chapter 5 provides a reflection on the acquired results and recommends techniques for future work. Finally chapter 6 concludes this study.

2

Volterra Series System Identification: A Bayesian Approach

This chapter provides the theory needed to understand the work that has been done in this study. Section 2.1 discusses the theory behind Volterra Series, i.e. the non-linear model structure. Section 2.2 is devoted to the three levels of Bayesian Inference, which include system identification, model comparison and predictive analysis. Finally, section 2.3 discloses the techniques used to evaluate the output of a system and to evaluate the performance of the modeled systems.

2.1. Volterra Series

The Volterra Series is a model for non-linear systems of which the output of the system depends on all past inputs, given that the energy of excited input signal is limited. The discrete-time, non-linear and time-invariant Volterra Series for single-input single-output (SISO) systems can be represented as:

$$\begin{cases} q(n) = h_0 + \sum_{d=1}^{D} q^d(n) \\ q^d(n) = \sum_{\tau_1=1}^{L} \cdots \sum_{\tau_d=1}^{L} h_d(\tau_1,\ldots,\tau_d) \prod_{i=1}^{d} u(n-\tau_i). \end{cases} \tag{2.1}$$

Here, d, D and L denote the order, degree and maximum lag of the Volterra series respectively. Note the difference between order and degree: a third degree Volterra Series ($D = 3$) contains Volterra kernels of order 0, 1, 2 and 3. Furthermore, $h_d(\tau_1,\ldots,\tau_d)$ represents each real valued d-th order symmetric Volterra kernel consisting the parameters to be determined, $u(n-\tau_i)$ represents the lagged input of the system, $q(n)$ is the modeled output and $\tau_1,\ldots,\tau_d$ denote the lag variables. The model structure in eq. (2.1) is linear in the parameters, meaning that it is suitable for linear optimization techniques. The following example aids in finding the general SISO linear model structure, which follows the procedure well described by Batselier et al. [2].

Example 1. *Consider the second degree SISO Volterra Series with lag 2, which is described by*

$$\begin{aligned} q(n) = & h_0 \\ & + h_1(1)u(n-1) + h_1(2)u(n-2) \\ & + h_2(1,1)u(n-1)^2 + h_2(1,2)u(n-1)u(n-2) + h_2(2,2)u(n-2)^2. \end{aligned} \tag{2.2}$$

Here it is assumed that all the higher order ($d > 1$) Volterra kernels are symmetric. Equation (2.2) can alternatively be written as:

$$q(n) = h_0 + \mathbf{h}_1^T \mathbf{u}_n + \mathbf{h}_2^T (\mathbf{u}_n \otimes \mathbf{u}_n), \tag{2.3}$$

which illustrates the contribution of each d-th order Volterra system. Here, $\otimes$ denotes the Kronecker product, $\mathbf{u}_n \in \mathbb{R}^2$ is a vector with its entries given by lagged inputs and $h_0 \in \mathbb{R}$, $\mathbf{h}_1 \in \mathbb{R}^2$ and $\mathbf{h}_2 \in \mathbb{R}^4$ contains the vectorized zeroth, first and second order kernel coefficients respectively, such that:

$$\begin{aligned} \mathbf{u}_n &= \begin{bmatrix} u(n-1) & u(n-2) \end{bmatrix}^T, \\ \mathbf{h}_1^T &= \begin{bmatrix} h_1(1) & h_1(2) \end{bmatrix}, \\ \mathbf{h}_2^T &= \begin{bmatrix} h_2(1,1) & h_2(2,1) & h_2(2,1) & h_2(2,2) \end{bmatrix}. \end{aligned}$$

The Kronecker product in eq. (2.3) represents the multiplication of each element of the first matrix with each element of the second matrix. In the current example, this product yields:

$$\mathbf{u}_n \otimes \mathbf{u}_n = \begin{bmatrix} u(n-1)^2 & u(n-1)u(n-2) & u(n-2)u(n-1) & u(n-2)^2 \end{bmatrix}. \tag{2.4}$$

Note that the kernel coefficient $h_2(2,1)$ is repeated in the Volterra Kernel due to the symmetry of the product, i.e. $u(n-1)u(n-2) = u(n-2)u(n-1)$. This approach leads to the general linear model structure for a second degree Volterra series with lag 2:

$$q(n) = \begin{bmatrix} h_0 & \mathbf{h}_1^T & \mathbf{h}_2^T \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{u}_n \\ \mathbf{u}_n \otimes \mathbf{u}_n \end{bmatrix}. \tag{2.5}$$

The approach of example 1 can be extended for D-th degree Volterra series with memory L, such that eq. (2.1) can be written as:

$$q(n) = \begin{bmatrix} h_0 & \mathbf{h}_1^T & \dots & \mathbf{h}_D^T \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{u}_n \\ \vdots \\ \mathbf{u}_n \otimes^D \mathbf{u}_n \end{bmatrix} = \mathbf{H}^T \mathbf{U}_n \qquad \begin{matrix} \mathbf{H} \in \mathbb{R}^{1+L+\dots+L^D} \\ \mathbf{U}_n \in \mathbb{R}^{1+L+\dots+L^D}. \end{matrix} \tag{2.6}$$

Here, $\mathbf{u}_n \otimes^D \mathbf{u}_n$ denotes the D-th order Kronecker product, i.e. $\mathbf{u}_n \otimes^D \mathbf{u}_n = \mathbf{u}_n \otimes^1 \mathbf{u}_n \otimes^2 \dots \otimes^{D-1} \mathbf{u}_n$. Furthermore, $\mathbf{h}_d$ denotes each d-th order vectorized Volterra kernel and $\mathbf{u}_n \in \mathbb{R}^L$ is defined as:

$$\mathbf{u}_n = \begin{bmatrix} u(n-1) & u(n-2) & \dots & u(n-L) \end{bmatrix}.$$

Subsequently, duplicate columns in $\mathbf{H}^T$ corresponding to duplicate rows in $\mathbf{U}_n$ are being removed in order to reduce the computational effort of the simulations, leaving only unique Volterra kernel parameters. The number of unique parameters n_{h_d} in each d-th order Volterra kernel can be calculated via [6]:

$$n_{h_d} = \begin{cases} \left(\frac{1}{d!}\right) \prod_{i=0}^{d-1} (L+1-i), & \text{if} \quad L > 1 \\ 1, & L = 1 \end{cases}. \tag{2.7}$$

Equation (2.7) is essentially a binomial coefficient. This coefficient describes the number of ways of picking unordered outcomes from possibilities in a list. In Volterra sense, this list is defined by $\mathbf{u}_n$. Having said this, the number of unique parameters in each Volterra kernel in eq. (2.7) can alternatively be written as:

$$n_{h_d} = \begin{cases} \binom{L+d-1}{d} = \frac{(L+d-1)!}{d!(L-1)!} & L > 1 \\ 1 & L = 1 \end{cases}. \tag{2.8}$$

Consequently, the total number of unique parameters is defined as $n_H = 1 + n_{h_1} + n_{h_2} + \dots + n_{h_D}$.

Figure 2.1 illustrates the growth of the unique parameters among the first, second, third and fourth order Volterra series. In the figure, both axes are shown logarithmic. The number of parameters involved in the first order Volterra kernel $h_1(\tau_1)$ (blue line) grows linearly with the lag, hence showing a straight line. Knowing this, it can be seen that the number of parameters of the higher order Volterra kernels grow exponentially with the lag, where with increased order the growth of the parameters increases more exponentially. This *curse of dimensionality* requires more attention when it comes to modeling Volterra series of a higher degree, since, depending on the available data size, it may happen that the number of unique parameters to be determined exceeds the number of observations. This will lead to an ill-posed problem with infinitely many solutions. In this situation, there is no proper method in distinguishing the quality of every solution, which means that no judgement can be made about which solution fits the data the best. Fortunately, Bayesian Inference offers a method to incorporate prior knowledge, which mitigates this issue. The concept of prior knowledge is further explained in section 2.2.1

Finally, a stochastic variable ϵ is included in the model structure which represents the error, due to model discrepancy and noise corruption. This results in the following model structure:

$$y(n) = q(n) + \epsilon(n). \tag{2.9}$$

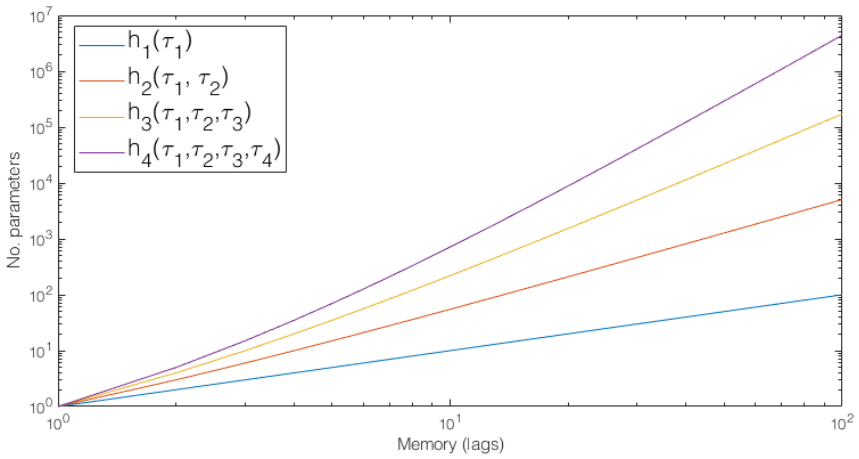


Figure 2.1: The number of unique parameters up to the fourth order Volterra series

Here, at time instance n, $y(n) \in \mathbb{R}$ denotes the measured output and $q(n) \in \mathbb{R}$ is the modeled output, i.e. the output which follows from eq. (2.6). Combining eq. (2.6) with eq. (2.9), the standard linear equation for SISO systems can be derived as

$$\mathbf{Y}_N = \mathbf{U}^T\mathbf{H} + \epsilon \tag{2.10}$$

$$\begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} \mathbf{U}_1^T \\ \mathbf{U}_2^T \\ \vdots \\ \mathbf{U}_N^T \end{bmatrix} \mathbf{H} + \begin{bmatrix} \epsilon(1) \\ \epsilon(2) \\ \vdots \\ \epsilon(N) \end{bmatrix}, \tag{2.11}$$

where the vector $\mathbf{Y}_N \in \mathbb{R}^N$ incorporates the N observed output samples and $\mathbf{U} \in \mathbb{R}^{n_H \times N}$ is the regression matrix with its entries given by (the multiplication of) lagged inputs. Furthermore, this report uses the general notation $\mathcal{M}_{D,L}$ to describe the Volterra Series model class defined by the degree D and lag L.

2.2. Bayesian Inference

Conventional system design techniques usually use a single mathematical model to predict the dynamic response of a system. However, no model is expected to return the perfect predictions for the following reasons. To begin with, the computer model is commonly an approximate representation of the real system behaviour due to complex dynamics. Moreover, since the measurements of the real system are often corrupted with noise, it ensures uncertainty regarding the identified parameters involved in the mathematical model. This motivates the explicit quantification of modeling uncertainty in the response predictions. Bayesian Inference has already been successfully applied in solving inverse problems in EEG applications [19, 33] and has also proven its worth in different fields, such as image processing [31] and a variety of simulation models [4, 13]. So far, Bayesian Inference has not been applied to the system identification of Volterra series.

Bayesian Inference is a mathematical framework that updates hypotheses in a probabilistic manner as more information becomes available. The method relies on the theory of stochastic dynamics, however this does not mean that the world is seen as a stochastic changing nature. The probabilities quantify a degree of belief about different hypotheses. The framework underlying Bayesian Inference is Bayes' rule, which is defined as:

$$p(\mathcal{H}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathcal{H})\,p(\mathcal{H})}{p(\mathcal{D})}. \tag{2.12}$$

Here, $p(\mathcal{H})$ denotes the prior Probability Density Function (PDF) regarding the hypotheses $\mathcal{H}$ and $p(\mathcal{D}|\mathcal{H})$ expresses the likelihood of observing the information $\mathcal{D}$ conditionally on the hypotheses. Furthermore, $p(\mathcal{H}|\mathcal{D})$ describes the posterior PDF and $p(\mathcal{D})$ is the normalization constant.

Bayesian inference not only provides a method to determine parameters, it also equips a mechanism for comparing models and making predictions. This approach has been summarized by Mackay [20] in the following three-level process.

Level 1. Given the available measurements and the defined model class, the posterior PDF of the parameters is found according to Bayes' rule.

Level 2. Again, according to Bayes' rule, different alternative model classes are compared based on the available measurements.

Level 3. The uncertainty is taken into account when subsequent predictions are made, which is obtained by summing over the model class set each weighted by its level of significance.

The three levels of Bayesian Inference are discussed in sequence in sections 2.2.1 to 2.2.3.

2.2.1. Parameter Estimation

The first level of inference aims at finding the posterior parameter PDF. Doing so, the following points should be kept in mind.

1. The product of two Gaussian PDFs yields in another Gaussian PDF [20].
2. The evidence in eq. (2.12), i.e. $p(\mathcal{D})$, is a scaling factor, hence it does not alter the statistical properties of the numerator.

Having said this, it is assumed in this study that both the likelihood as well as the prior are Gaussian distributed. By making this assumption, one can find the statistical properties of the posterior parameter PDF by calculating the properties of the numerator in eq. (2.12). The statistical properties of the latter, such as the mean and the variance, do not change when the distribution scales, which matches therefore the statistical properties of the posterior parameter PDF. In the following sections, the definitions of the Likelihood, Prior, and Evidence are further explained. Finally, the last section provides the calculation for the posterior parameter PDF.

The Likelihood

The Likelihood expresses the probability of observing the measurements $\mathbf{Y}_N$, given the input sequence $\mathbf{U}_N$ and the chosen model structure $\mathcal{M}_{D,L}$ including the parameters $\mathbf{H}$. Mathematically, this is denoted as:

$$\text{Likelihood} = p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathbf{H} \cap \mathcal{M}_{D,L}).$$

Defining the Likelihood requires making assumptions regarding the statistical properties of the prediction-error term ϵ in eq. (2.10). In this study it is assumed that ϵ at time instance n is described by a zero mean Gaussian distribution with a variance σ^2. Therefore, the Likelihood of observing the data $\mathbf{Y}_N$ conditional on the input data $\mathbf{U}_N$, parameter vector $\mathbf{H}$ and the model structure $\mathcal{M}_{D,L}$ is defined as:

$$\begin{aligned} p\left(y(n)|u(n) \cap \mathbf{H} \cap \mathcal{M}_{D,L}\right) &\sim \mathcal{N}\left(\mathbf{U}_n^T\mathbf{H}, \sigma^2\right) \\ &= \frac{1}{\sqrt{(2\pi)}\sigma} \exp\left(-\frac{1}{2\sigma^2}\left(y(n) - \mathbf{U}_n^T\mathbf{H}\right)^T\left(y(n) - \mathbf{U}_n^T\mathbf{H}\right)\right). \end{aligned} \tag{2.13}$$

Assuming that each observation is identically distributed and mutually independent, the Likelihood for a full data sequence is written as as a multivariate Gaussian distribution, i.e.:

$$\begin{aligned} p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathbf{H} \cap \mathcal{M}_{D,L}) &= \prod_{n=1}^{N}\left(y(n)|u(n) \cap \mathbf{H} \cap \mathcal{M}_i\right) \\ &= \frac{1}{(2\pi)^{N/2}\sigma^N} \exp\left(-\frac{1}{2\sigma^2}(\mathbf{Y}_N - \mathbf{U}^T\mathbf{H})^T(\mathbf{Y}_N - \mathbf{U}^T\mathbf{H})\right). \end{aligned} \tag{2.14}$$

This method assumes that the noise variance is constant throughout the data sequence (i.e. $\epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I}_N)$, where $\mathbf{I}_N$ is the N-dimensional identity matrix). The general Likelihood for $\epsilon \sim \mathcal{N}(0, \Sigma_\epsilon)$ is:

$$\begin{aligned} p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathbf{H} \cap \mathcal{M}_{D,L}) &\sim \mathcal{N}(\mathbf{U}^T\mathbf{H}, \Sigma_\epsilon) \\ &= \frac{1}{(2\pi)^{N/2}|\Sigma_\epsilon|^{N/2}} \exp\left(-\frac{1}{2}(\mathbf{Y}_N - \mathbf{U}^T\mathbf{H})^T\Sigma_\epsilon^{-1}(\mathbf{Y}_N - \mathbf{U}^T\mathbf{H})\right). \end{aligned} \tag{2.15}$$

Here $|\cdot|$ represents the determinant function.

The Prior

The Prior express the prior belief regarding the parameters of interest $\mathbf{H}$ given the chosen model structure $\mathcal{M}_{D,L}$. In mathematical notation, the Prior is described as:

$$\text{Prior} = p(\mathbf{H}|\mathcal{M}_{D,L}).$$

Note that in this notation the assumptions are made without observing any data $\mathbf{Y}_N$ and $\mathbf{U}_N$. Therefore, the Prior sums up the available knowledge regarding the parameters of interest $\mathbf{H}$ *a priori*. Since it assumed that the Likelihood in eq. (2.15) is a Gaussian distribution and knowing that the product of two Gaussian distributions yields a Gaussian distribution, it is assumed that the Prior is as well a Gaussian distribution with an arbitrary mean μ_p and variance Σ_p, mathematically described as:

$$p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(\mu_p, \Sigma_p\right). \tag{2.16}$$

The following sections go into more detail about designing the mean μ_p and the variance Σ_p in eq. (2.16).

Imposing prior knowledge The choice of the prior is driven by informativeness. In this study, the terms *informative* and *uninformative* are used to express the amount of information embedded in the prior that is specifically designed for the underlying system. An informative prior contains definite and substantive information regarding the variable of interest, which are substantiated from certain properties of the system. An uninformative prior only expresses general information and is often the first starting point for modeling an arbitrary system. In order to understand the influence of different priors, this study will test on both an informative and an uninformative prior.

The uninformative prior is described by a zero-mean Gaussian distribution with finite variance $\Sigma_p = \alpha \mathbf{I}_{n_H}$, i.e.

$$p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(\mathbf{0}, \alpha \mathbf{I}_{n_H}\right). \tag{2.17}$$

Restricting this distribution to a zero mean inherently means that the parameters are presumed to be around 0 *a priori*, while the choice of the covariance matrix then describes the degree of (un)certainty of this assumption. The prior mean is set to zero and if the observations provide sufficient amount of information, the posterior parameter PDF is set to a different value. However, the more certainty is expressed in advance, the more information the observations should contain [28]. The scaling constant α is used to describe this prior certainty and is yet to be determined. Furthermore, $\mathbf{I}_{n_H}$ denotes the n_H dimensional identity matrix.

However, by describing the covariance matrix with an identity matrix, it is assumed that there is no mutual correlation between the parameters. In addition, one is also limited in imposing prior information on specific parameters, since the variance is by definition equal for all the parameters. The informative prior is described in such a way to not only distinguish between parameters but also to describe correlations between them. This approach follows the procedure introduced by Birpoutsoukis et al. [6], who assumed that the Volterra kernels hold the properties of decaying and smooth. The informative prior is defined as a zero mean Gaussian distribution with finite variance $\Sigma_p = \text{blkdiag}(P_0, P_1, \ldots, P_D)$, where P_d denotes the d-th order Volterra kernel variance.

Birpoutsoukis et al. [5, 6] define the covariance matrix of the first kernel ($P_1 = \text{E}[h_1(\tau_1)h_1(\tau_1)^T]$) as follows:

$$P_{1,TC} = c_1 \alpha_1^{\max(i,j)}, \tag{2.18}$$

which is knows as the Tuned-Correlated matrix, also known as the Stable Spline kernel [23, 30]. Here, $c_1 \geq 0$ is a scaling factor and $0 \leq \alpha < 1$ controls the exponential decay.

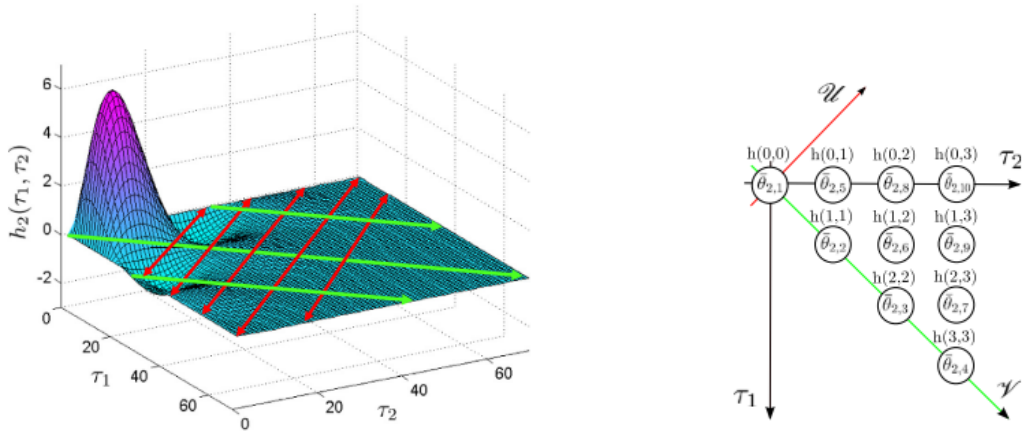
The second order covariance matrix is constructed in a similar fashion. Birpoutsoukis et al. [5] define the following properties

Property 1. The Volterra kernel decays along any direction and neigbouring coefficients are correlated.

Property 2. The second order kernel is symmetric, meaning that $h_2(\tau_1, \tau_2) = h_2(\tau_2, \tau_1) \quad \forall \tau_1, \tau_2$.

Property 3. The covariance matrix P_2 should be constructed to be a symmetric, positive-semidefinite matrix.

To maintain these properties, a rotated coordinate system is introduced as shown in fig. 2.2. Fig. 2.2a illustrates the decaying and smoothness properties along the green and red directions, which correspond to the directions of the new coordinate system. Figure 2.2b shows the top view of a part of the Volterra kernel.



(a) 3D view of a second order Volterra Kernel

(b) Zoomed top view of the Volterra Kernel

Figure 2.2: An example of a second-order Volterra kernel [5] including the rotated coordinate system.

The new coordinate system $\langle \mathcal{V}, \mathcal{U} \rangle$ is rotated 45° with respect to the old coordinate system $\langle \tau_1, \tau_2 \rangle$, which is mathematically described as:

$$\begin{bmatrix} \mathcal{U} \\ \mathcal{V} \end{bmatrix} = \begin{bmatrix} \cos(45°) & -\sin(45°) \\ \sin(45°) & \cos(45°) \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}. \tag{2.19}$$

This new coordinate system is chosen since it makes the properties of exponential decaying and smoothness more understandable. Subsequently, the prior information is imposed along the new axes. The resulting covariance matrix then yields:

$$P_2(i,j) = c_2 \alpha_u^{\max(U_i,U_j)} \alpha_v^{\max(V_i,V_j)}. \tag{2.20}$$

Estimation of the hyperparameters The informative covariance matrix of a 2nd-degree Volterra kernel depends on 6 hyperparameters, e.g. $\theta_{hp} = [c_0, c_1, c_2, \alpha_1, \alpha_u, \alpha_v]$. The hyperparameters define the structure of the prior knowledge, hence determining the values plays a crucial role in the estimation performance of the model. The Maximum Likelihood approach, also known as Empirical Bayes, maximizes the joint density of the output measurements and the impulse response. The estimated hyperparameters are defined as [24],

$$\begin{aligned} \hat{\theta}_{hp} &\triangleq \underset{\theta_{hp}}{\operatorname{argmax}}\, p(\mathbf{Y}_N|\theta_{hp}) \\ &= \underset{\theta_{hp}}{\operatorname{argmin}}\, \mathbf{Y}_N^T \Sigma_Y^{-1} \mathbf{Y}_N + \log|\Sigma_Y|, \end{aligned} \tag{2.21}$$

where $\Sigma_Y = \mathbf{U}^T \mathcal{P}(\theta_{hp}) \mathbf{U} + \sigma^2 I_N$ corresponds to the covariance matrix of the measured data and $|\cdot|$ represents its determinant. Furthermore, $c_0, c_1, c_2 > 0$ and $0 < \alpha_1, \alpha_u, \alpha_v < 1$. I_N denotes the N-dimensional identity matrix and σ^2 is the noise variance. Note that σ^2 is typically not known in advance and thus included in θ_{hp}. Recall that $\mathbf{U} \in \mathbb{R}^{n_H \times N}$ and $\mathcal{P} \in \mathbb{R}^{n_H \times n_H}$. The objective function in equation (2.21) is non-convex in the hyperparameters, therefore a non-linear optimization solver and a multi-start program to avoid local minima are required.

The Evidence

The Evidence in Eq. (2.12) acts as a normalization constant, such that the total probability of the posterior parameter PDF equals one. The Evidence expresses the probability of observing the output $\mathbf{Y}_N$ given the input sequence $\mathbf{U}_N$ and the model class $\mathcal{M}_{D,L}$, mathematically noted as:

$$\text{Evidence} = p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L}).$$

Note that in this notation there is no dependency on the parameters $\mathbf{H}$. The equation for the Evidence is found by marginalizing the product of the Likelihood and the Prior over the parameters $\mathbf{H}$, ensuring that the posterior parameter PDF has a total probability of 1. According to the Total Probability Theorem, this is defined as:

$$p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L}) = \int_{\mathbf{H}} p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathbf{H} \cap \mathcal{M}_{D,L}) p(\mathbf{H}|\mathcal{M}_{D,L}) d\mathbf{H}. \tag{2.22}$$

Here, $\mathbf{H}$ can be of different dimensions per model structure $\mathcal{M}_{D,L}$. The subscripts D, L are omitted since the structure of $\mathbf{H}$ is inherently defined by $\mathcal{M}_{D,L}$. Although the Evidence is a scaling factor and hence does not affect the shape of the distribution function, it will play an important role in the model averaging process, as described in section 2.2.2.

However, since usually $\mathbf{H}$ being a high dimensional vector, the integral in Eq. (2.22) is often intractable, meaning that alternatives methods are needed in order to calculate the integral in Eq. (2.22). Well known estimation methods are Markov Chain Monte Carlo [11, 26, 31] or Laplace's Approximation methods [4], however, provided that the Likelihood and the Prior are Gaussian distribution functions, $p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L})$ can be computed analytically. Consider the general model structure in Eq. (2.10), with $\epsilon \sim \mathcal{N}(0, \Sigma_\epsilon)$, and the posterior distribution given in Eq. (2.16). The statistical properties of $p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L})$ can be found via:

$$p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L}) \sim \mathcal{N}(\mu_{\text{ev}}, \Sigma_{\text{ev}}) \tag{2.23}$$

$$\begin{aligned}\mu_{\text{ev}} &= \mathbb{E}[\mathbf{Y}_N] = \mathbb{E}\left[\mathbf{U}^T\mathbf{H} + \epsilon\right] \\ &= \mathbf{U}^T\mathbb{E}[\mathbf{H}] + \mathbb{E}[\epsilon] \\ &= \mathbf{U}^T\mu_p + 0\end{aligned} \tag{2.24}$$

$$\begin{aligned}\Sigma_{\text{ev}} &= \mathbb{E}\left[(\mathbf{Y}_N - \mathbb{E}[\mathbf{Y}_N])(\mathbf{Y}_N - \mathbb{E}[\mathbf{Y}_N])^T\right] \\ &= \mathbb{E}\left[\left(\mathbf{U}^T\mathbf{H} + \epsilon - \mathbf{U}^T\mu_p\right)\left(\mathbf{U}^T\mathbf{H} + \epsilon - \mathbf{U}^T\mu_p\right)^T\right] \\ &= \mathbf{U}^T\mathbb{E}\left[\mathbf{H}\mathbf{H}^T\right]\mathbf{U} + \cancel{\mathbf{U}^T\mathbb{E}\left[\mathbf{H}\epsilon^T\right]} - \mathbf{U}^T\mathbb{E}[\mathbf{H}]\mu_p^T\mathbf{U} \\ &\quad + \cancel{\mathbb{E}\left[\epsilon\mathbf{H}^T\right]\mathbf{U}} + \mathbb{E}\left[\epsilon\epsilon^T\right] - \cancel{\mathbb{E}[\epsilon]\mu_p^T\mathbf{U}} \\ &\quad - \mathbf{U}^T\mu_p\mathbb{E}\left[\mathbf{H}^T\right]\mathbf{U} - \cancel{\mathbf{U}^T\mu_p\mathbb{E}[\epsilon]} + \mathbf{U}^T\mu_p\mu_p^T\mathbf{U} \\ &= \mathbf{U}^T\left[\Sigma_p + \mu_p\mu_p^T\right]\mathbf{U} - \mathbf{U}^T\mu_p\mu_p^T\mathbf{U} + \Sigma_\epsilon \\ &= \mathbf{U}^T\Sigma_p\mathbf{U} + \Sigma_\epsilon.\end{aligned} \tag{2.25}$$

Here, the following relation is used to decompose $\mathbb{E}\left[\mathbf{H}\mathbf{H}^T\right]$:

$$\text{var}(X, X) = \mathbb{E}[XX] - \mathbb{E}[X]\mathbb{E}[X]. \tag{2.26}$$

Furthermore, it is assumed that the parameters $\mathbf{H}$ and the noise vector ϵ are uncorrelated, i.e. $\mathbb{E}\left[\epsilon\mathbf{H}^T\right] = \mathbb{E}\left[\mathbf{H}\epsilon^T\right] = 0$.

Computing the posterior parameter PDF

So far, the definitions are given for the Likelihood, Prior and Evidence, which are needed to build the equation for the first level of Bayesian Inference. According to Bayes' theorem, the first level is defined as:

$$p(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L}) = \frac{p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathbf{H} \cap \mathcal{M}_{D,L})\, p(\mathbf{H}|\mathcal{M}_{D,L})}{p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L})}. \tag{2.27}$$

However, since it is known that both the Likelihood and the Prior are Gaussian distributions and the Evidence is a scaling factor, the statistical properties of the posterior parameter PDF are found by evaluating the numerator in eq. (2.27). In other words, the posterior parameter PDF is proportional to the numerator in eq. (2.27), i.e.

$$p(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L}) \propto p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathbf{H} \cap \mathcal{M}_{D,L})\, p(\mathbf{H}|\mathcal{M}_{D,L}). \tag{2.28}$$

This offers opportunity to find the statistical properties of the posterior parameter PDF, such as the mean and variance. Since scaling does not affect the shape of the distribution and therefore does not alter the statistical properties of the distribution of interest, knowing the statistical properties of the scaled distribution inherently yields the statistical properties of the posterior distribution. It can be shown that the product of the Likelihood and the Prior, therefore the posterior parameter PDF as well, holds the following properties [27, 33]:

$$\begin{aligned}p(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L}) &\sim \mathcal{N}\left(\mu_{\text{post}}, \Sigma_{\text{post}}\right) \\ \mu_{\text{post}} &= \left(\Sigma_p^{-1} + \mathbf{U}\Sigma_\epsilon^{-1}\mathbf{U}^T\right)^{-1}\left(\mathbf{U}\Sigma_\epsilon^{-1}\mathbf{Y}_N + \Sigma_p^{-1}\mu_p\right). \\ \Sigma_{\text{post}} &= \left(\Sigma_p^{-1} + \mathbf{U}\Sigma_\epsilon^{-1}\mathbf{U}^T\right)^{-1}\end{aligned} \tag{2.29}$$

Equation (2.29) allows us to understand the meaning of an informative prior and informative training data. First, Σ_{post} is positively correlated with both Σ_p and Σ_ϵ, meaning that Σ_{post} will increase with the prior uncertainty (variance) and the measurement noise variance. Second, on the other hand, Σ_{post} is inversely correlated with $\mathbf{U}\mathbf{U}^T$. This means that if the regression matrix $\mathbf{U}$ contains little information, therefore $\mathbf{U}\mathbf{U}^T$ as well, the posterior distribution in eq. (2.29) becomes more dependent on the specific choice of the prior. In other words, as well described by Ljung [18], Verhaegen [34], the input signal $\mathbf{U}_N$ should contain enough information such that the regression matrix $\mathbf{U}$ is full rank. This mathematical notion is well known as *persistency of excitation*. Similarly, in situations where the output is highly corrupted with noise, $\mathbf{U}\Sigma_\epsilon^{-1}\mathbf{U}^T$ becomes small and forces Σ_{post} to be relatively large on its own. To prevent this, the prior needs to contain enough information such that Σ_p small enough to compensate for the high noise environment and to keep the variance on the parameters relatively low.

2.2.2. Model Comparison

This section provides the second level of Bayesian Inference, which compares different alternative model classes based on the available data. Suppose a model class set $\mathbf{M}$ contains n_m different model classes. Again, by using Bayes' Theorem, the probability of an arbitrary model class $\mathcal{M}_{D_i,L_i}$ conditionally on the available data can be computed via:

$$p[\mathcal{M}_{D_i,L_i}|\mathbf{Y}_N \cap \mathbf{U}_N] = \frac{p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D_i,L_i})\, p[\mathcal{M}_{D_i,L_i}]}{p(\mathbf{Y}_N|\mathbf{U}_N)}. \tag{2.30}$$

Here, $p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D_i,L_i})$ denotes the evidence as described in equation (2.27), which is weighted by the prior probability $p[\mathcal{M}_{D_i,L_i}]$. The latter is a pre-assigned probability for the model class with which preferences can be expressed for certain models structures, without involving the data. Subsequently, the probability in eq. (2.30) can be used to calculate the Bayes' factor, which quantifies the support for a model $\mathcal{M}_{D_i,L_i}$ over another model $\mathcal{M}_{D_j,L_j}$. The Bayes' factor is given as:

$$B_{i,j} = \frac{p[\mathcal{M}_{D_i,L_i}|\mathcal{D}]}{p[\mathcal{M}_{D_j,L_j}|\mathcal{D}]} = \frac{p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D_i,L_i})\, p[\mathcal{M}_{D_i,L_i}]}{p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D_j,L_j})\, p[\mathcal{M}_{D_j,L_j}]}. \tag{2.31}$$

Here, $p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L})$ represents the evidence of eqs. (2.23) to (2.25) evaluated at $\mathbf{Y}_N$. Regarding the prior model probability $p\left[\mathcal{M}_{D_i,L_i}\right]$, a reasonable approach would be to consider all model classes equally plausible *a priori*, i.e.,

$$p\left[\mathcal{M}_{D,L}\right] = \frac{1}{n_m} \quad \forall\ D \in \mathbf{D}\ ,\ L \in \mathbf{L}. \tag{2.32}$$

Consequently, the probability of a model in the prior MCD is dominated by the evidence, which is given in eqs. (2.23) to (2.25).

The posterior distribution of $\mathcal{M}_{D_i,L_i}$ conditionally on the data $\mathcal{D}$ and the model class set $\mathbf{M}$ is found by calculating the Bayes' factor with respect to $\mathcal{M}_0$ and marginalizing over the model class set. This is formally defined as [33]:

$$p[\mathcal{M}_{D_i,L_i}|\mathcal{D} \cap \mathbf{M}] = \frac{B_{i,0}}{\sum_{m=0}^{n_m} B_{m,0}}. \tag{2.33}$$

It is worth noting that the model comparison step is a prior act, which does not require any posterior distributions of the parameters. This can be deduced from the calculation of the evidence in eq. (2.23), which is only dependent on the prior distribution $p(\mathbf{H}|\mathcal{M}_{D,L})$, the noise distribution Σ_ϵ and the regression matrix $\mathbf{U}$. For that reason, the distribution in eq. (2.33) is referred to as the prior Model Class Distribution (MCD).

Furthermore, from a computational point of view, it would be advantageous to perform model comparison first before computing any posterior parameter PDFs, so that the candidate models that have no added value can be excluded from the model class set $\mathbf{M}$. However, in this report the performance of the complete model class set are examined during both modeling and validation, hence no model class is left out.

2.2.3. Predictive Analysis

Bayesian Inference differentiates itself from traditional methods in making subsequent predictions regarding some arbitrary quantity $\mathbf{y}$. In conventional methods, a parameter class is tested and only one parameter vector $\mathbf{H}$ is accepted at some level of significance. However, this approach ignores model uncertainty, which may lead to over-confident decisions. Bayesian predictive analysis incorporates this probabilistic information in

the Robust Predictive PDF, which is known as Posterior Robust Predictive Analysis. In addition to the best estimates (the mean of the PDF), this distribution also offers the uncertainty intervals of the predictions of the system. This is formed by the parameter uncertainty and the noise corruption, which corresponds to the variance of the Robust Predictive PDF.

However, this approach is implied by a single model class $\mathcal{M}_{D,L} \in \mathbf{M}$. Using similar reasoning, rather than picking one single model class, Bayesian Hyper Robust Predictive Analysis obtains predictions by summing over the complete model set $\mathbf{M}$ weighted by each model class probability. By doing this, the means and variances of all Robust Predictive PDFs are weighted by its level of significance and are merged together into one robust predictive PDF, namely the Hyper Robust Predictive (HRP) PDF [3].

Robust Predictive Analysis

Given a model class $\mathcal{M}_{D,L}$, one is interested in estimating the quantity y_{N+1}, which is defined as:

$$y_{N+1} = (\mathbf{U}_{N+1})^T \mathbf{H} + \epsilon(N+1). \tag{2.34}$$

Similarly as in eqs. (2.23) to (2.25), the distribution of y_{N+1}, i.e. $p(y_{N+1}|\mathcal{D} \cap \mathcal{M}_{D,L})$ can be calculated analytically (given that the prior and likelihood are both Gaussian distributions). Gaussian Prior and Likelihood consequently means that the posterior of $\mathbf{H}$ is normally distributed. This allows one to rewrite $p(y_{N+1}|\mathcal{D} \cap \mathcal{M}_{D,L})$ as a sum of Gaussian distributions. Consider the following posterior parameter PDF and noise distribution:

$$\begin{aligned} p(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L}) &\sim \mathcal{N}(\mu_{\text{post}}, \Sigma_{\text{post}}) \\ \epsilon(n) &\sim \mathcal{N}(0, \sigma_\epsilon^2) \quad \forall n. \end{aligned}$$

Here it is assumed that the noise distribution is stationary and therefore constant throughout all time instances. Following the derivation in eqs. (2.23) to (2.25), the Robust Predictive PDF yields:

$$\begin{aligned} p(y_{N+1}|\mathcal{D} \cap \mathcal{M}_i) &\sim \mathcal{N}(\mu_y, \sigma_y) \\ \mu_y &= (\mathbf{U}_{N+1})^T \mu_{\text{post}} \\ \sigma_y &= \underbrace{(\mathbf{U}_{N+1})^T \Sigma_{\text{post}} (\mathbf{U}_{N+1})}_{\text{Parameter Uncertainty}} + \underbrace{\sigma_\epsilon^2}_{\text{Noise Corruption}}. \end{aligned} \tag{2.35}$$

Equation (2.35) shows that both the parameter uncertainty and the noise corruption have an effect on the variance of the Robust Predictive PDF.

Hyper Robust Predictive Analysis

Hyper Robust Predictive Analysis provides a systematic mechanism to combine different model classes in making predictions. This method includes model uncertainty and it is demonstrated by, among others, Hoeting et al. [15], Raftery et al. [25] that this approach may result in a decrease in prediction error.

Consider the competitive model class set

$$\mathcal{M}_{D,L} = \{V(D,L) : D \in \mathbf{D}, L \in \mathbf{L}\}, \tag{2.36}$$

where $V(D,L)$ denotes the Volterra model structure as described in section 2.1. Furthermore, the first N observations are stored in $\mathcal{D}$, i.e. $\mathcal{D} = (\mathbf{Y}_N, \mathbf{U}_N)$. The posterior PDF is found by using the Total Probability Theorem, i.e.,

$$p(y_{\text{HRP}}|\mathcal{D} \cap \mathbf{M}) = \sum_{D,L} p(y_{N+1}|\mathcal{D} \cap \mathcal{M}_{D,L}) p[\mathcal{M}_{D,L}|\mathcal{D} \cap \mathbf{M}]. \tag{2.37}$$

Here, each Robust Predictive PDF is weighted by $p[\mathcal{M}_{D,L}|\mathcal{D} \cap \mathbf{M}]$ from Eq. (2.33), which represents the relative level of significance per model class. The Robust Predictive PDF $p(y_{N+1}|\mathcal{D} \cap \mathcal{M}_{D,L})$ is found using eq. (2.35). This method was verified by Trujillo-Barreto et al. [33], who applied the Bayesian formulation to the inverse problem of finding the posterior current density in the brain to locate brain activity. The following example helps in finding the generic expression for eq. (2.37)

Example 2. *Consider the case with two model classes as described in Sec. 2.2.1 for y_{N+1}:*

$$\mathcal{M}_1 : y_1 = \mathbf{U}_1^T \mathbf{H}_1 + \epsilon \tag{2.38}$$

$$\mathcal{M}_2 : y_2 = \mathbf{U}_2^T \mathbf{H}_2 + \epsilon, \tag{2.39}$$

with $\epsilon \sim \mathcal{N}(0,\sigma^2)$ *considered to be stationary,* $\mathbf{H}_1 \sim \mathcal{N}(\mu_{H1},\Sigma_{H1})$ *and* $\mathbf{H}_2 \sim \mathcal{N}(\mu_{H2},\Sigma_{H2})$*. For the ease of notation, the subscript* $N+1$ *is omitted in* y_i *and* $\mathbf{U}_i$*. To clarify,* $y_1 = y_{N+1}$ *and* $\mathbf{U}_1^T = \mathbf{U}_{N+1}^T$ *defined by model class* $\mathcal{M}_{D_1,L_1}$*. Equivalently,* $y_2 = y_{N+1}$ *and* $\mathbf{U}_2^T = \mathbf{U}_{N+1}^T$ *defined by model class* $\mathcal{M}_{D_2,L_2}$*. The HRP PDF in eq. (2.37) can then be written as:*

$$p(y_{HRP}|\mathcal{D}\cap\mathbf{M}) = p(y_1|\mathcal{D}\cap\mathcal{M}_{D_1,L_1})\underbrace{p[\mathcal{M}_{D_1,L_1}|\mathcal{D}\cap\mathbf{M}]}_{\beta_1} + p(y_2|\mathcal{D}\cap\mathcal{M}_{D_2,L_2})\underbrace{p[\mathcal{M}_{D_2,L_2}|\mathcal{D}\cap\mathbf{M}]}_{\beta_2}, \tag{2.40}$$

such that $\beta_1 = p[\mathcal{M}_{D_1,L_1}|\mathcal{D}\cap\mathbf{M}]$ *and* $\beta_2 = p[\mathcal{M}_{D_2,L_2}|\mathcal{D}\cap\mathbf{M}]$*. Using equivalent reasoning as described in eqs. (2.23) to (2.25), the statistical properties of* $p(y_{HRP}|\mathcal{D}\cap\mathbf{M})$ *can be found via:*

$$\begin{aligned}
p(y_{HRP}|\mathcal{D}\cap\mathbf{M}) &\sim \mathcal{N}(\mu_{HRP},\sigma_{HRP}) \\
\mu_{HRP} &= \mathbb{E}\left[\beta_1\left(\mathbf{U}_1^T\mathbf{H}_1+\epsilon\right)+\beta_2\left(\mathbf{U}_2^T\mathbf{H}_2+\epsilon\right)\right] \\
&= \beta_1\mathbf{U}_1^T\mu_{H1}+\beta_2\mathbf{U}_2^T\mu_{H2} \\
\\
\sigma_{HRP} &= \mathbb{E}\left[\left(\beta_1\left(\mathbf{U}_1^T\mathbf{H}_1+\epsilon\right)+\beta_2\left(\mathbf{U}_2^T\mathbf{H}_2+\epsilon\right)-\beta_1\mathbf{U}_1^T\mu_{H1}-\beta_2\mathbf{U}_2^T\mu_{H2}\right)(\cdots)^T\right] \\
&= \beta_1^2\mathbf{U}_1^T\mathbb{E}\left[\mathbf{H}_1\mathbf{H}_1^T\right]\mathbf{U}_1+\beta_2^2\mathbf{U}_2^T\mathbb{E}\left[\mathbf{H}_2\mathbf{H}_2^T\right]\mathbf{U}_2-\beta_1^2\mathbf{U}_1^T\mathbb{E}[\mathbf{H}_1]\mu_{H1}^T\mathbf{U}_1 \\
&\quad -\beta_2^2\mathbf{U}_2^T\mathbb{E}[\mathbf{H}_2]\mu_{H2}^T\mathbf{U}_2+\underbrace{(\beta_1+\beta_2)^2}_{=1}\mathbb{E}[\epsilon\epsilon^T] \\
&= \beta_1^2\mathbf{U}_1^T\left[\Sigma_{H1}+\cancel{\mu_{H1}\mu_{H1}^T}\right]\mathbf{U}_1+\beta_2^2\mathbf{U}_2^T\left[\Sigma_{H2}+\cancel{\mu_{H2}\mu_{H2}^T}\right]\mathbf{U}_2 \\
&\quad -\cancel{\beta_1^2\mathbf{U}_1^T\mu_{H1}\mu_{H1}^T\mathbf{U}_1}-\cancel{\beta_2^2\mathbf{U}_2^T\mu_{H2}\mu_{H2}^T\mathbf{U}_2}+\sigma_\epsilon^2 \\
&= \beta_1^2\mathbf{U}_1^T\Sigma_{H1}\mathbf{U}_1+\beta_2^2\mathbf{U}_2^T\Sigma_{H2}\mathbf{U}_2+\sigma_\epsilon^2.
\end{aligned} \tag{2.41}$$

Here, it is assumed that $\mathbb{E}[\mathbf{H}_1\mathbf{H}_2^T] = \mathbb{E}[\mathbf{H}_1]\mathbb{E}[\mathbf{H}_2^T]$ *(independence). This assumption is not substantiated with representative literature, however this simplifies the expression for the HRP PDF. This is discussed further in section 5.1.1.*

Having said this, the generic expression for HRP using Volterra Series is defined as:

$$\begin{aligned}
p(y_{\text{HRP}}|\mathcal{D}\cap\mathbf{M}) &\sim \mathcal{N}\left(\mu_{\text{HRP}},\sigma_{\text{HRP}}\right) \\
\mu_{\text{HRP}} &= \sum_{D,L}\left(p\left[\mathcal{M}_{D,L}|\mathcal{D}\cap\mathbf{M}\right]\mathbf{U}_{N+1}^T\mu_{\text{post}}\right) \\
\sigma_{\text{HRP}} &= \underbrace{\sum_{D,L}\left(p\left[\mathcal{M}_{D,L}|\mathcal{D}\cap\mathbf{M}\right]^2(\mathbf{U}_{N+1})^T\Sigma_{\text{post}}(\mathbf{U}_{N+1})\right)}_{\text{Parameter uncertainty}} + \underbrace{\sigma_\epsilon^2}_{\text{Noise Corruption}}
\end{aligned}. \tag{2.42}$$

Here, the subscripts D,L are omitted in $\mathbf{U}_{N+1}$, μ_{post} and Σ_{post}, since its shape and values are inherently defined by $\mathcal{M}_{D,L}$ in $p\left[\mathcal{M}_{D,L}|\mathcal{D}\cap\mathbf{M}\right]$. Furthermore, μ_{post} and Σ_{post} describe the mean and variance respectively of the posterior parameter PDF in eq. (2.29). Equivalently as shown in the Robust Predictive PDF in eq. (2.37), from eq. (2.42) it can be deduced that the variance of the HRP PDF is driven by two terms. On the one hand the parameter uncertainty involved each model class and on the other hand the uncertainty caused by noise corruption.

Note that in eq. (2.42) the variance of the noise corruption σ_ϵ is considered to be stationary among different model classes. In practice, this value might be found via the non-linear optimization as discussed in section 2.2.1, yielding in a different value per model class. In this situation, eq. (2.42) displays a simplified representation.

2.3. Model Evaluation

During this study a variety of techniques will be used to evaluate the output of the system or to evaluate the performance of the modeled system. First, the Signal-to-Noise Ratio (SNR) is elaborated. Second, the conventional methods used in existing studies to evaluate the performance of a system is explained, including the Root Mean Squared Error (RMSE) and the Variance Accounted For (VAF). Finally, the posterior MCD is disclosed, which is used to evaluate the validation of the Predictive PDFs.

Signal-to-Noise ratio
The Signal-to-Noise Ratio describes a ratio of the power in the ground truth signal with respect to the added noise in decibel. This value can therefore only be calculated for the computer models, since the true noiseless signal is not known for the cortical response data. The SNR is mathematically described as:

$$\text{SNR} = 20\log_{10}\left(\frac{\|\mathbf{Y}_{\text{N,true}}\|_2}{\|\epsilon_N\|_2}\right). \tag{2.43}$$

Here, ϵ_N denotes the noise sequence for N time instances en $\|\cdot\|_2$ denotes the 2-norm.

Root-Mean-Squared-Error
The Root-Mean-Squared-Error describes a measure of the difference between an estimator and the true values. It represents the root of the quadratic mean of these differences. The RMSE is mathematically described as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_n \left(\hat{y}(n) - y_{\text{true}}(n)\right)^2}. \tag{2.44}$$

Here, $\hat{y}(n)$ denotes the modeled output at time instance n and $y_{\text{true}}(n)$ is its corresponding true value. Furthermore, N represents the sample size.

Variance-Accounted-For
The Variance-Accounted-For is used to evaluate the performance of a model by comparing the variance of the output of the modeled system and the observations. The output is a percentage of the similarity in variance. The VAF is mathematically described as:

$$\text{VAF} = 100\% \cdot \left(1 - \frac{\text{var}(\hat{y} - y_{\text{true}})}{\text{var}(y_{\text{true}})}\right). \tag{2.45}$$

Here, $\text{var}(\hat{y} - y_{\text{true}})$ represents the variance of the difference between the modeled output and the true output. Thus, having a constant difference yields a VAF of 100%.

Posterior Model Class Distribution
The posterior Model Class Distribution evaluates the performance of a single model class with respect to the competitive model class set. The evaluation of a single model class is based on the Posterior Likelihood Fit (PLF), which is described as:

$$p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L})_{\text{PLF}} = \int_{\mathbf{H}} p\left(\mathbf{Y}_N|\mathbf{U}_N \cap \mathbf{H} \cap \mathcal{M}_{D,L}\right) p\left(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L}\right) d\mathbf{H}. \tag{2.46}$$

Subsequently, the statistical properties of the are found in a similar fashion as described in eq. (2.23), which yields:

$$\begin{aligned} p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L})_{\text{PLF}} &\sim \mathcal{N}\left(\mu_{\text{PLF}}, \Sigma_{\text{PLF}}\right) \\ \mu_{\text{PLF}} &= \mathbf{U}^T \mu_{\text{post}} \\ \Sigma_{\text{PLF}} &= \mathbf{U}^T \Sigma_{\text{post}} \mathbf{U} + \Sigma_\epsilon. \end{aligned} \tag{2.47}$$

By evaluating eq. (2.47) $\mathbf{Y}_N$ one finds a probability of observing the measurements given the model structure and the posterior parameter PDFs. Logically, the greater the probability is, the better the proposed model class fits the ground truth model, however this number itself does not say much since there is no maximum which represents a perfect data fit (such as 100% VAF or 0 RMSE). In addition, the probability not only depends on the average fit, but also on the length of the sample size. For this reason, the relative probability between model classes in the model class set is found by computing the posterior Model Class Distribution, which follows the procedure as described in section 2.2.2.

Doing so, however, one should be aware that no claim is being made regarding the performance of the model with the highest probability in the posterior MCD with respect to the true underlying system. The only statement made is that the relative performance of the model class with respect to the competitive model class.

3

Computer Simulations

This chapter is devoted to the implementation of the discussed theory on two types of computer models, namely the Volterra system and the Neural Network. First, section 3.1 discusses the different experimental setups and introduces the two different ground truth models. Second, the modeling approach is explained in section 3.2. Section 3.3 presents the acquired results of the discussed theory on the two different ground truth models. Finally, section 3.4 concludes this chapter.

3.1. Experimental Setups

This section provides an overview of the experimental setups used for the computer simulations. First, section 3.1.1 elaborates the Volterra ground thruth model structure. Second, the Neural Network ground truth system is explained in section 3.1.2.

3.1.1. Volterra

In this experiment, the ground truth system is modeled as a second degree Volterra series with lag 20 ($D = 2$, $L = 20$ in eq. (2.6)). The kernel parameters, i.e. h_0, $h_1(\tau_1)$, $h_2(\tau_1, \tau_2)$ are drawn from a zero mean Gaussian distribution with finite variance Σ. Here, Σ is constructed according to the theory proposed by Birpoutsoukis et al. [6], as discussed in section 2.2.1. The corresponding hyperparameters are shown in table 3.1.

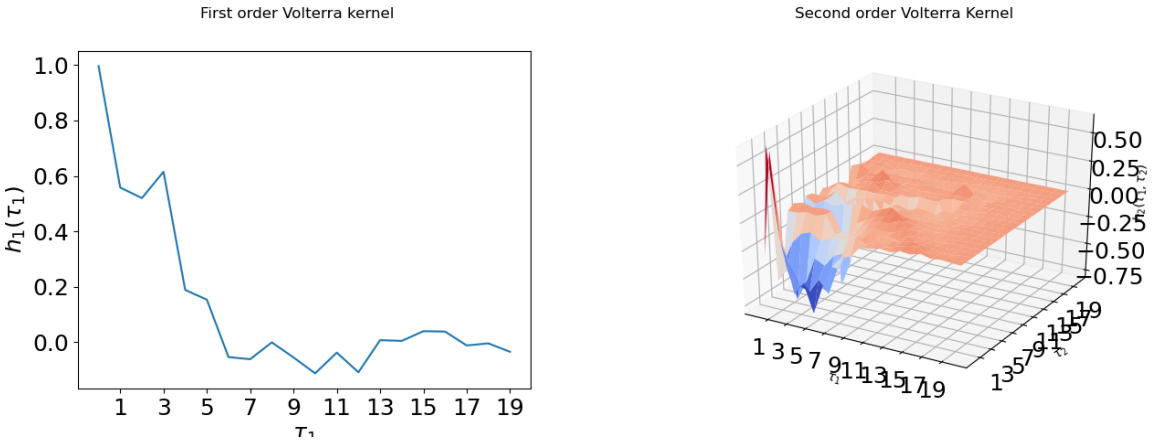
Table 3.1: Volterra ground truth: Hyperparameters used to construct the prior variance matrix

θ_{hp}	Value
c_0	0.95
c_1	0.08
α_1	0.32
c_2	0.59
α_u	0.72
α_v	0.65

The resulting Volterra kernels are illustrated in fig. 3.1. Here, fig. 3.1a illustrates the first order Volterra kernel. The x-axis denotes the lag variable τ_1 and the y-axis depicts the corresponding value of the kernel. Figure 3.1b illustrates the second order kernel. Here, the x-axis and y-axis denote the lag variables τ_1 and τ_2 respectively and the z-axis denotes the corresponding Volterra kernel value. Furthermore, the zeroth order Volterra kernel is drawn from the distribution $\mathcal{N}(0, 0.95)$. The output of the Volterra system Y_{true} is exposed to mild noise circumstances, such that $\epsilon \sim \mathcal{N}(0, \mathbf{I}_N)$. This value is chosen so that the SNR is around 20 dB. This results in the observation vector $\mathbf{Y}_N$.

This experiment has the following objectives:

- Given that the ground truth model structure is included in the competitive model candidate set, the first objective is to test the Bayesian Inference algorithm whether it is able to reconstruct the ground truth model.



(a) First order Volterra system

(b) Second order Volterra system

Figure 3.1: Ground truth Volterra model. Left: $h_1(\tau_1)$, right: $h_2(\tau_1,\tau_2)$.

- As shown in eq. (2.27), in low SNR systems the posterior distribution becomes more dependent on the specific choice of the prior. The second objective is to expose the contribution of an informative prior on the performance of the model averaging process in low SNR systems with respect to an uninformative prior.

3.1.2. Noisy Neural Network

In this experiment, the ground truth system is modeled as a Neural Network as illustrated in fig. 3.2. The neural network contains of one input layer of size 11, one hidden layer of size 10 and one output layer of size 1. The nodes of the layers all have a linear activation function and the weight matrices M_1 and M_2 include a bias B_1 and B_2, of which its corresponding values are uniformly drawn from $\mathcal{U}(0,5)$. The true output of the system $y(n)_{\text{true}}$ is first exposed to mild noise circumstances, i.e. $\epsilon \sim \mathcal{N}(0,\mathbf{I}_N)$. Subsequently, the understand the influence of noise corruption, in the second part the system is highly corrupted with noise, i.e. $\epsilon \sim \mathcal{N}(0,70\cdot\mathbf{I}_N)$. Doing so, the SNR is around -30dB.

Having said this, the experiment has the following objectives:

- The first goal is to investigate the consequences of the Bayesian Inference using Volterra Series algorithm when applied to a ground truth model that differs from the competitive model class set.
- It is expected that noise corruption will affect the uncertainty margins of the parameters, consequently playing an important role in the model averaging process. The second objective is to understand the effect of high noise environments (low SNR) on performance of the model averaging process.

3.2. Modeling approach

In this section the modeling approach is elaborated. First, the competitive model class set is explained in section 3.2.1. Second, the chosen excitation signal is elaborated in section 3.2.2.

3.2.1. Competitive Model Class Set

To deal with the uncertainty of which model class is most capable to represent the systems dynamic behaviour, a set of competitive model classes is chosen. Here, $\mathbf{M}$ represents a set containing the candidate model classes, which is mathematically defined as:

$$\mathcal{M}_{D,L} = \{V(D,L) : D \in \mathbf{D}, L \in \mathbf{L}\} \in \mathbf{M}. \tag{3.1}$$

Here, $V(D,L)$ denotes the Volterra model structure as discussed in section 2.1. $\mathbf{D},\mathbf{L}$ represents the two dimensional space in which the candidate models lie, which is defined as:

$$\mathbf{D} = \{1,2\}$$
$$\mathbf{L} = \{10,20,30,40,50,60\}.$$

Hence, the candidate model class set contains of 12 different model classes ($n_M = 12$).

Throughout the experiments, the candidate model classes are constructed with either an uninformative or an informative prior. When designing the uninformative prior, we seek a balance in the freedom the algorithm is given to tune the parameters. Lowering the variance forces the system to describe the input-output relationship with fewer parameters, since one is relatively certain *a priori* that the parameters lie around 0. Equivalently, when the variance of the prior is greater, it gives the algorithm freedom to describe the system with more parameters. Hence, the choice of variance influences the sparsity of the posterior parameter PDF. Having said this, the uninformative prior is defined as

$$p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(\mathbf{0}, 0.1 \cdot \mathbf{I}_{n_H}\right), \tag{3.2}$$

where $\mathbf{I}_{n_H} \in \mathbb{R}^{n_H \times n_H}$ denotes the identity matrix and $\mathbf{0} \in \mathbb{R}^{n_H}$ is a zero vector. The subscripts D, L are omitted here, since the value of n_H is inherently defined by $\mathcal{M}_{D,L}$. The scaling factor 0.1 (α in eq. (2.17)) is found by experimenting and is chosen sufficiently small enough such that sparsity is supported.

The informative prior is constructed according to the method proposed by Birpoutsoukis et al. [6] and described in section 2.2.1. The hyperparameters are found using the non-linear optimization solver Sequential Least Squares Programming (SLSQP) in Python. Although the objective function of the hyperparameter optimization is neither linear nor convex and therefore requires a multi-start optimization technique, during the research it appeared that this makes a minimal difference. Therefore, to reduce computational effort, the hyperparameters were found with a single run.

3.2.2. Excitation Signal

According to Ljung [18], the input sequence $\mathbf{U}_N$ should contain enough information in order to identify particular input-output relations of the system of interest. It is of importance, because as the regression matrix $\mathbf{U}$ less information and thus $\mathbf{U}\mathbf{U}^T$ contains less information, the posterior distribution in eq. (2.29) becomes more dependent on the specific choice of the prior. The amount of information incorporated in the regression matrix is described by the notion of *persistency of excitation* [34]. The singular value decomposition (SVD) allows one to factorize the regression matrix in order to reveal the singular values, which represent a quantification the persistency of excitation.

Figure 3.3 illustrates the singular values of the regression matrix $\mathbf{U}$ for six different Volterra model structures as discussed in section 2.1. The input sequence $\mathbf{U}_N$ is constructed in two different ways. The first (blue line) input sequence is constructed with a multisine signal as proposed by Vlaar et al. [36], who used the odd frequencies in the range of 1 Hz to 23 Hz to generate the signal. This input signal is also been used in the dataset provided for this study. Alternatively, the excitation sequence is constructed as a Gaussian White Noise (GWN), i.e. $u(n) \sim \mathcal{N}(0,1)$ (orange dotted line).

In fig. 3.3, it is shown that the singular values decrease significantly towards zero as the degree of the Volterra series increases for multisine input realizations. Consequently, the eigenvalues of $\mathbf{U}\mathbf{U}^T$ in eq. (2.29) decrease significantly towards zero, which means that the regression matrix $\mathbf{U}$ becomes less decisive with

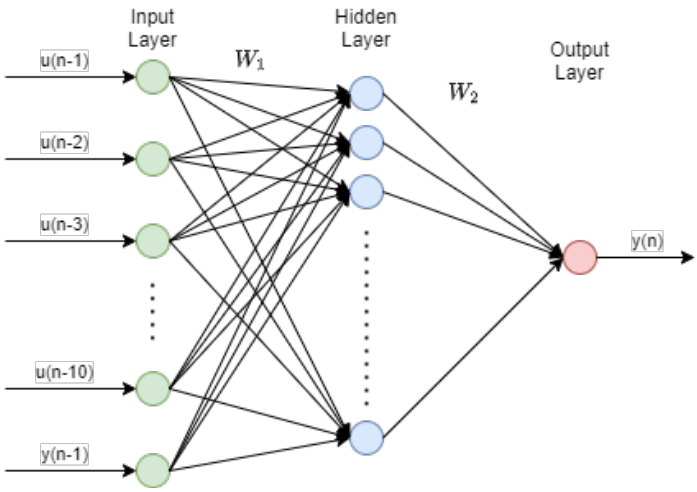


Figure 3.2: The ground truth Neural Network model structure

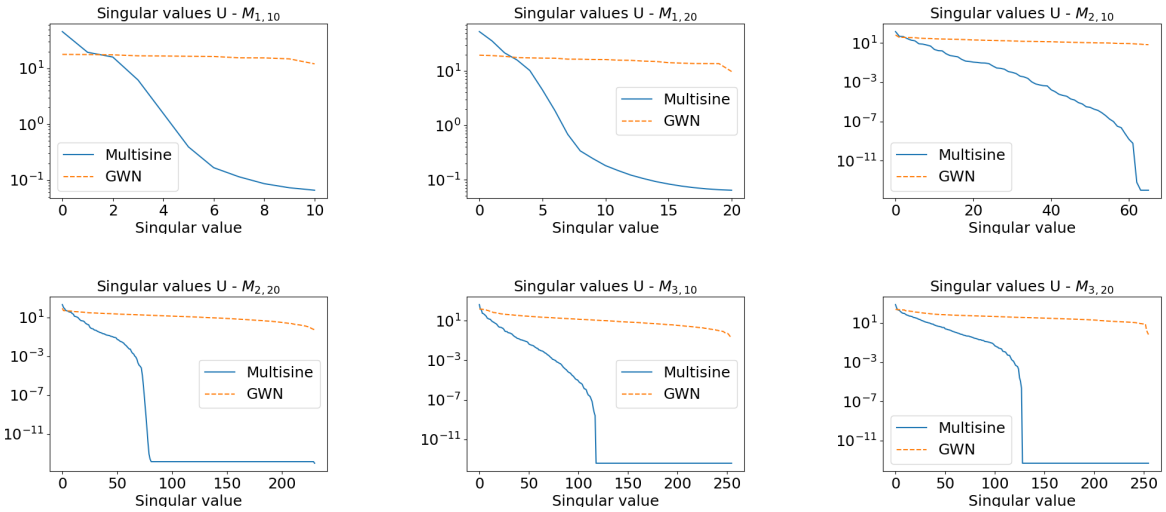


Figure 3.3: The singular values of the regression matrix **U** for six different model structures constructed with both a multisine input and a GWN input.

respect to the posterior distribution of the parameters compared to the prior distribution. In situations like this, it is important that the prior distribution contains definite and substantive information regarding the parameters of interest. Although fig. 3.3 illustrates a similar decrease in singular values for GWN inputs, it turns out that the singular values are generally significantly higher than the singular values of **U** constructed with a multisine input, indicating that the regression matrix contains more information.

A side note should be made. Theoretically, a GWN input signal is ideal, but in practice this is often not possible due to hardware limitations. This combined with the fact that the dataset has already been made available with the multisine input, the latter is used in this study. One should be aware that this may negatively affect the performance of the candidate model class set.

3.3. Results

This section provides the results of the experiments done. First, in section 3.3.1 the algorithm is tested on a Volterra ground truth system. Second, the results of the approach applied on the ground truth Neural Network are provided in section 3.3.2.

3.3.1. Volterra

Figure 3.4 shows the predictions of the system identification phase during the final 250 time steps for the twelve model classes. The posterior parameter PDF is obtained with an uninformative prior and the system acts in a low noise environment, i.e. $\epsilon(n) \sim \mathcal{N}(0,1)\ \forall\ n$. The figure illustrates the high probability regions of the predictions of each model class obtained with the posterior parameter distribution functions. The probability margins are acquired with the variance at time instance n, i.e. σ_n, and σ, 2σ and 3σ denote the 68.2%, 27.2% and 4.4% probability margins respectively. Furthermore, $\mathbf{Y}_N$ is the noise corrupted observation sequence. The corresponding prior model class distribution is illustrated in fig. 3.5.

The following is observed from fig. 3.4 and fig. 3.5:

Observation 1. In fig. 3.4, the first degree candidate model classes $\mathcal{M}_{1,\cdot}$ cannot cope with the ground truth model complexity, which means that the noisy observation sequence $\mathbf{Y}_N$ acts in low probability regions. This corresponds to the first six model classes in the prior model class distribution in fig. 3.5, where it can be seen that all the first degree model classes have zero probability.

Observation 2. At first sight, it seems that the second degree candidate model classes perform equally well. That is, the observations $\mathbf{Y}_N$ are in the high probability regions in fig. 3.4, however fig. 3.5 shows that only $\mathcal{M}_{2,20}$ has a probability in the model class distribution.

The second observation is in line with the study by Muto and Beck [22], who stated that the log-evidence can

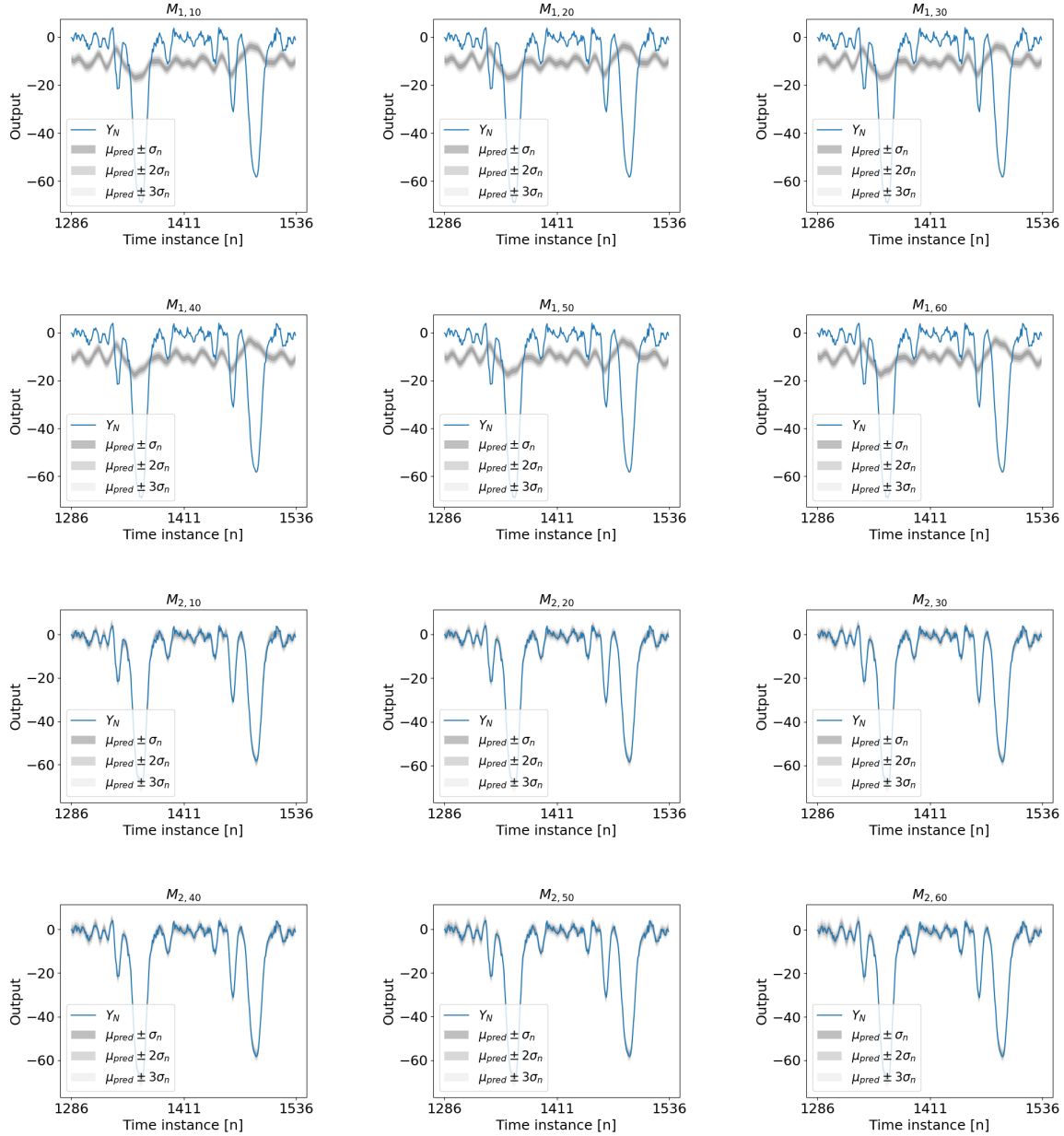


Figure 3.4: The performance of the competitive model class set while modeling the final 250 time steps of the Volterra ground truth system

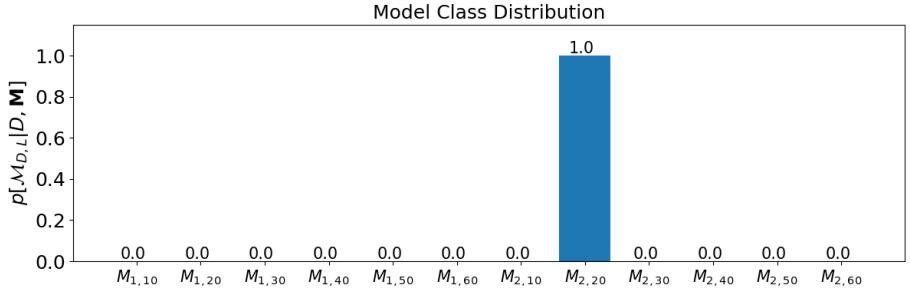


Figure 3.5: The prior MCD of the competitive model obtained with an uninformative prior

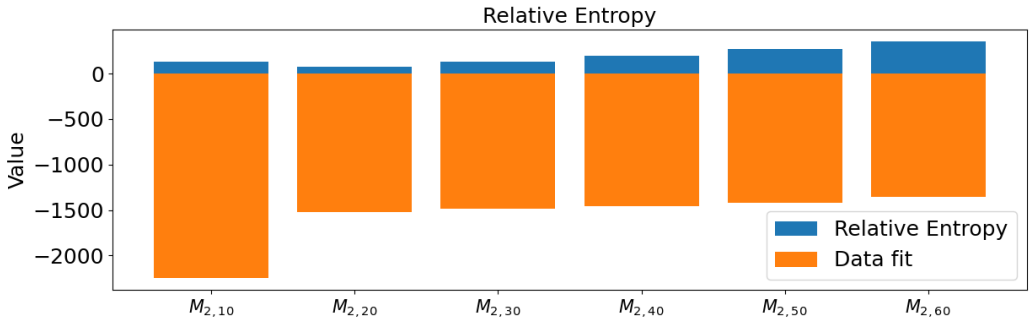


Figure 3.6: Relative Entropy vs. log-datafit of the second degree model classes

Table 3.2: Volterra ground truth: evaluation of the Log-Evidence for the second degree model classes

	$\mathcal{M}_{2,10}$	$\mathcal{M}_{2,20}$	$\mathcal{M}_{2,30}$	$\mathcal{M}_{2,40}$	$\mathcal{M}_{2,50}$	$\mathcal{M}_{2,60}$
$\ln p[\mathbf{Y}_N \vert \mathbf{U}_N \cap \mathcal{M}_{D,L}]$	-2324.98	**-1608.34**	-1624.11	-1626.43	-1701.80	-1748.55

be rewritten as:

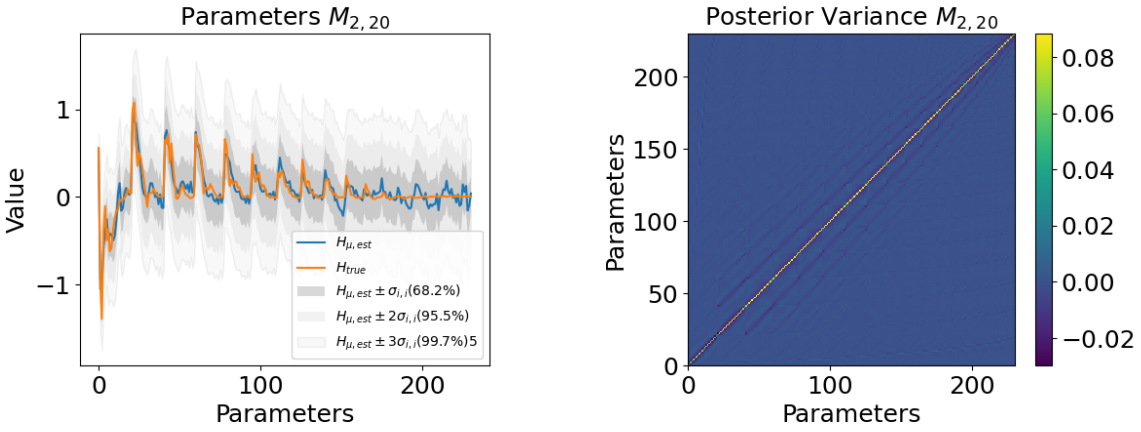
$$\underbrace{\ln\left[p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathcal{M}_{D,L})\right]}_{\text{Log-Evidence}} = \underbrace{\int \ln\left[p(\mathbf{Y}_N|\mathbf{U}_N \cap \mathbf{H} \cap \mathcal{M}_{D,L})\right] p(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L}) d\mathbf{H}}_{\text{Posterior Log-Likelihood fit}} - \underbrace{\int \ln\left[\frac{p(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L})}{p(\mathbf{H}|\mathcal{M}_{D,L})}\right] p(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L}) d\mathbf{H}}_{\text{Relative Entropy}}. \tag{3.3}$$

Here, the first part represents the posterior data fit of the log-likelihood function and the second part represents the information gained from $\mathcal{D}$ to update the parameters $\mathbf{H}$, also known as the *Relative Entropy* or *Kullback-Leibler Divergence* [17, 29]. The log evidence is therefore a combination of a data fit term and a penalty term for models that extract more information from the data. Regarding the Relative Entropy, it represents how different a PDF is relative to a reference distribution. The exact mathematical equation for two arbitrary Gaussian distributions in $\mathbb{R}^n$, $P_1 \sim \mathcal{N}(\mu_1, \Sigma_1)$ and $P_2 \sim \mathcal{N}(\mu_2, \Sigma_2)$, is derived by Duchi [9] and is given by:

$$\begin{aligned} D(P_1||P_2) &= E\left[\log P_1 - \log P_2\right] \\ &= \frac{1}{2}\left(\log \frac{\det \Sigma_2}{\det \Sigma_1} - n + \operatorname{tr}(\Sigma_2^{-1}\Sigma_1) + (\mu_2 - \mu_1)^T \Sigma_2^{-1}(\mu_2 - \mu_1)\right). \end{aligned} \tag{3.4}$$

Figure 3.6 illustrates the comparison between the Relative Entropy and the log-data fit of the second degree candidate model classes. Here, the orange bar graph is found by evaluating the posterior log-likelihood, i.e. eq. (2.47), for the measured values $\mathbf{Y}_N$. While modeling, this represents the fit of each model class. The blue bar graph denotes the Relative Entropy per model class. In fig. 3.6, it can be seen that the Relative Entropy increases as the model structures become more complex (except $\mathcal{M}_{2,10}$), which acts as a penalty in eq. (3.3). Although the posterior datafit increases as the model structures become more complex, it appears that the relative entropy increases faster than the data fit term, consequently meaning that $\mathcal{M}_{2,20}$ is the most probable model structure. This is substantiated in table 3.2. The table depicts the evaluation of the log-evidence as defined in eq. (3.3) set out per model class, where it also appears that $\mathcal{M}_{2,20}$ performs best. This results in the model class distribution as shown in fig. 3.5.

Figure 3.7 shows a simplified representation of the posterior parameter PDF of model class $\mathcal{M}_{2,20}$. In fig. 3.7a, the orange line denotes the true parameters of the ground truth system and the blue line represents the mean of the posterior parameter PDF. Figure 3.7b shows the variance matrix of the posterior parameter PDF, i.e. Σ_{post} in eq. (2.29). The axes denote the parameters involved in the model class, such that the diagonal entries represent the variance of the parameters and the off-diagonal entries depict the covariance between parameters. Figure 3.7b shows that the off-diagonal entries of the variance of the posterior distribution change only minimally due to the acquired observations. For this reason, the uncertainty interval in



(a) Comparison between the true and estimated parameters

(b) Variance of the estimated parameters

Figure 3.7: The posterior parameter PDF compared with the ground truth parameters obtained with an uninformative prior

fig. 3.7a, which illustrates the uncertainty associated with the identification of the parameters, is determined using the diagonal entries of the variance of the multivariate posterior distribution. Although the ground truth parameters do not match the mean of the posterior parameter PDF, it remains within the σ-interval ($\mu \pm \sigma$), which corresponds to a 68.2% probability. Knowing that the posterior distribution is a Gaussian, this region contains the parameters with the highest probability. However, looking at fig. 3.7a, this area is relatively stretched compared to the mean of the distribution, which means that the mean automatically lies in a lower probability region compared to situations when there is less uncertainty present about what the parameters are. This uncertainty affects subsequent predictions in eq. (2.35), hence closing this interval is advantageous for making predictions and the uncertainty involved.

A proper method to do this is by imposing informative prior knowledge. Here, the method discussed in section 2.2.1 introduced by Birpoutsoukis et al. [6] is applied. Figure 3.8 illustrates the resulting prior MCD and fig. 3.9 depicts the comparison between the posterior parameter PDF obtained with an informative prior and the ground truth parameters. The following is observed from this:

Observation 1. Not only is the average fit of the parameters better in fig. 3.9a compared to fig. 3.7a, the uncertainty has also decreased significantly. By suppressing the parameter uncertainty, the forecast uncertainty also remains low in eq. (2.35).

Observation 2. By including an informative prior, the model class distribution is no longer as decisive as it was with the obtained parameters with an uninformative prior in fig. 3.5.

The second observation builds on the earlier discussed arguments following eq. (3.3), more specifically the Relative Entropy. The Relative Entropy is outlined in fig. 3.10 for the twelve different candidate model classes. Here, the blue bar graph denotes the Relative Entropy of the model classes obtained with an uninformative

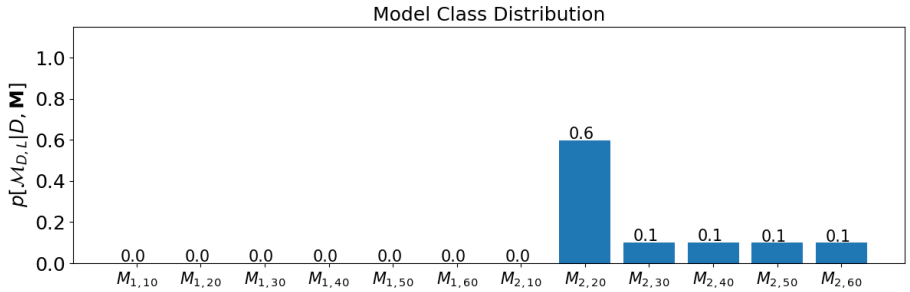
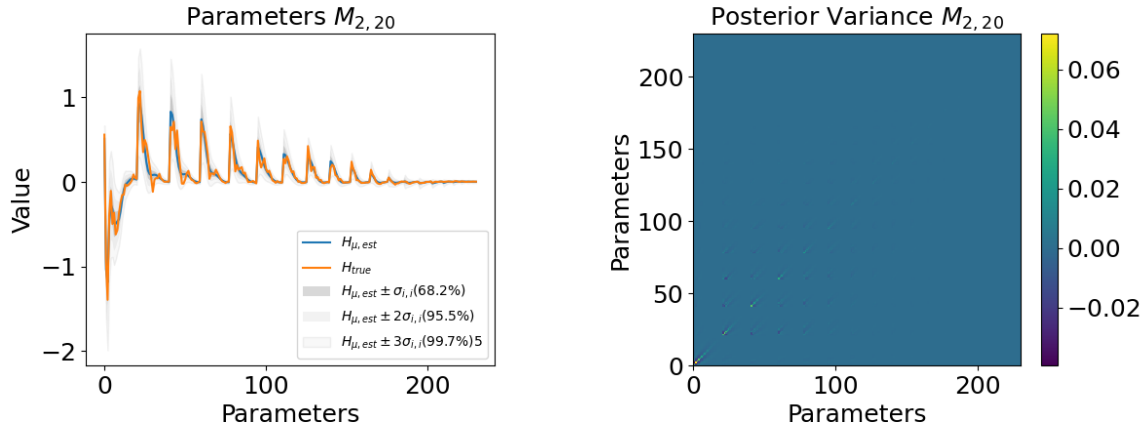


Figure 3.8: Prior MCD obtained with an informative prior



(a) Comparison between the true and estimated parameters

(b) Variance of the estimated parameters

Figure 3.9: The posterior parameter PDF compared with the ground truth parameters obtained with an informative prior

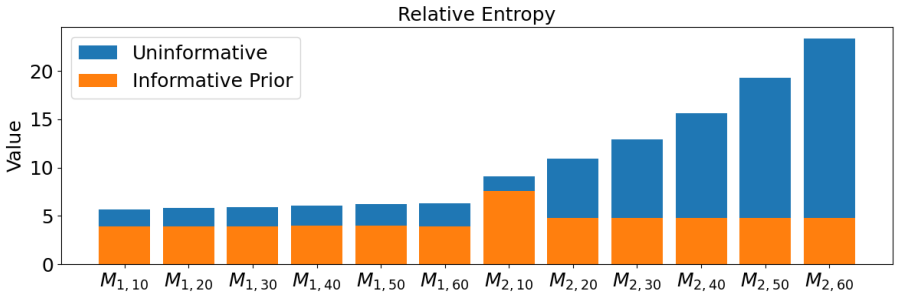


Figure 3.10: Relative Entropy of the twelve candidate model classes compared between the informative and uninformative prior

prior and the orange bar graph represents the Relative Entropy of the model classes obtained with an informative prior. The figure shows that the models $\mathcal{M}_{2,10}$-$\mathcal{M}_{2,60}$ have an increasing Relative Entropy for model classes constructed with an uninformative prior, meaning that the relative entropy compensates for the amount of information gained from the data in eq. (3.3). In contrast, the Relative Entropy remains at a comparable level for the candidate model classes found with an informative prior, which means that the information obtained between the prior and posterior from the data is relatively low. This can be explained because during the hyperparameter optimization procedure as much information as possible was already extracted from the data and embedded in the prior knowledge, by maximizing the marginal likelihood. So the prior contains a certain amount of information so that Bayesian Inference updates the parameters only minimally. Thus, in the current situation, imposing an informative prior influences the prior MCD negatively.

So far, an analysis has been done how different priors influence the uncertainty of the estimated parameters and how this affects the prior MCD. The next section examines how this affects making forecasts.

Predictive Analysis

Figure 3.11 illustrates the posterior MCD for the competitive model class set obtained with an uninformative prior evaluated for the validation dataset. The graph shows two different PDFs. First, the blue bar graph displays the relative probability of the competitive model class set excluding the HRP. This bar graph is used to understand the performance of the separate model classes with respect to each other. The orange bar graph displays the MCD including the HRP. This graph is used to understand the performance of the HRP compared to the competitive model class set. In addition, table 3.3 depicts the performance of the mean of the competitive model class set and the HRP. The RMSE and VAF are found by comparing the mean of the respective distribution with the true value of the validation set $\mathbf{Y}_{\text{true,val}}$.

From fig. 3.11 it is observed that the posterior MCD is in line with the prior MCD in fig. 3.5, indicating that, given the circumstances, Bayesian Inference using Volterra Series is a reliable method to reconstruct

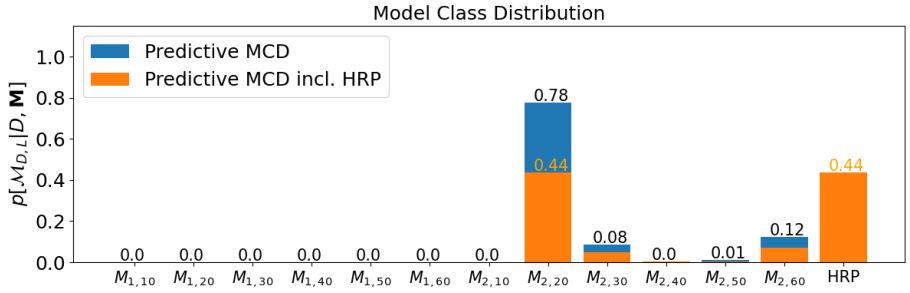


Figure 3.11: Predictive posterior MCD obtained with an uninformative prior

Table 3.3: Volterra ground truth: performance of candidate models obtained with an uninformative prior

	$\mathcal{M}_{1,10}$	$\mathcal{M}_{1,20}$	$\mathcal{M}_{1,30}$	$\mathcal{M}_{1,40}$	$\mathcal{M}_{1,50}$	$\mathcal{M}_{1,60}$	$\mathcal{M}_{2,10}$	$\mathcal{M}_{2,20}$	$\mathcal{M}_{2,30}$	$\mathcal{M}_{2,40}$	$\mathcal{M}_{2,50}$	$\mathcal{M}_{2,60}$	**HRP**
RMSE	6.08	6.08	6.09	6.1	6.07	6.08	0.59	0.11	0.18	0.2	0.25	0.26	0.11
VAF [%]	36.02	36.04	35.9	35.68	36.21	36.13	99.4	99.98	99.95	99.93	99.89	99.88	99.98

the ground truth model structure, which subsequently performs best during the validation phase. This is supported by table 3.3, where it is found that $\mathcal{M}_{2,20}$ performs best in terms of RMSE and VAF, namely 0.11 and 99.98% respectively.

Equivalently, fig. 3.12 illustrates posterior MCD of the competitive model class set obtained with an informative prior. Here it can be seen that the second degree model classes with lag 20 and onward perform (almost) equally well in the posterior MCD. This is in contrast with the model classes obtained with an uninformative prior, where $\mathcal{M}_{2,20}$ is strongly favored by the data. Hence, while using an uninformative prior PDF, the model classes $\mathcal{M}_{2,30} - \mathcal{M}_{2,60}$ are noticeable overfitted during modeling. Imposing an informative prior prevents the too complex model classes from being overfitted during modeling.

This is also supported by table 3.4. The table shows the fit of the mean of the respective distribution by evaluating the RMSE or the VAF for the true underlying signal $\mathbf{Y}_{\text{true,val}}$. It can be seen that both the RMSE and the VAF remain at a comparable level while increasing the model complexity more than necessary. In contrast, the performance of the RMSE and VAF decrease as the model complexity increases for the model classes obtained with an uninformative prior in table 3.3. This supports the argument that these model classes are overfitted.

The mentioned results are a direct consequence of the way the prior distribution is determined. The minimization of the objective function in eq. (2.21) is constrained by a zero-mean prior distribution, which means that the optimization algorithm pre-tunes the parameters towards zero, i.e. low variance, based on the input-output relationship available. This is substantiated by fig. 3.13. The figure shows the mean of the posterior parameter PDFs for the second degree model classes including the uncertainty levels. For each subfigure, the x-axis denotes the parameters involved in the Volterra kernel and the y-axis denotes its corresponding value. Here, it can be seen that the mean of the posterior $p(\mathbf{H}|\mathcal{D} \cap \mathcal{M}_{D,L})$ is steered towards zero as the lag becomes larger. The non-zero parameters that remain are those that correspond to the model structure of $\mathcal{M}_{2,20}$.

The HRP PDFs obtained with an uninformative and informative prior are illustrated in figs. 3.14a and 3.14b

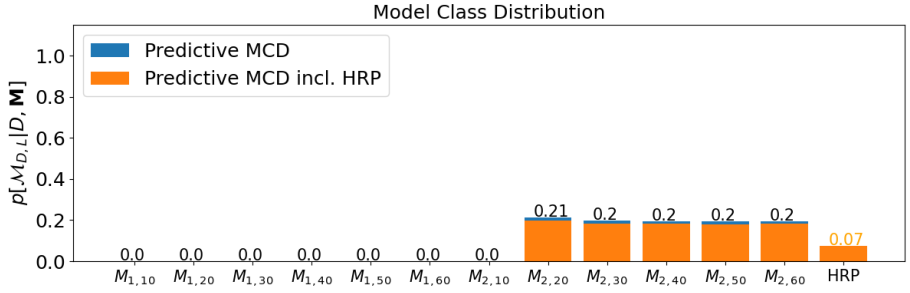


Figure 3.12: Predictive posterior MCD obtained with an informative prior

Table 3.4: Volterra ground truth: performance of candidate models with an informative prior

	$\mathcal{M}_{1,10}$	$\mathcal{M}_{1,20}$	$\mathcal{M}_{1,30}$	$\mathcal{M}_{1,40}$	$\mathcal{M}_{1,50}$	$\mathcal{M}_{1,60}$	$\mathcal{M}_{2,10}$	$\mathcal{M}_{2,20}$	$\mathcal{M}_{2,30}$	$\mathcal{M}_{2,40}$	$\mathcal{M}_{2,50}$	$\mathcal{M}_{2,60}$	**HRP**
RMSE	6.08	6.08	6.09	6.1	6.08	6.07	0.54	0.09	0.1	0.1	0.1	0.1	0.09
VAF [%]	36.04	36.04	35.97	35.68	36.05	36.35	99.49	99.99	99.98	99.98	99.98	99.98	99.99

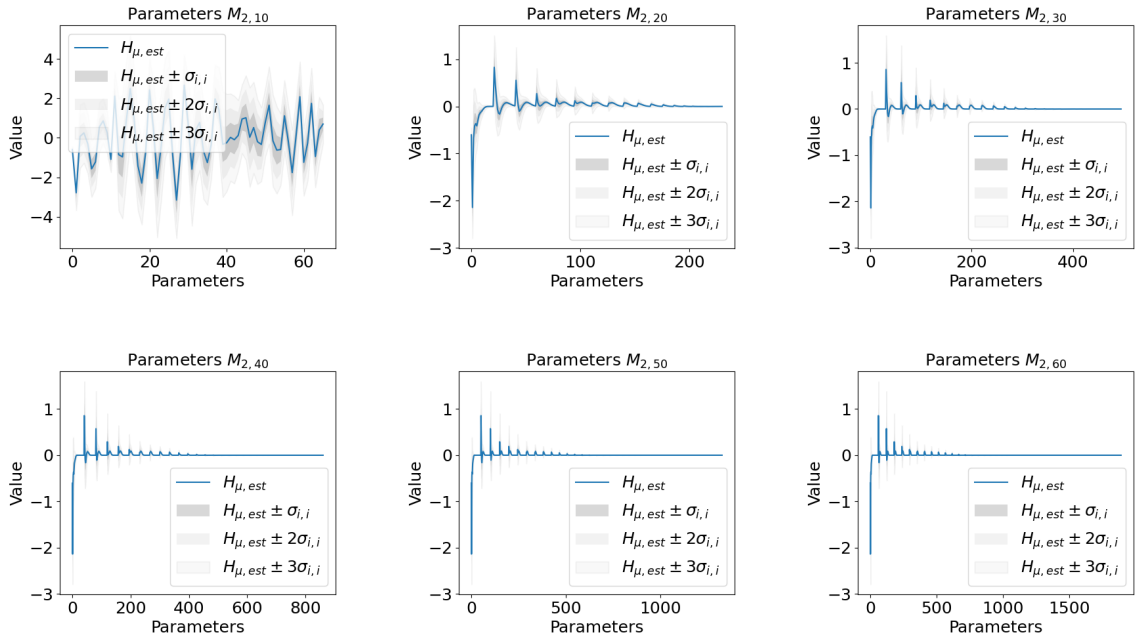
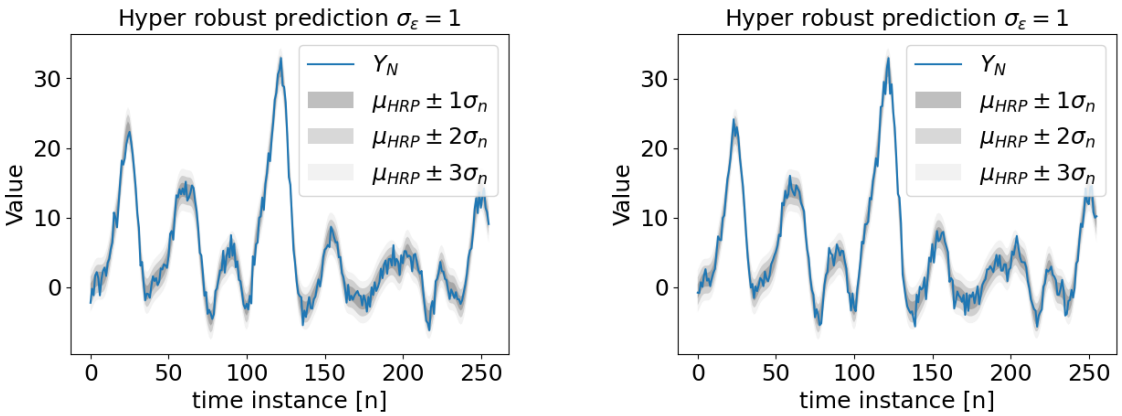


Figure 3.13: Posterior parameters of the second degree model classes

and the corresponding MCDs are illustrated in figs. 3.11 and 3.12 respectively. As expected from fig. 3.5, the HRP obtained with an uninformative prior performs equally well in a Bayesian sense as $\mathcal{M}_{2,20}$. This is also reflected in terms of RMSE and VAF in table 3.3, where both outperform the other candidate models, having a RMSE and VAF of 0.11 and 99.98% respectively. The HRP PDF obtained with an informative prior, however, has a lower probability in the MCD in fig. 3.12 compared with the averaged model classes $\mathcal{M}_{2,20} - \mathcal{M}_{2,60}$, indicating that, given the circumstances, averaging over the model class distribution leads to false confidence. This is supported in table 3.4, since the model classes with which $\mathcal{M}_{2,20}$ is averaged perform slightly worse in terms of RMSE and VAF.



(a) Uninformative prior

(b) Informative prior

Figure 3.14: Hyper Robust Predictions

This experiment has shown that averaging over the model class set does not necessarily lead to improved results. In the situation where the uninformative prior was used, it turned out that Bayesian Inference was well capable of reconstructing the ground truth model structure ($p\left[\mathcal{M}_{2,20}|\mathcal{D}\cap\mathbf{M}\right]=1$), therefore only averaging over a single model class. During validation this model class, as well as the HRP, performed best in terms of RMSE, VAF and the predictive MCD. In contrast, using an informative prior HRP has had a negative influence on the performance of the forecast, where it turned out that it is less likely that the noisy observations $\mathbf{Y}_{\text{N,val}}$ are produced by the HRP with respect to $\mathcal{M}_{2,20}-\mathcal{M}_{2,60}$. In retrospect, it would have been wise to continue with the model that performed best in the model class distribution ($\mathcal{M}_{2,20}$ in fig. 3.8), as it also performed best in the validation phase.

There are, however, some important comments to report regarding the ground truth model structure with respect to EEG signals. It is well known that EEG signals are usually highly corrupted with noise. Consequently, increasing noise corruption may disrupt the model class distribution in figs. 3.5 and 3.8, such that the best performing model class does not necessarily outperform the other model classes during the validation phase. Furthermore, the experiments are done knowing that the ground truth model structure is part of the model class set. However, in reality, it is not plausible that the relation between wrist joint manipulations and the cortical responses behaves like a Volterra Series. Having said this, in the next experiment the ground truth system is modeled as a Neural Network and throughout the experiment the noise is increased.

3.3.2. Noisy Neural Network

The previous experiment assumed that the system of interest was in a low noise environment. However, it is well known that EEG signals have a poor SNR, ranging between -10db and -40db [35, 36]. In addition, in section 3.3.1 it was assumed that the ground truth system has a Volterra structure, however, it is not likely that the underlying model has the same structure as the candidate models. In order to mimic the human nervous system, the ground truth system is adjusted such that it exhibits non-Volterra and non-linear dynamics and is highly corrupted with noise. First, the system is considered under mild conditions. This is to examine the performance of the Bayesian Inference algorithm knowing that the ground truth system structure does not match any of the models in the model class set. Second, the system is considered under tough conditions, meaning that the noise corruption is increased such that the SNR ranges between 0dB and -30dB.

Mild conditions

Figure 3.15 illustrates the prior MCD obtained with an uninformative prior for the NN ground truth system in low-noise conditions, i.e. $\epsilon(n)\sim\mathcal{N}(0,1)\ \forall\ n$. Furthermore, table 3.5 depicts the corresponding performance of the mean of the respective model class expressed in terms of the VAF and the RMSE and fig. 3.16 illustrates the posterior MCD, both evaluated during the validation phase.

In contrast to the system identification of the Volterra ground truth system, using the current modeling approach, Bayesian Inference is not able to find the model class in the prior MCD that performs best during the validation phase. In table 3.5, $\mathcal{M}_{1,20}-\mathcal{M}_{1,60}$ and $\mathcal{M}_{2,20}$ perform equally well, having an approximate RMSE and VAF of 0.77 and 99.33% respectively. The prior MCD, however, indicates full confidence for $\mathcal{M}_{2,60}$, which performs worse in terms of RMSE and VAF, respectively 1.53 and 97.67%. This misconception is supported in fig. 3.16, which shows that the first degree model classes $\mathcal{M}_{1,20}-\mathcal{M}_{1,60}$ have a higher probability in the posterior MCD.

Furthermore, comparing table 3.5 with fig. 3.16 reveals an additional advantage using Bayesian Infer-

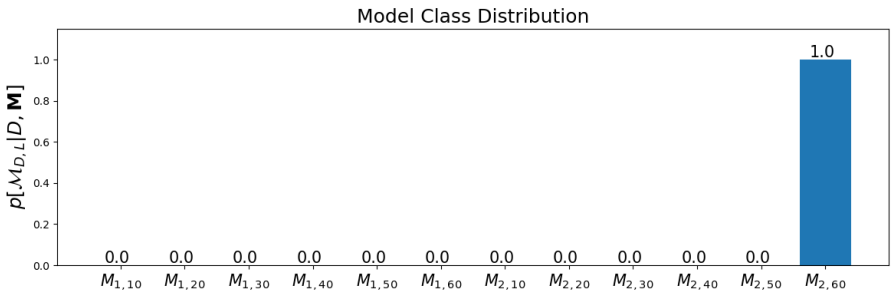


Figure 3.15: Prior MCD obtained with an uninformative prior

Table 3.5: NN ground truth: performance of candidate models with an uninformative prior

	$\mathcal{M}_{1,10}$	$\mathcal{M}_{1,20}$	$\mathcal{M}_{1,30}$	$\mathcal{M}_{1,40}$	$\mathcal{M}_{1,50}$	$\mathcal{M}_{1,60}$	$\mathcal{M}_{2,10}$	$\mathcal{M}_{2,20}$	$\mathcal{M}_{2,30}$	$\mathcal{M}_{2,40}$	$\mathcal{M}_{2,50}$	$\mathcal{M}_{2,60}$	**HRP**
RMSE	0.85	0.77	0.77	0.77	0.77	0.77	0.84	0.77	0.8	0.95	1.23	1.53	1.53
VAF [%]	99.2	99.33	99.33	99.33	99.33	99.33	99.22	99.35	99.32	99.09	98.49	97.67	97.67

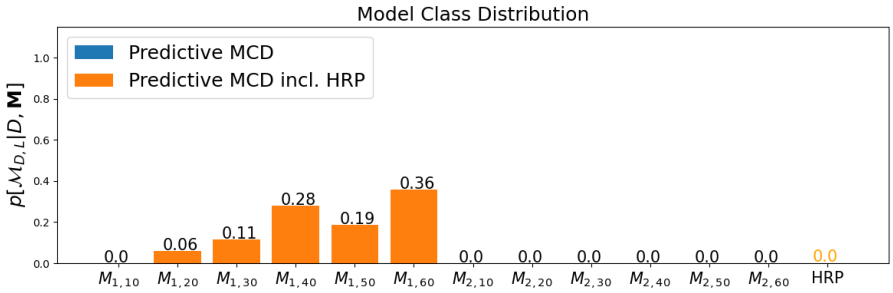


Figure 3.16: Predictive posterior MCD obtained with an uninformative prior

ence for model evaluation. Based on the RMSE and VAF with respect to $\mathbf{Y}_{\text{true,val}}$, table 3.5 indicates that the model classes $\mathcal{M}_{1,20} - \mathcal{M}_{1,60}$ and $\mathcal{M}_{2,20}$ perform equally well, however, these calculations do not include either parameter or noise uncertainty. Besides, the true noiseless output measurements $\mathbf{Y}_{\text{true,val}}$ are often not available. The predictive MCD in fig. 3.16 not only calculates the model probability based on the noisy observations $\mathbf{Y}_{\text{N,val}}$, it also includes uncertainty regions that are used to determine the relative model fit. Consequently, it is more certain that the model class $\mathcal{M}_{1,60}$ followed by $\mathcal{M}_{1,40}$ corresponds with the ground truth model compared to the model class set.

In section 3.3.1 it is discussed that the specific choice for the prior distribution influences the model distribution as well as the performance of the algorithm. While modeling the Volterra ground truth system, the chosen prior distribution, i.e. $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}(0, 0.1\mathbf{I}_N)$, appeared to contain sufficient information to find the model class in the prior MCD which subsequently performed best during validation. In the current situation, the algorithm is not provided with a sufficient amount of information during modeling to detect the right model class which performs best during validation. In figs. 3.15 to 3.16, the model class $\mathcal{M}_{2,60}$ is clearly overfitted during modeling and it requires a penalty in order to move the probability in the prior MCD towards the first degree model classes. Recall that the Relative Entropy in the evidence, as described in section 3.3.1, acts as a penalty in order to compensate the Log-Likelihood fit for complex models for the amount of information transferred from the prior parameter PDF to the posterior parameter PDF. Since the more complex model classes contain more parameters and can therefore transfer more information from the prior to the posterior parameter PDF, it is expected that by raising the prior variance the Relative Entropy increases faster for more complex model classes, since it provides the algorithm more freedom in tuning the parameters.

In the following experiment the prior variance is altered slightly in order to resolve the issues mentioned above. Here, the prior PDF is defined as:

$$p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.4 \cdot \mathbf{I}_{n_H}\right). \tag{3.5}$$

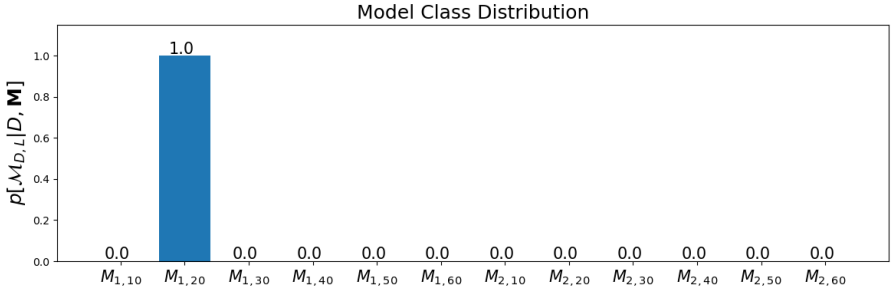


Figure 3.17: Prior MCD obtained with the altered prior $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.4 \cdot \mathbf{I}_{n_H}\right)$

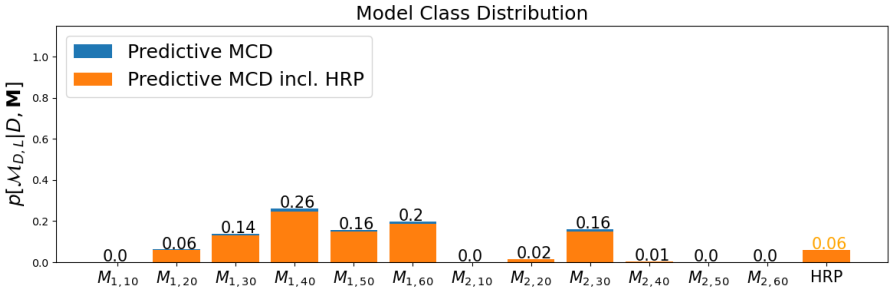


Figure 3.18: Predictive posterior MCD obtained with the altered prior $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.4 \cdot \mathbf{I}_{n_H}\right)$

Table 3.6: NN ground truth: performance of candidate models obtained with the altered prior $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.4 \cdot \mathbf{I}_{n_H}\right)$

	$\mathcal{M}_{1,10}$	$\mathcal{M}_{1,20}$	$\mathcal{M}_{1,30}$	$\mathcal{M}_{1,40}$	$\mathcal{M}_{1,50}$	$\mathcal{M}_{1,60}$	$\mathcal{M}_{2,10}$	$\mathcal{M}_{2,20}$	$\mathcal{M}_{2,30}$	$\mathcal{M}_{2,40}$	$\mathcal{M}_{2,50}$	$\mathcal{M}_{2,60}$	**HRP**
RMSE	0.81	0.76	0.77	0.77	0.77	0.77	0.81	0.76	0.76	0.78	0.84	0.92	0.76
VAF [%]	99.24	99.33	99.33	99.33	99.33	99.33	99.25	99.34	99.34	99.31	99.22	99.08	99.33

Figure 3.17 illustrates the prior MCD with the altered prior variance and table 3.6 and fig. 3.18 show the corresponding performance during the validation phase. It can be seen that the favoured prior model class $\mathcal{M}_{1,20}$ performs equally well as the model classes $\mathcal{M}_{2,20}$ and $\mathcal{M}_{2,30}$ in terms of RMSE and VAF, respectively 0.76 and 99.33%. However, the model classes $\mathcal{M}_{1,40}$ and $\mathcal{M}_{1,60}$ remain the best performing model classes in terms of the posterior MCD in fig. 3.18. This indicates that these model classes describe the output signal with more certainty.

Here, the assumption was made that by raising the prior variance the Relative Entropy increases faster for more complex model classes. Figure 3.19 illustrates the comparison of the Relatieve Entropy between the two different priors. The blue and the orange graph denote the Relative Entropy for $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.1 \cdot \mathbf{I}_{n_H}\right)$ and $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.4 \cdot \mathbf{I}_{n_H}\right)$ respectively. Here, σ_p denotes the scaling factor for the variance. First of all, it can be seen that the Relative Entropy for $\sigma_p = 0.1$ is in general higher than the Relative Entropy for $\sigma_p = 0.4$. This indicates that in the initial situation the algorithm updates the prior PDF more rigorously compared to the second situation. Second, as the model classes become more complex, the Relative Entropy decreases for $\sigma_p = 0.1$, while it increases for $\sigma_p = 0.4$. Intuitively, the latter makes more sense, given that it imposes a penalty on more complex models. The graph shows that there are inaccuracies present in the former situation, which comes to light by evaluating the prior and posterior MCD.

However, in none of the experiments done led Bayesian Model Averaging to improved results. In both the Neural Network experiments the prior MCD was decisive, providing full probability to a single model class. It was demonstrated in section 3.3.1 that imposing an informative prior influences the prior MCD in a sense

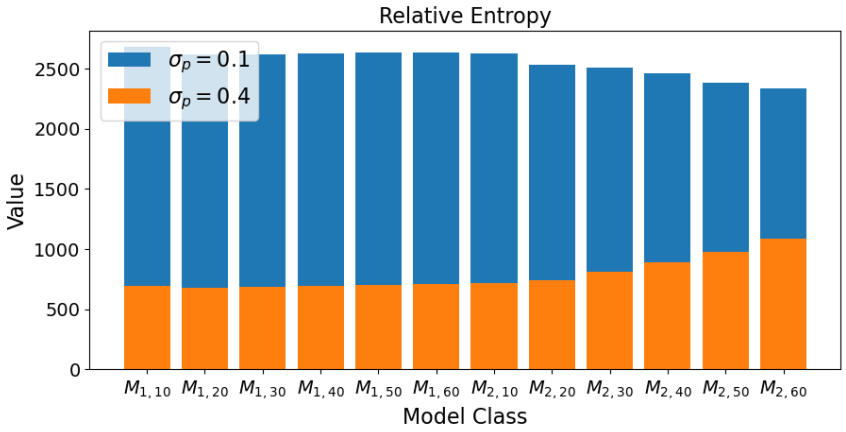


Figure 3.19: The Relative Entropy comparison for two different prior variances

that it is less decisive. In these situations the question, however, is which preconditions must be established such that model averaging yields in more certain predictions. This is explained in the following illustrative example.

Example 3. *Consider the two Gaussian distribution functions predicting an arbitrary value* $\mathbf{y}$ *as shown in fig. 3.20. Here, the blue line represents an arbitrary model class which performs best in the MCD (e.g.* $\mathcal{M}_{2,20}$ *in fig. 3.8) and the orange line represents the HRP, which is averaged over at least two model classes.*

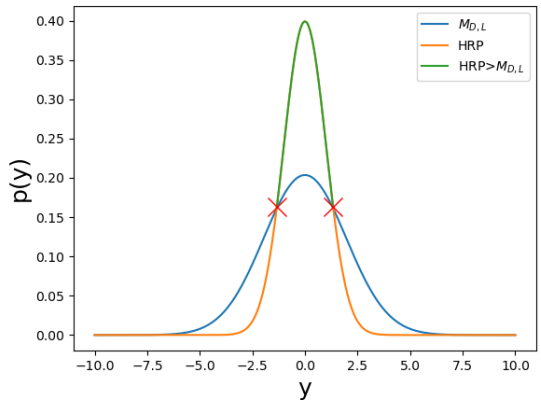


Figure 3.20: Two normal distributions

It is known that both PDFs have an equal mean μ *(0) and a different variance, such that* $\sigma_{\mathcal{M}_{D,L}} > \sigma_{HRP}$*. To make sure that the HRP is in favour with respect to* $\mathcal{M}_{D,L}$*, the measured value of* $\mathbf{y}$ *should lie in between the intersections of both distributions, since then* $p(y)_{HRP} > p(\mathbf{y})_{\mathcal{M}_{D,L}}$ *(green line). Having said this, the distance between the mean of the prediction* μ *and the measured value is bounded by half the distance between the intersections.*

For two zero-mean normal distributions, it can be shown that this intersection equals:

$$\mathbf{y}_{intersect} = \frac{\sigma_{HRP}\,\sigma_{\mathcal{M}_{D,L}}\sqrt{2\ln\left(\frac{\sigma_{\mathcal{M}_{D,L}}}{\sigma_{HRP}}\right)}}{\sqrt{\sigma^2_{\mathcal{M}_{D,L}} - \sigma^2_{HRP}}}. \tag{3.6}$$

Here, σ_{HRP} *and* $\sigma_{\mathcal{M}_{D,L}}$ *denote the standard deviation of HRP and* $\mathcal{M}_{D,L}$ *respectively. During the following experiment it is assumed that the parameter uncertainty does not change with different values for noise corruption, therefore, according to eq. (2.42), the difference between the value of the variance of the HRP and* $\mathcal{M}_{D,L}$

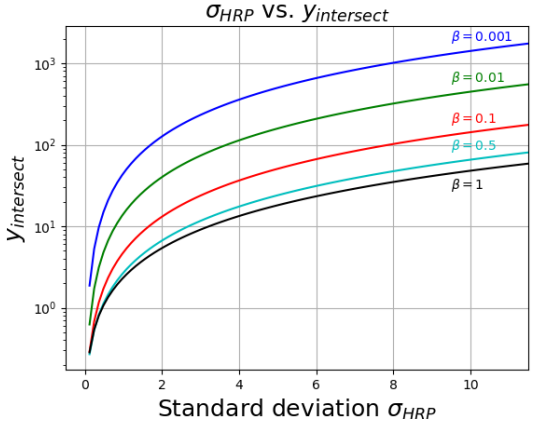


Figure 3.21: Relation between σ_{HRP} and $\mathbf{y}_{\text{intersect}}$

remains the same as the noise variance increases. Having said this, the variance of $\mathcal{M}_{D,L}$ *can be written as:* $\sigma_{\mathcal{M}_{D,L}} = \sigma_{HRP} + \beta$*, where* β *denotes the constant difference in variance. Equation* (3.6) *then becomes:*

$$\mathbf{y}_{intersect} = \frac{\left(\sigma_{HRP}^2 + \beta\sigma_{HRP}\right)\sqrt{2\ln\left(1+\frac{\beta}{\sigma_{HRP}}\right)}}{\beta}. \tag{3.7}$$

Since it is assumed that the uncertainty parameter does not change as the noise increases, σ_{HRP} *increases linearly with the noise variance. The corresponding relation between* σ_{HRP} *and* $\mathbf{y}_{intersect}$ *is illustrated in fig. 3.21.*

The figure shows the exponential increase of $\mathbf{y}_{intersect}$ *with respect to* σ_{HRP} *for various values of* β*. It can be seen that* $\mathbf{y}_{intersect}$ *increases as* β *decreases. Furthermore,* $\mathbf{y}_{intersect}$ *grows exponentially as* σ_{HRP} *increases linearly. Therefore, it is expected that model averaging yields more confident results in environments with increased uncertainty.*

Noisy conditions

In the current situation, the NN ground truth system is exposed to noisy conditions, i.e. $\epsilon \sim \mathcal{N}(0,50)\ \forall\ n$. First, the model classes are obtained with the prior $p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 0.1\cdot\mathbf{I}_{n_H}\right)$. Figure 3.22 illustrates the prior MCD and fig. 3.23 and table 3.7 display the performance of each model class based on the validation data. Similarly as described for the Neural Network exposed under mild conditions, the model class $\mathcal{M}_{2,60}$ has incorrectly been assigned the highest probability in fig. 3.22, since it does not perform best during validation with respect to the model class set. In fig. 3.23, it is most certain that the observed noise corrupted output data is generated by $\mathcal{M}_{2,50}$. It is difficult to assign a best performing model class based on table 3.7, since none of the model classes outperforms the model class set both in terms of RMSE and VAF. In addition, the overall performance of the mean of the model classes has decreased significantly compared to the experiments under mild conditions in table 3.5.

This deterioration in performance can be traced back to the calculation of the posterior parameter PDF in eq. (2.29), specifically the balance between the prior variance Σ_p and the influence of the acquired data $\mathbf{U}\Sigma_\epsilon^{-1}\mathbf{U}^T$. It is discussed in section 3.2.2 that the regression matrix $\mathbf{U}$ is close to singular for higher degree Volterra model classes. With increasing noise corruption, Σ_ϵ increases and therefore the amount of information embedded in $\mathbf{U}\Sigma_\epsilon^{-1}\mathbf{U}^T$ decreases even further. Consequently, the dependency of the specific choice of the prior rises. During this experiment an attempt was made to improve the results by adjusting the factor that scales the identity matrix of the prior variance, but this was unsuccessful. This means that the identity matrix is not sufficient and an alternative method must be used to design the prior. Therefore, in the following experiment the method proposed by Birpoutsoukis et al. [6] as described in section 2.2.1 is applied to design the informative prior.

Figure 3.24 illustrates the prior MCD obtained with an informative prior and table 3.8 and fig. 3.25 depict the corresponding performance of the model class set based on the RMSE, VAF and posterior MCD respectively during the validation phase. It appears from the prior MCD in fig. 3.24 that $\mathcal{M}_{2,50}$ is most likely to match

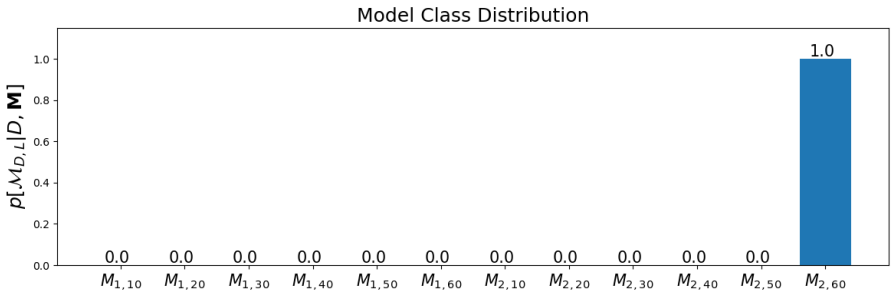


Figure 3.22: Prior MCD in noisy conditions obtained with an uninformative prior

Table 3.7: NN ground truth: performance of candidate models in noisy conditions obtained with an uninformative prior

	$\mathcal{M}_{1,10}$	$\mathcal{M}_{1,20}$	$\mathcal{M}_{1,30}$	$\mathcal{M}_{1,40}$	$\mathcal{M}_{1,50}$	$\mathcal{M}_{1,60}$	$\mathcal{M}_{2,10}$	$\mathcal{M}_{2,20}$	$\mathcal{M}_{2,30}$	$\mathcal{M}_{2,40}$	$\mathcal{M}_{2,50}$	$\mathcal{M}_{2,60}$	**HRP**
RMSE	22.63	22.34	22.36	22.39	22.3	22.31	16.9	14.1	13.0	13.65	14.25	14.49	14.49
VAF [%]	52.07	62.05	61.95	61.53	61.91	61.88	-4.35	22.21	22.6	-16.18	-42.78	-67.74	-67.74

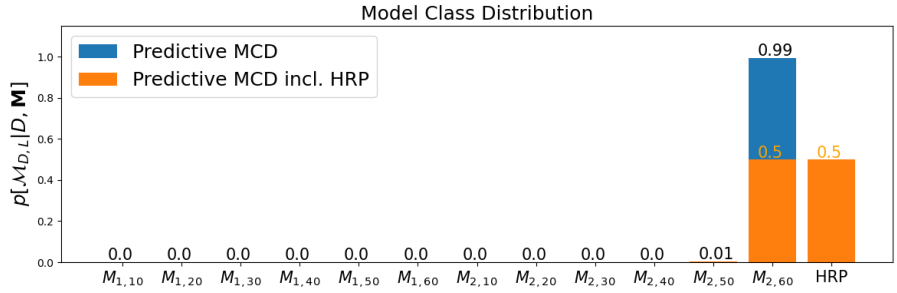


Figure 3.23: Predictive posterior MCD in noisy conditions obtained with an uninformative prior

the system identification data, followed by $\mathcal{M}_{2,60}$, $\mathcal{M}_{2,40}$ and $\mathcal{M}_{2,10}$. Using conventional methods, $\mathcal{M}_{2,50}$ would have been picked for validation yielding in a RMSE and a VAF of 2.84 and 94.37% respectively. However, looking at table 3.8, excluding $\mathcal{M}_{2,10}$, $\mathcal{M}_{2,50}$ is the worst performing model class in terms of RMSE and VAF. This is also reflected in fig. 3.16, where it appears that $\mathcal{M}_{2,50}$ is the least likely to match the observed output data. In such cases, it makes sense to be able to average the worst-performing model class with better model classes, so that the HRP ultimately performs better.

This is supported by the RMSE and VAF of the HRP in table 3.8 and the posterior MCD in fig. 3.25. The HRP performs better in terms of RMSE and VAF, namely 2.67 and 95.12% to 2.84 and 94.37% respectively. The posterior MCD in fig. 3.25 also favors the HRP over $\mathcal{M}_{2,50}$, however, the algorithm still has trouble finding the best model class during system identification which performs best during validation, since the HRP is outperformed by the model classes $\mathcal{M}_{1,10}-\mathcal{M}_{1,60}$, both in terms of RMSE and VAF in table 3.8 and the posterior MCD in fig. 3.25.

Furthermore, based on the RMSE and VAF, it can be seen that imposing an informative prior yield a significant improvement in performance in table 3.8 compared to table 3.7. These results confirm that the prior has more influence on the performance of the competitive model class set when the information embedded in $\mathbf{U}\Sigma_e^{-1}\mathbf{U}^T$ decreases, either caused by increased noise corruption or an excitation signal which is not persistent for the degree for the model class to be determined.

3.4. Conclusion

In this chapter, the Bayesian Inference algorithm is applied on two different computer models. First, given that the ground truth system is a Volterra system and the model class set is obtained with an uninformative prior, the algorithm was able to reconstruct the ground truth model. The respective model class appeared to be most likely in the posterior MCD obtained with the validation dataset and also performed best in terms

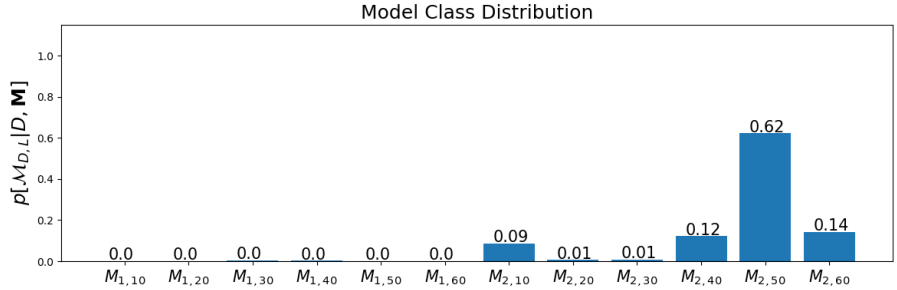


Figure 3.24: Prior MCD in noisy conditions obtained with an informative prior

Table 3.8: NN ground truth: performance of the candidate models in noisy conditions obtained with an informative prior

	$\mathcal{M}_{1,10}$	$\mathcal{M}_{1,20}$	$\mathcal{M}_{1,30}$	$\mathcal{M}_{1,40}$	$\mathcal{M}_{1,50}$	$\mathcal{M}_{1,60}$	$\mathcal{M}_{2,10}$	$\mathcal{M}_{2,20}$	$\mathcal{M}_{2,30}$	$\mathcal{M}_{2,40}$	$\mathcal{M}_{2,50}$	$\mathcal{M}_{2,60}$	**HRP**
RMSE	2.28	2.13	2.13	2.14	2.14	2.14	2.93	2.41	2.39	2.58	2.84	2.59	2.67
VAF [%]	96.64	97.45	97.49	97.45	97.41	97.4	93.2	96.03	96.19	95.33	94.37	95.32	95.12

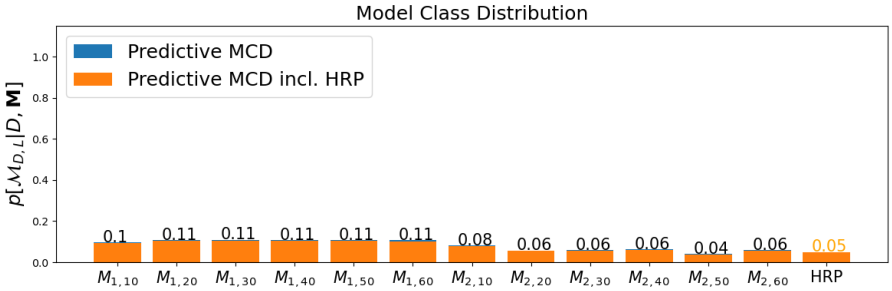


Figure 3.25: Predictive posterior MCD in noisy conditions obtained with an informative prior

of RMSE and VAF. By imposing an informative prior, the respective model class performed slightly better, however this action distorted the prior MCD, making it less decisive in the most probable model classes.

Second, the algorithm was applied on a Neural Network ground truth system while increasing the noise corruption. It appeared that under mild noise conditions, the prior variance had an significant influence on which model class experienced the highest probability in the prior MCD. However, by adjusting the variance it was possible to improve the performance of the model classes. With increased noise corruption, adjusting the prior variance did not yield satisfactory results. By imposing the informative prior as discussed in section 2.2.1, the performance of the model classes improved significantly.

One of the main objectives of this study was to investigate whether Bayesian Model Averaging yield satisfactory results. It is shown that under mild conditions while imposing an uninformative prior, model averaging did not matter, since in all the experiments a single model class was given full preference. However, when an informative prior was imposed, the prior MCD became less decisive. Unfortunately, in this situation averaging over the candidate model class did not outperform the single model classes. This is in contrast to the noisy conditions, where averaging across the model class has been shown to led to better performance.

4

Cortical Responses

This chapter is devoted to the implementation of the discussed theory on the cortical responses evoked by wrist joint manipulations. First of all, in section 4.1 the experimental setup is elaborated briefly as performed by Vlaar et al. [35, 36]. Second, the modeling approach is explained in section 4.2. Finally, the results are shown in section 4.3.

4.1. Experimental Setup

The data is recorded from ten different participants (six men, four women) and the experiment is approved by the Human Research Ethics Committee of the Delft University of Technology. The participants were instructed to gaze at a static screen, while having their hand strapped to a robotic manipulator used to evoke the wrist joint (see fig. 1.4). Furthermore, EEG is used to measure the cortical activity using a 128-channel cap, which is subsequently digitized at 2048 Hz and stored.

Figure 4.1 illustrates a schematic overview of the system identification problem of interest. The human nervous system is excited with a robotic manipulation input sequence $\mathbf{U}_N$ and the resulting cortical activity measurements $\mathbf{Y}_N$ are subsequently used to model the nervous system of the ten participants. The wrist joint of each participant is evoked with seven different multisine input realizations, constructed with odd frequencies ranging from 1 Hz to 23 Hz. For each participant, the first six input realizations are used for modeling and the seventh is used for validation.

To extract the main cortical source activity needed for system modeling, Vlaar et al. [35, 36] used independent component analysis (ICA). The independent components are subsequently filtered with an ideal filter to remove wiring noise (50 Hz) and to remove all frequencies from 100 Hz onward. Finally, the signals are resampled to 256 Hz and the signal with the highest SNR is used for subsequent modeling. For each trial, the first and last three seconds are removed to reduce the effect of transient dynamics. The reader is referred to [35, 36] for a more detailed explanation of the data.

This experiment has the following objectives:

- The previous study performed by Vlaar et al. [35, 36] used an informative prior covariance matrix to estimate the involved parameters, which requires one to solve the non-convex minimization of the non-linear function described in eq. (2.21). This optimization, however, is a time consuming process and there is a possibility that the algorithm gets stuck in a local minimum. The first objective is to find a best performing (set of) model class(es) obtained with an uninformative prior during modeling which performs best during validation.

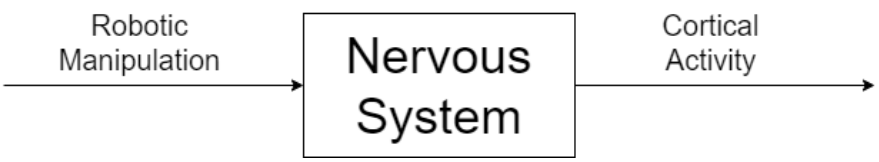


Figure 4.1: Schematic representation of the nervous system

- The Marginal Likelihood approach in eq. (2.21) not only finds a proper prior variance matrix, it provides as well a method to estimate the noise variance σ_ϵ based on the model structure and the input-output data by including it as a hyperparameter during optimization. While avoiding this minimization, the prior and the noise variance will have to be estimated in advance. Therefore, the second objective is to understand the effect of different values of the noise and prior variance during validation.

- The final objective is to compare the results of the model classes obtained with an uninformative prior with the model classes obtained with an informative prior.

4.2. Modeling Approach

This section provides the modeling approach of the cortical responses. First, in section 4.2.1 the modeling approach is elaborated for a single participant. This experiment is first done to understand how the algorithm responds to the cortical data. Second, in section 4.2.2 the general modeling method is elaborated for all the participants available in the dataset provided by Vlaar et al. [35, 36].

4.2.1. Single participant

During this experiment, the Bayesian Inference algorithm is applied to the first participant available in the dataset provided by Vlaar et al. [35, 36]. The competitive model class set is chosen in a similar fashion as described in section 3.2. That is, the model class set $\mathbf{M}$ is described as:

$$\begin{aligned} \mathcal{M}_{D,L} &= \{V(D,L) : D \in \mathbf{D}, L \in \mathbf{L}\} \in \mathbf{M} \\ \mathbf{D} &= [1,2] \\ \mathbf{L} &= [10,20,30,40,50,60]\,. \end{aligned} \tag{4.1}$$

Here, $V(D,L)$ denotes the Volterra model structure as described in section 2.1. First, the algorithm will be applied in an uninformative setting, which means that the prior and noise PDFs are estimated in advance and are defined as:

$$\epsilon(n) \sim \mathcal{N}(0, 0.5) \tag{4.2}$$

$$p(\mathbf{H}|\mathcal{M}_{D,L}) \sim \mathcal{N}\left(0, 5 \cdot \mathbf{I}_{n_H}\right). \tag{4.3}$$

Here, $\epsilon(n)$ denotes the noise corruption at time instance n and is considered to be stationary. Furthermore, $\mathbf{I}_{n_H}$ denotes the n_H-dimensional identity matrix, where n_H corresponds to the number of unique parameters in the Volterra kernel. Subsequently, the model classes are obtained with an informative prior as presented in section 2.2.1.

It is already been elaborated in section 3.2.2 that the used input sequence may cause problems in solving the Bayesian Inference. The goal of this experiment is to understand whether this has a influence on the performance of the model class set.

4.2.2. All participants

During this experiment, the Bayesian Inference using Volterra series algorithm is applied on the ten different participants of the dataset provided by Vlaar et al. [35, 36]. The competitive model class set is defined as:

$$\begin{aligned} \mathcal{M}_{D,L} &= \{V(D,L) : D \in \mathbf{D}, L \in \mathbf{L}\} \in \mathbf{M} \\ \mathbf{D} &= [1] \\ \mathbf{L} &= [10,20,30,40,50,60]\,. \end{aligned} \tag{4.4}$$

Here, the second degree Volterra model classes are omitted with respect to eq. (4.1), which is further explained in section 4.3.1. The algorithm will be applied in an informative setting. The hyperparameters $\theta_{\text{hp}} = [c_0, c_1, \alpha_1, \sigma_\epsilon]$ in eq. (2.21) are obtained using the multi-start (5 times) non-linear optimization technique Sequential Least Squares Programming (SLSQP) in Python.

The goal of this experiment is to examine the plausibility whether a first degree Volterra system, which corresponds to a Finite Impulse Response (FIR), is able to explain and predict cortical response data.

4.3. Results

This section provides the results of the Bayesian Inference algorithm on the cortical response data. First, section 4.3.1 provides the results of the algorithm on a single participant. Second, the results on all participants are given in section 4.2.2.

4.3.1. Single participant

This section provides the results of the Bayesian Inference algorithm on a single participant. First, the system is imposed with an uninformative prior. Subsequently, the model class set is obtained with an informative prior.

Uninformative Prior

Figure 4.2 illustrates the prior MCD for a single participant and fig. 4.3 shows the corresponding posterior MCD during validation. It can be seen that $\mathcal{M}_{2,60}$ performs best during modeling and validation, however, there is no claim being made that $\mathcal{M}_{2,60}$ is the actual true underlying system. This is because of the following reasons. First, the prior MCD strongly depends on the chosen values for the noise and prior variance. However, defining different values leads to a discrepancy between the prior MCD during system identification and the posterior MCD during validation, which means that the best performing model during modeling does not perform best during validation, indicating that this model class is overfitted. Second, with the current method, nothing can be said about the performance of the algorithm. The only statement being made is the relative probability of the model class $\mathcal{M}_{2,60}$ with respect to the competitive model class set. Despite that, it can be said that Bayesian Inference is able to find a model class during system identification which performs best during validation.

For reading purposes, fig. 4.4a depicts the final 250 time steps of $\mathcal{M}_{2,60}$ during the system identification phase and fig. 4.4b illustrates the predictions during validation. In both figures, the grey area denotes the σ, 2σ and 3σ uncertainty intervals. While modeling, the Volterra model structure is capable of tracking the cortical responses to a certain level of significance (fig. 4.4a), having a VAF of 69.69% and a RMSE of 0.54, and the observations $\mathbf{Y}_N$ remain within the uncertainty margins of the model. In contrast, $\mathcal{M}_{2,60}$ fails to track the cortical responses during validation, having a VAF of -45% and a RMSE of 1.33, and the measured output values lie in the low probability regions. These unstable results cast doubt on the credibility of model $\mathcal{M}_{2,60}$ being the true underlying system.

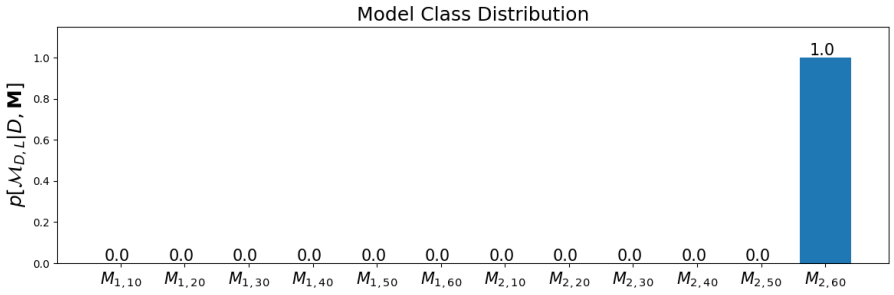


Figure 4.2: Prior MCD obtained with an uninformative prior

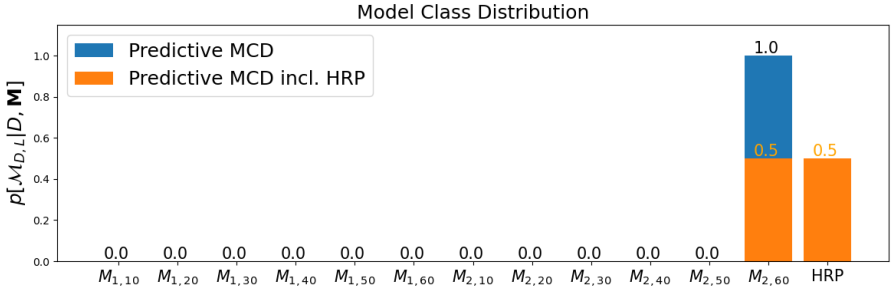
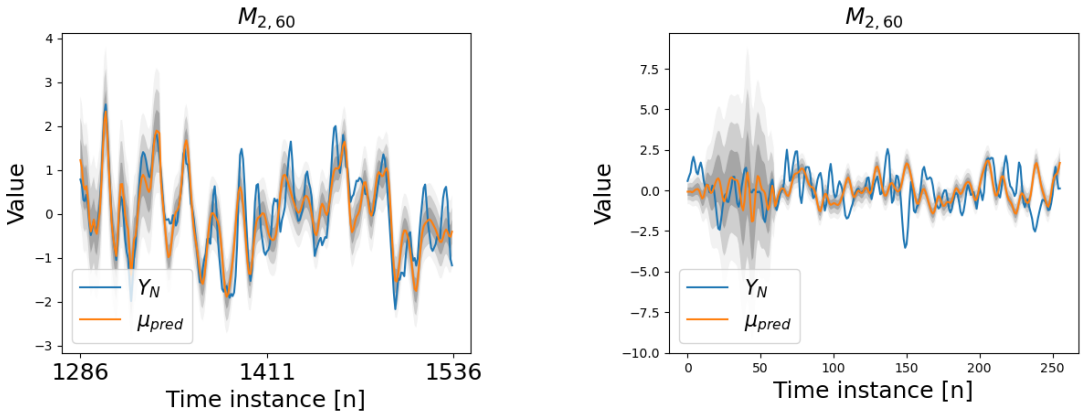


Figure 4.3: Predictive posterior MCD obtained with an uninformative prior



(a) Final 250 steps during system identification

(b) Validation sequence

Figure 4.4: Perforamce of $\mathcal{M}_{2,60}$

In section 3.3.2 it turned out that imposing the algorithm with an informative prior contributed significantly to the performance of the system. For that reason, in the next part the performance of the algorithm imposed with an informative prior is tested.

Informative Prior

An alternative method is to obtain the model class using an informative prior. Figures 4.5 and 4.6 illustrate the prior MCD during modeling and the posterior MCD during validation respectively. Equivalently as illustrated in fig. 4.2, using the informative prior, the prior MCD in fig. 4.5 is decisive in such a way that $\mathcal{M}_{2,60}$ is given full preference. However, looking at fig. 4.6, the first degree Volterra model classes, particularly $\mathcal{M}_{1,10}$, $\mathcal{M}_{1,50}$ and $\mathcal{M}_{1,60}$, appear to fit the validation data best compared to the model class set.

The provided results demonstrate the instability of the dataset which has been made available. It is discussed in section 3.2.2 that the input sequence $\mathbf{U}_N$ should contain enough information so that the Bayesian Inference algorithm does not depend much on the chosen prior. Figure 4.7 illustrates the singular values of $\mathbf{U}$

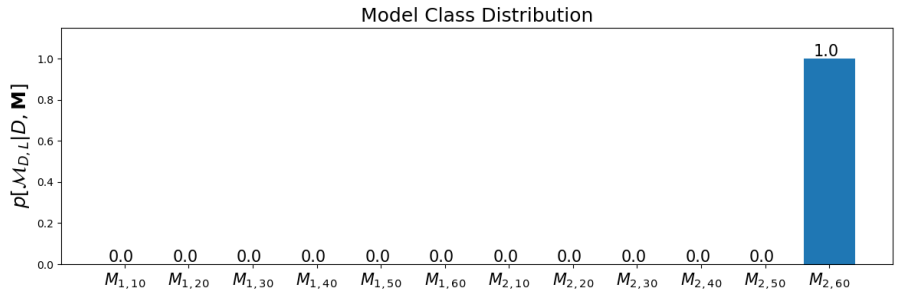


Figure 4.5: Prior MCD obtained with an informative prior

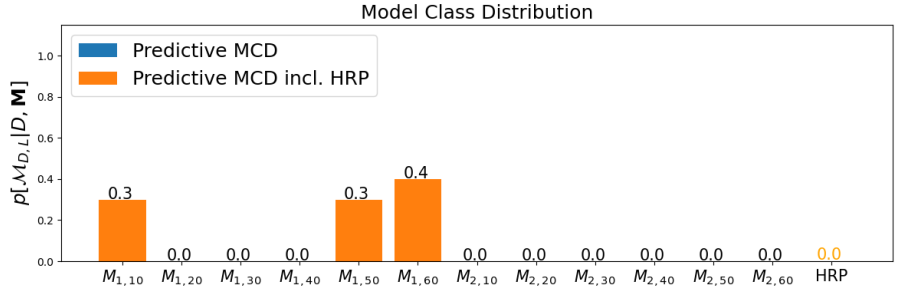
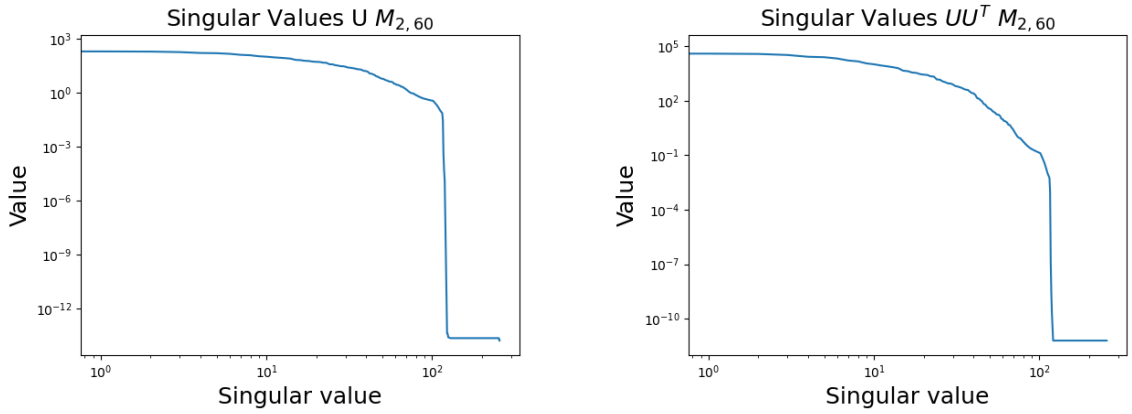


Figure 4.6: Predictive posterior MCD during validation



(a) Singular values $\mathbf{U}$

(b) Singular values $\mathbf{U}\mathbf{U}^T$

Figure 4.7: Singular values of the regression matrix of $\mathcal{M}_{2,60}$

(fig. 4.7a) and $\mathbf{U}\mathbf{U}^T$ (fig. 4.7b) for the model class $\mathcal{M}_{2,60}$. It can be seen that both matrices are close to singular, consequently meaning that the optimization problem does not yield an unique set of parameters.

Table 4.1 depicts the rank of the matrix $\mathbf{U}$ set out per model class. It turns out that this problem continues appearing for the model class $\mathcal{M}_{2,20}$ onward, since for each model class the regression matrix $\mathbf{U}$ is not full rank. Imposing an informative prior as proposed by Birpoutsoukis et al. [5, 6] and performed by Vlaar et al. [35, 36] has proven its success, however it requires the solution of the minimization of the non-linear and non-convex objective function as described in section 2.2.1. This objective function can include many local minima, which requires the optimization algorithm to be restarted several times. Even when this is done, there is no assurance that the global optimum has been found.

Having said this, the experiments for all the participants only include the first degree model classes in the model class set imposed with an informative prior. The hyperparameters of the prior variance in eq. (2.18) are obtained using a multi-start (5 times) non-linear optimization.

4.3.2. All participants

Tables 4.2 and 4.3 depict the prior and posterior MCD of the ten different participants respectively. The first column denotes the ten different subjects and each row represents the MCD for that subject, which sum up approximately to 1 (due to rounding errors). Each entry denotes the probability of the respective model class for that specific participant. The last column denotes the probability of the HRP. Recall that there is no claim being made that one of the model classes is the true underlying system. The numbers represent a relative probability with respect to the model class set. From tables 4.2 and 4.3 the following is observed:

Observation 1. During validation, HRP does not yield positive results for any of the participants compared

Table 4.1: Rank of $\mathbf{U}$ per model class

Model Class	Kernel size	Rank U
$\mathcal{M}_{1,10}$	11	11
$\mathcal{M}_{1,20}$	21	21
$\mathcal{M}_{1,30}$	31	31
$\mathcal{M}_{1,40}$	41	41
$\mathcal{M}_{1,50}$	51	51
$\mathcal{M}_{1,60}$	61	61
$\mathcal{M}_{2,10}$	66	66
$\mathcal{M}_{2,20}$	231	202
$\mathcal{M}_{2,30}$	496	309
$\mathcal{M}_{2,40}$	861	408
$\mathcal{M}_{2,50}$	1326	491
$\mathcal{M}_{2,60}$	1891	555

Table 4.2: Cortical responses: performance of candidate models obtained with an informative prior

#	$\mathcal{M}_{1,10}$	$\mathcal{M}_{1,20}$	$\mathcal{M}_{1,30}$	$\mathcal{M}_{1,40}$	$\mathcal{M}_{1,50}$	$\mathcal{M}_{1,60}$
1	0.00	0.00	0.00	0.00	0.30	0.70
2	0.00	0.13	0.21	0.22	0.22	0.22
3	0.17	0.17	0.17	0.17	0.17	0.17
4	0.00	0.00	0.00	0.00	1.00	0.00
5	0.00	0.00	0.00	0.17	0.47	0.36
6	0.17	0.17	0.17	0.17	0.17	0.17
7	0.00	0.00	0.61	0.00	0.39	0.00
8	0.00	0.00	0.55	0.00	0.00	0.45
9	0.00	0.00	0.00	0.00	0.00	1.00
10	0.17	0.17	0.17	0.17	0.17	0.17

Table 4.3: Cortical responses: performance of candidate models obtained with an informative prior

#	$\mathcal{M}_{1,10}$	$\mathcal{M}_{1,20}$	$\mathcal{M}_{1,30}$	$\mathcal{M}_{1,40}$	$\mathcal{M}_{1,50}$	$\mathcal{M}_{1,60}$	**HRP**
1	0.00	0.00	0.00	0.01	0.83	0.16	0.00
2	0.92	0.02	0.02	0.02	0.02	0.02	0.00
3	0.17	0.17	0.17	0.17	0.17	0.17	0.00
4	0.00	0.00	0.00	0.00	0.38	0.25	0.37
5	0.13	0.14	0.13	0.31	0.15	0.13	0.00
6	0.17	0.17	0.17	0.17	0.17	0.17	0.00
7	0.00	0.00	0.56	0.00	0.44	0.00	0.00
8	0.01	0.01	0.48	0.01	0.01	0.49	0.00
9	0.00	0.00	0.00	0.23	0.12	0.32	0.32
10	0.17	0.17	0.17	0.17	0.17	0.17	0.00

to continuing with the best performing model class during modeling.

Observation 2. The model classes with the highest probability while modeling for the participants 1,2, 5 and 8 are not the best performing model classes during validation. This indicates that these model classes are overfitted with false confidence.

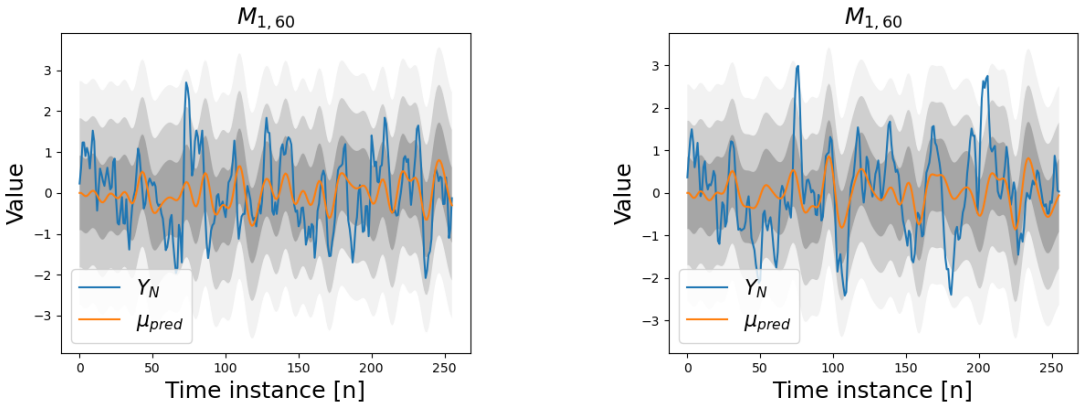
Observation 3. The prior MCD for the participants 3 and 6 are indecisive.

Observation 4. For the participants 4,7 and 9, the Bayesian Inference algorithm manages to find the model class during modeling which performs best during validation, namely $\mathcal{M}_{1,50}$, $\mathcal{M}_{1,30}$ and $\mathcal{M}_{1,60}$ respectively.

Having a best performing model class while modeling which performs best during validation gives confidence that the modeled system approaches the true underlying nervous system, however, fig. 4.4 has shown that this is not necessarily true. To test the plausibility of the results, the performance of model classes $\mathcal{M}_{1,50}$ and $\mathcal{M}_{1,60}$ of participants 4 and 9 respectively are further examined.

Figure 4.8 illustrates the validation sequence of $\mathcal{M}_{1,50}$ and $\mathcal{M}_{1,60}$ of the fourth and ninth participant respectively. Here, the dark, medium and light grey areas represent the σ, 2σ and 3σ uncertainty intervals respectively. The blue line represents the measured values $\mathbf{Y}_{\mathrm{N,val}}$ and the orange line denotes the mean of the predictive distribution. In both figures, it can be seen that the algorithm predicts a clear nonzero signal, where the measured values $\mathbf{Y}_{\mathrm{N,val}}$ remain within the uncertainty regions of the predictor. The question, however, is whether the residuals, that is the difference between the measured values $\mathbf{Y}_N$ and the mean of the predictions, exhibit the noisy behaviour as estimated during the hyperparameter optimizations. If the residual distribution approaches the noise distribution, it makes it plausible that the modeled system resembles the true underlying system. While modeling, it was assumed that the cortical responses are corrupted with white noise, which is defined as a normal distribution with stationary and finite variance σ_ϵ. This distribution has the following properties:

Property 1. The residuals are normally distributed with zero mean and finite variance σ_ϵ.



(a) Validation participant 4 for $\mathcal{M}_{1,50}$

(b) Validation participant 9 for $\mathcal{M}_{1,60}$

Figure 4.8: Validation sequence of the best performing models of participants 1 and 7

Property 2. The residuals have a constant spectrum, which means that the autocorrelation function is an impulse at lag zero. This is mathematicall defined as:

$$\mathbb{E}[\epsilon(n)\epsilon(n-\tau)] = \begin{cases} \sigma_\epsilon^2, & \tau = 0 \\ 0, & \text{otherwise.} \end{cases} \tag{4.5}$$

Property 3. The residuals and the input vector $\mathbf{U}_N$ are uncorrelated, which means that the relationship between the input and output data is fully embedded in the model class.

In figs. 4.9 to 4.11 the residuals are further investigated. The method used goes against the Bayesian philosophy,since the residuals are found by taking the difference of the mean of the predictive PDF and the measured values $\mathbf{Y}_N$, which yields a deterministic value. From a Bayesian perspective, the analysis of the residuals should be done in terms of distribution functions, as all residuals are a distribution functions. However, this frequentist approach has been used because it provides a good indication of how the best estimate of the distribution function is performing.

Figure 4.9 illustrates the comparison between the distribution of the residuals and the modeled noise PDF. Here, the x-axis denotes the different values of the residuals and the blue histogram illustrates the number of appearances per value. The red line represents the modeled noise PDF and the green line depicts the fitted distribution of the residuals during the validation. Furthermore, fig. 4.9a and fig. 4.9b illustrate the analysis for participant 4 and 9 respectively. From the figure it can be seen that, although the fitted PDF approaches the modeled noise distribution, both do not fully match. This indicates that there are dynamics active in the measured output signal $\mathbf{Y}_{\text{N,val}}$ which does not come from the noise.

This finding is substantiated in fig. 4.10. The figure illustrates the autocorrelation of the residuals for the two participants. In an ideal situation, the autocorrelation should have an peak at $\tau = 0$ with the height of the modeled variance of the noise and the correlation should maintain within the red 95% confidence bounds elsewhere. This interval represents the range of residual values that are for 95% insignificant and is found via [21]:

$$\text{conf} = \pm \frac{\sqrt{2}\text{erf}^{-1}(0.95)}{\sqrt{N}}, \tag{4.6}$$

where N denotes the sample length. For both participants, there is a clear spike at $\tau = 0$ in the autocorrelation function. This corresponds to an expected noise signal, however there are clear spikes present that exceed the confidence intervals. This indicates that there still remains a correlation between the residuals, therefore it can be said that the residual signal is not white.

Finally, the cross correlation between the input vector $\mathbf{U}_N$ and the residuals are illustrated in fig. 4.11. In an ideal situation, the autocorrelation function in the figures remain within confidence intervals, implying that the information embedded in the input sequence is fully captured by the model classes. However, it can be seen in fig. 4.11 that the cross correlation function exceeds the confidence intervals for both participants. When examining the outliers, keep in mind of the following points [18].

1. The correlation between $\epsilon(n)$ and $U_N(n-\tau)$ for negative τ, that is that the current residual affect future inputs, is an indication of output feedback. This does not mean that the model is faulty.

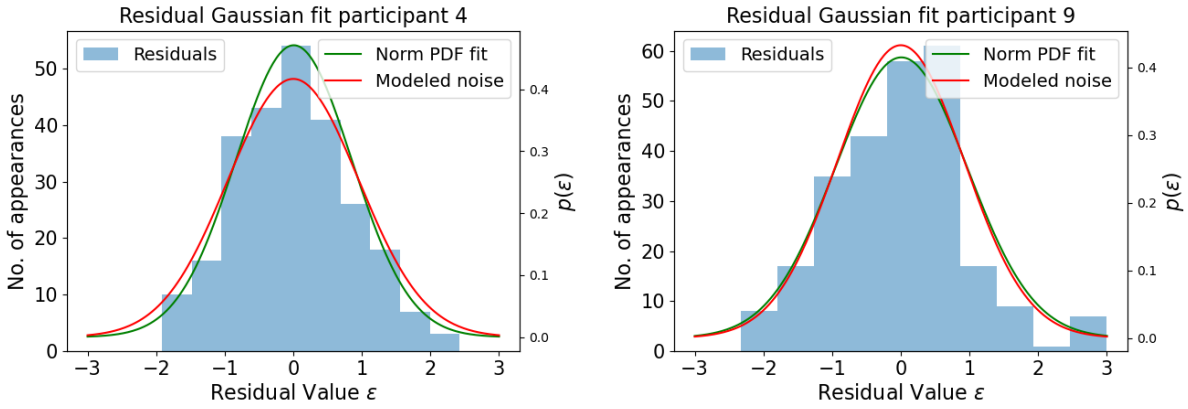
2. The correlation between ϵ and $U_N(n-\tau)$ for positive τ means that there are traces of the past input present in the current residuals, which indicates that the model class needs improvement.

For both participants, the confidence intervals are exceeded for positive lag, which means that the model class need improvement. During this study, experiments were done by extending the maximum lag of the proposed model classes, however similar results were obtained. Therefore it is assumed that higher order dynamics are present in the system.

4.4. Conclusion

This experiment has shown that with the current dataset it is difficult to design higher order Volterra model classes that are well able to predict the cortical response data, since the regression matrices are close to singular. The nervous system was then approached as a linear problem. Although it is not necessarily believed that a first degree Volterra Series can accurately describe the relationship between wrist joint manipulations and cortical responses, fig. 4.8a indicates the potential of the method used, as the noise corrupted observations $\mathbf{Y}_N$ maintain within the uncertainty intervals.

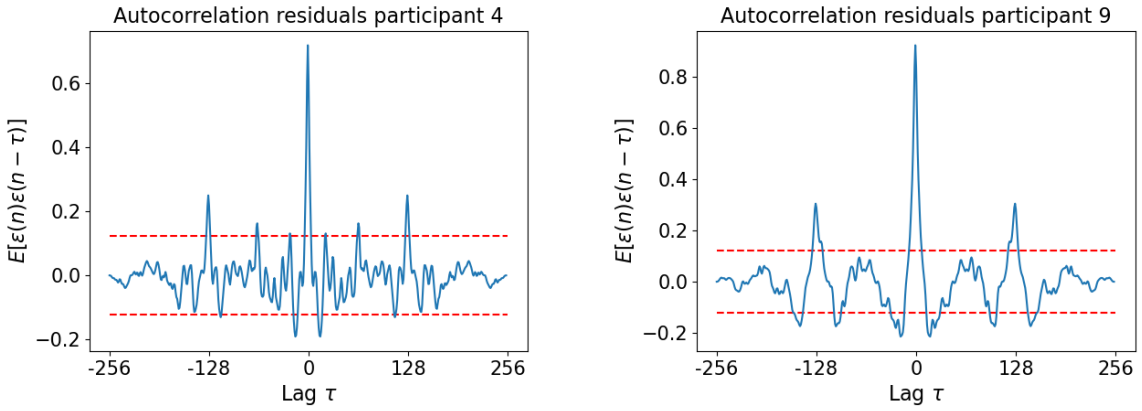
The experiments provided a method to analyse the plausibility of the chosen model classes, by examining the residuals. It is shown that the chosen model structures do not meet the complexity of the nervous system, because after modeling there was still a correlation between the input sequence and the residuals. While



(a) Participant 4

(b) Participant 9

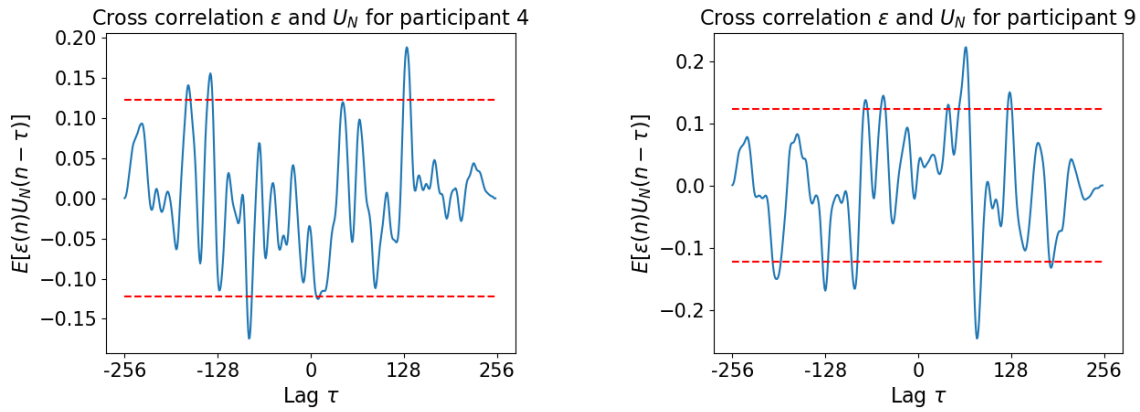
Figure 4.9: Autocorrelation of the residuals



(a) Participant 4

(b) Participant 9

Figure 4.10: Autocorrelation of the residuals



(a) Participant 4

(b) Participant 9

Figure 4.11: Cross correlation of the residuals with the input vector $\mathbf{U}_N$

redoing the experiments, it is of importance that the excitation signal is persistent for higher degree Volterra systems.

5

Discussion and Recommendations

5.1. Discussion

In this section the acquired results are discussed. First, section 5.1.1 discusses the independency of the model classes. Second, the specific choice for the input signal is evaluated in section 5.1.2. This section also provides an alternative input signal. The reflections on model averaging and the modeling approach are discussed in section 5.1.3 and section 5.1.3 respectively. Finally, the relation between the current study and previous studies is reviewed in section 5.1.5.

5.1.1. Independency of the Model Classes

While deriving the general equation for the HRP PDF, it is assumed that the parameters of two arbitrary different Volterra model classes in the competitive model class set are independent, i.e. $\mathbb{E}\left[\mathbf{H}_1\mathbf{H}_2^T\right] = \mathbb{E}\left[\mathbf{H}_1\right]\mathbb{E}\left[\mathbf{H}_2^T\right]$. However, this is a questionable assumption, since each higher degree Volterra series contains the lower degree Volterra Series with equivalent lag in its model structure. The same holds for the Volterra series with varying lag and constant degree; the Volterra Series with more lag contain the Volterra series with less lag in the model structure. According to the general equation for the covariance

$$\operatorname{cov}\left(\mathbf{H}_1, \mathbf{H}_2^T\right) = \mathbb{E}\left[\mathbf{H}_1\mathbf{H}_2^T\right] - \mathbb{E}\left[\mathbf{H}_1\right]\mathbb{E}\left[\mathbf{H}_2^T\right], \tag{5.1}$$

by ignoring this dependency, the covariance term is missing from the final derivation for the variance of the HRP PDF, consequently leading to false confidence.

5.1.2. Choice of Input Sequence

Capturing the non-linear dynamics using Volterra Series requires a different input sequence to perturb the wrist joint, such that the regression matrix $\mathbf{U}$ avoids singularity. It is shown in fig. 3.3 that a Gaussian White Noise input sequence contains a sufficient amount of information so that the higher order regression matrices avoid singularity, however, doing practical experiments one is often bounded by the physical limitations of the setup. For example, the actuator is not able to track high frequency components and the wrist joint has a certain range of motion. This makes it difficult to use a GWN as an input sequence in practice.

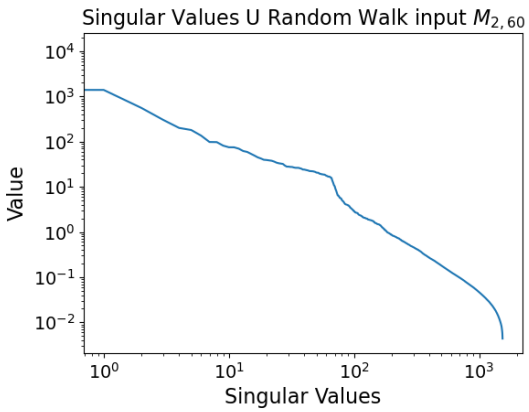
Figure 5.1a shows an alternative input sequence. This signal is defined as:

$$u_n = \mathcal{N}(u_{n-1}, 0.1) \quad u_{-1} = 0. \tag{5.2}$$

It is assumed that the robotic manipulator used by Vlaar et al. [35, 36] is able to track references up to 23 Hz, therefore the signal in eq. (5.2) is subsequently filtered with an ideal filter from 23 Hz onward. Finally, the input signal is scaled to $(-1, 1)$ such that it remains within the physical range of motion of the wrist joint.

Figure 5.1b illustrates the singular values of the proposed input sequence for the regression matrix of the second degree Volterra systems. It can be seen that the regression matrices contain more information compared to the regression matrix constructed with the input sequence proposed by Vlaar et al. [35, 36] (e.g. fig. 4.7). This provides more possibilities that the algorithm is able to find a relation between the cortical responses and the wrist joint manipulations.

(a) Random walk input sequence



(b) Singular Values of the Regression Matrix **U**

Figure 5.1: Alternative random walk input sequence

5.1.3. Reflection on Bayesian Model Averaging

This study has shown that Bayesian Model Averaging does not necessarily lead to improved results. Only when modeling the noise corrupted Neural Network, averaging over the competitive model class set led to improved performance during validation. However, repeating this experiment in which the noise sequence is regenerated, different results were obtained. This shows that Bayesian Model Averaging is unstable in the current setup, however this does not throw off its principle completely. As explained by Beck and Taflanidis [4], given that two model classes react similarly to a specific excitation sequence, does not imply that a different excitation signal yield in similar results. Therefore, when the experiment is repeated with a randomized input sequence, it cannot be ruled out that model averaging will yield better results.

Furthermore, it is shown as increasing noise corruption occurs, it is more likely that HRP is useful. The dataset used contained averaged input and output signals in order to reduce the noise corruption. By doing this, a lot of information was lost which was present in the averaged signals. It may be possible to avoid this step, however an alternative Bayesian parameter updating technique is needed described by Mackay [20], explained further in section 5.1.4. Doing this leads to more available information to construct the model parameters and an increased noise corruption.

5.1.4. Reflection on Modeling Approach

In the current study, the uninformative model class set is constructed by varying the degree and maximum lag of the Volterra Series. The Prior PDF and noise PDF variance must be determined in advance, which is difficult to estimate. While doing experiments, it appeared that the variance of the noise corruption influences the prior MCD, in a sense that lowering the variance inherently shifts the prior MCD favouring the Volterra Series of higher degree. Vice versa, this also meant increasing the noise variance shifts the probability to the model classes of lower degree. During the experiment it was assumed that the settings where the best performing prior model class corresponds with the best performing posterior model class is the most plausible. However, it would make sense to define to competitive model class set by varying the degree, lag, prior variance and noise variance, hence yielding a four dimensional model class set.

Second, the dataset is acquired by averaging the input and output signals across periods, in order to reduce the noise corruption. In addition, the original signals are resampled from 2048 Hz to 256 Hz. The original signal hence contains significantly more data samples, which makes it computationally expensive to compute posterior distributions, due to the inversion of large matrices. However, Mackay [20] provided a method to perform Bayesian model updating, which relies on the following updating rule for the posterior parameter PDF:

$$
\begin{aligned}
p(\mathbf{H}|\mathcal{D}\cap\mathcal{M}_{D,L}) &\sim \mathcal{N}\left(\mu_k, \Sigma_k\right) \\
\mu_k &= \left(\Sigma_{k-1}^{-1} + \mathbf{U}\Sigma_\epsilon^{-1}\mathbf{U}^T\right)^{-1}\left(\mathbf{U}\Sigma_\epsilon^{-1}\mathbf{Y}_N + \Sigma_{k-1}^{-1}\mu_{k-1}\right). \\
\Sigma_k &= \left(\Sigma_{k-1}^{-1} + \mathbf{U}\Sigma_\epsilon^{-1}\mathbf{U}^T\right)^{-1}
\end{aligned}
\tag{5.3}
$$

Here μ_k and Σ_k denote the mean and variance at time instance k respectively. This model updating method

allows one to use the extended dataset while suppressing the computational effort.

Third, in this study, it is assumed that the Volterra kernel is a Gaussian distribution *a priori*. This assumption ensured that the posterior distribution functions can be computed analytically, since the noise is normally distributed and the summation or multiplication of two Gaussian distributions yield another Gaussian distribution. However, it cannot be ruled out that the Volterra kernels are distributed differently. However, if one decides to alter the distribution type, different sampling techniques are required in order to find posterior distributions. This complicates the modeling process and increases the computational effort. Alternative distributions are for example a Laplacian, which is more sharpened around the mean compared to a Gaussian, consequently favoring the mean increasingly in advance. The interested reader is referred to [8, 19] for examples in Bayesian Inference using different prior distribution functions.

5.1.5. Relation with Previous Performed Studies

The results obtained in this study are different compared to previous studies performed by Vlaar et al. [36] and Tian et al. [32], which evaluated the model class performance during validation based on the VAF. First, Tian et al. [32] managed to obtain a model class which explained the validation data up to 95% (VAF). This model included lagged auto-regressive terms and both position and velocity as input to mimic the neuronal circuit. Furthermore, a Neural Network was implemented to cope for higher order dynamics. First, Tian et al. [32] specifically designed the model structure based on the human nervous system. In this study, the model classes were designed such that no prior knowledge is needed regarding the underlying structure of the system of interest. This choice ensures that the system contains less information and may result in decreased performance. Second, It has been shown that the excited input signal is not suitable for modeling second degree or higher Volterra models. Therefore, the assumption is made that the true underlying system is linear, which means that a larger portion of the measured signal is caused by noise. This suppresses the maximum achievable VAF. For this reason, the two studies are difficult to compare in terms of performance.

Vlaar et al. [36] obtained a second order Volterra Series combined with a Best Linear Approximation and managed to explain the validation data up to 60% (VAF). During this study, an attempt was made to mimic the results of Vlaar et al. [36] with the proposed method, based on the mean of the predictive PDF. However, the results fluctuated around 0% (VAF) and thus did not come close to the results obtained by Vlaar et al. [36]. The exact cause of this cannot be said with certainty, however the following difference between both approaches may contribute. The informative prior constructed in this study is less complex than the prior constructed by Vlaar et al. [36]. The latter designed the prior with more hyperparameters, which introduced an extra degree of freedom in the $\mathcal{U}$-axis in fig. 2.2. However, this increases the chance that the optimization algorithm gets stuck in a local minimum, hence requiring to restart the algorithm an increasing number of times in order to find a global minimum. During this study, the optimization algorithm is restarted with five different initial values, which might not be sufficient in order to find the global optimum.

Furthermore, Vlaar et al. [36] specifically mentioned that the choice of which 6 multisine realizations to use for modeling the model classes has had a significant impact on performance, which reveals the instability of the dataset. It is believed that the singular regression matrices play a major role in this.

5.2. Recommendations

This section contains the recommendations for future research. First, section 5.2.1 proposes a Tensor decomposition method in order to cope with the high dimensional parameters. Second, approaching the problem in the frequency domain is presented in section 5.2.2.

5.2.1. Tensor Decomposition for Higher Order Volterra Series

The number of parameters involved in the Volterra Series model structure grow exponentially with order. This means that the regression matrix $\mathbf{U}$ grows exponentially with order, making the calculation of the posterior PDFs computationally expensive. Batselier et al. [2] proposed the tensor decomposition method for Volterra Series, which allowed one to estimate approximately 1e9 parameters in 1.5 seconds.

Tensors represent multi-dimensional arrays that extend general matrix theory to higher orders. A d-way tensor $\mathcal{A}$ is described as $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \ldots \times n_d}$. The study executed by Batselier et al. [2] rephrased the system identification problem such that the Volterra tensors are never explicitly constructed, but stored in a effective Tensor Network. It appeared that both methods proposed by Batselier et al. [2] are highly effective, so it has the potential to be applied on Volterra series to model the cortical responses. Furthermore, tensor decompositions has proven its effectiveness in different fields, e.g. [1, 10].

Bayesian Inference is already applied to tensor factorization (e.g. [14]), however, to the best of my knowledge, Tensor based Bayesian Inference using Volterra Series has not been researched before. This would be an interesting topic to delve further into.

5.2.2. Bayesian Inference for Frequency Analysis

In the current study, all predictions are made in the time domain. However, existing studies partition EEG activity in the predefined frequency bands, such as the, among others, delta (1.5-3.5 Hz), theta (3.5-7.5 Hz) and alpha (7.5-12.5 Hz) [7] bands. Therefore, it would make sense to approach the problem in the frequency domain in stead of the time domain, which was also applied by Vlaar et al. [36], who used the frequency domain representation of the model class to relate the performance of the model to underlying ideas of physiological origin.

This method requires a general expression for the Volterra series in the frequency domain, which is already studied extensively in the literature (e.g. [12, 16]). Bayesian Inference applied in the frequency domain is a lesser known subject. The interested reader is referred to [37] for a more extensive explanation of the techniques to be used. To the best my, Bayesian Inference using Volterra Series to explain EEG data in the frequency domain is not been studied before.

5.2.3. Estimating the Model Class Distribution

During this study, the MCD is found by first defining a discrete competitive model class set. Subsequently, the model classes with the highest probability are used to make predictions. The found model classes, however, are always part of the initial competitive model class set, therefore if one does not pre-define this model class set *a priori* correctly, there is a good chance that the performance of the predictions will be disappointing, regardless of how the prior is designed, because the model class is not able to explain the complex dynamics.

Having said this, it would be interesting to investigate whether it is possible to either interpolate or extrapolate the knowledge we have of the discrete model class set so that we can conclude something regarding model classes that were initially not included in the competitive set. To the best of my knowledge, this topic is not researched so far.

6

Conclusion

The goal of this study was to obtain a Bayesian Volterra model capable of explaining cortical activity evoked by wrist joint manipulations. In order to achieve this, the study was subdivide in three sub-objectives.

> **Sub-objective 1.** Understanding the effect of incorporating uncertainty on the parameter estimation and the model selection process.

It turned out that the Bayesian model selection process was well able to reconstruct the ground truth model, as it assigned a decisive probability of 1 in the prior MCD to the corresponding model class. This model class performed as well best during validation, which was confirmed by the VAF, RMSE and the posterior MCD. In situations where the ground truth model structure was not included in the competitive model class, the algorithm did not manage to find the model class in the prior MCD which performed best during validation. In addition, the VAF and RMSE decreased significantly, namely 99.98% to 97.67% and 0.11 to 1.53 respectively.

> **Sub-objective 2.** Examining whether it is beneficial to perform Bayesian Model Averaging compared to conventional methods.

While modeling the Volterra ground truth system, Bayesian model averaging was not an issue, as only a single model was given full preference in the prior MCD (probability of 1). However, given that the ground truth system was not a Volterra Series, Bayesian Model Averaging yielded worse results, having a zero probability in the posterior MCD compared to the competitive model class set. It is shown that with increasing noise corruption, the chance of Bayesian Model Averaging leading to improved results increases. This is substantiated with the noisy Neural Network ground truth model, where the HRP model yielded higher probability in the posterior MCD compared to the single model class chosen with conventional methods.

> **Sub-objective 3.** Understanding the effect of imposing different prior Gaussian distributions on the performance of the algorithm.

During this study, different prior distributions were imposed to understand the effect on both the model selection process as well as the predictive performance. It appeared that, while modeling the Volterra ground truth system, imposing an informative prior led to improved results with respect to the uninformative prior in terms of VAF and RMSE. However, doing so distorted the prior MCD in such a way that it was no longer decisive. While modeling the Neural Network, it is shown that the prior has a major influence on the performance of the algorithm. By increasing the variance by a factor of 4, the Bayesian model selection process found a model class with increased performance in terms of RMSE, VAF and posterior MCD.

> **Main objective** The Development of Non-Linear Bayesian System Identification of the Cortical Response Evoked by Wrist Joint Manipulation Using Volterra Series.

In this study, it is proven to be difficult to design a non-linear model to describe the cortical responses evoked by wrist joint manipulations. Due to the multisine input sequence combined with the Volterra model structure, it is difficult to design an accurate second degree or higher Volterra system, since the respective regression matrices **U** are close to singular. In this study, it is shown that in these situations the algorithm is more

dependent on the specific choice of the prior. However, it has been shown that an informative prior does not guarantee successful results. For that reason a linear Volterra system has been modeled, which have shown the potential of the method used. The validation sequence for the first participant maintained within the uncertainty intervals of the predictive PDF, which offers opportunity to detect abnormalities in the brain waves based on the modeled output. On the other hand, by performing residual analysis, it has been shown that a higher order model is more likely to yield improved results, but this requires the experiment to be redone with an alternative input sequence, such as the proposed random walk input.

Bibliography

[1] Kim Batselier, Zhongming Chen, Haotian Liu, and Ngai Wong. A tensor-based volterra series black-box nonlinear system identification and simulation framework. *IEEE/ACM International Conference on Computer-Aided Design, Digest of Technical Papers, ICCAD*, 07-10-Nove, 2016. ISSN 10923152. doi: 10.1145/2966986.2966996.

[2] Kim Batselier, Zhongming Chen, and Ngai Wong. Tensor Network alternating linear scheme for MIMO Volterra system identification. *Automatica*, 84(August):26–35, 2017. ISSN 00051098. doi: 10.1016/j.automatica.2017.06.033.

[3] James L. Beck. Bayesian system identification based on probability logic. *Structural Control Health monitoring unit*, pages n/a–n/a, 2010. ISSN 15452255. doi: 10.1002/stc. URL http://dx.doi.org/10.1002/stc.456.

[4] James L. Beck and Alexandros A. Taflanidis. Prior and Posterior Robust Stochastic Predictions for Dynamical Systems Using Probability Logic. *International Journal for Uncertainty Quantification*, 3(4):271–288, 2013. ISSN 2152-5080. doi: 10.1615/int.j.uncertaintyquantification.2012003641.

[5] Georgios Birpoutsoukis, Anna Marconato, John Lataire, and Johan Schoukens. Regularized nonparametric Volterra kernel estimation. *Automatica*, 82:324–327, 2017. ISSN 00051098. doi: 10.1016/j.automatica.2017.04.014. URL http://dx.doi.org/10.1016/j.automatica.2017.04.014.

[6] Georgios Birpoutsoukis, Péter Zoltán Csurcsia, and Johan Schoukens. Efficient multidimensional regularization for Volterra series estimation. *Mechanical Systems and Signal Processing*, 104:896–914, 2018. ISSN 10961216. doi: 10.1016/j.ymssp.2017.10.007.

[7] Ismael Clark, Rolando Biscay, Maribel Echeverría, and Trinidad Virués. Multiresolution decomposition of non-stationary eeg signals: A preliminary study. *Computers in Biology and Medicine*, 25(4):373–382, 1995. ISSN 00104825. doi: 10.1016/0010-4825(95)00014-U.

[8] Facundo Costa, Hadj Batatia, Thomas Oberlin, Carlos D'Giano, and Jean Yves Tourneret. Bayesian EEG source localization using a structured sparsity prior. *NeuroImage*, 144:142–152, 2017. ISSN 10959572. doi: 10.1016/j.neuroimage.2016.08.064. URL http://dx.doi.org/10.1016/j.neuroimage.2016.08.064.

[9] John Duchi. Derivations for Linear Algebra and Optimization. *Berkeley, California*, pages 1–13, 2007. URL http://stanford.edu/~jduchi/projects/general_notes.pdf.

[10] Gérard Favier and Thomas Bouilloc. Parametric complexity reduction of volterra models using tensor decompositions. *European Signal Processing Conference*, (Eusipco):2288–2292, 2009. ISSN 22195491.

[11] Charles J Geyer and Elizabeth A Thompson. Annealing Markov Chain Monte Carlo with Applications to Ancestral Inference. 90(September, 1995):909–920, 1995. doi: 10.1080/01621459.1995.10476590.

[12] S.Torkel Glad. Nonlinear system theory. *Automatica*, 23(4):545–546, 1987. ISSN 00051098. doi: 10.1016/0005-1098(87)90085-9.

[13] P. L. Green, E. J. Cross, and K. Worden. Bayesian system identification of dynamical systems using highly informative training data. *Mechanical Systems and Signal Processing*, 56:109–122, 2015. ISSN 10961216. doi: 10.1016/j.ymssp.2014.10.003. URL http://dx.doi.org/10.1016/j.ymssp.2014.10.003.

[14] Cole Hawkins. Variational Bayesian Inference for Robust Streaming Tensor Factorization and Completion.

[15] Jennifer A. Hoeting, David Madigan, Adrian E. Raftery, and Chris T. Volinsky. Bayesian model averaging: A tutorial. *Statistical Science*, 14(4):382–401, 1999. ISSN 08834237. doi: 10.1214/ss/1009212519.

[16] Xingjian Jing. *Frequency Domain Analysis and Design of Nonlinear Systems based on Volterra Series Expansion: A Parametric Characteristic Approach*. 2015. ISBN 978-331-91239-1-2. URL https://doi.org/10.1007/978-3-319-12391-2.

[17] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22 (1):79–86, 1951. ISSN 0003-4851. doi: 10.1214/aoms/1177729694.

[18] Lennart Ljung. *System Identification: Theory for the User*. NEw Jersey: Prentice-Hall, 2 edition, 1999. ISBN 9780136566953.

[19] Felix Lucka, Sampsa Pursiainen, Martin Burger, and Carsten H. Wolters. Hierarchical Bayesian inference for the EEG inverse problem using realistic FE head models: Depth localization and source separation for focal primary currents. *NeuroImage*, 61(4):1364–1382, 2012. ISSN 10538119. doi: 10.1016/j.neuroimage.2012.04.017. URL http://dx.doi.org/10.1016/j.neuroimage.2012.04.017.

[20] David J C Mackay. *Information Theory, Inference and Learning Algorithms*. 2007. ISBN 0521642981. URL papers2://publication/uuid/85B84725-0CB2-4E5D-892C-E8340F725CE6.

[21] Contact Mathworks. Image Processing Toolbox ™ User ' s Guide R 2014 b. 2014.

[22] Matthew Muto and James L. Beck. Bayesian updating and model class selection for hysteretic structural models using stochastic simulation. *JVC/Journal of Vibration and Control*, 14(1-2):7–34, 2008. ISSN 10775463. doi: 10.1177/1077546307079400.

[23] G. Pillonetto and A. Chiuso. Tuning complexity in kernel-based linear system identification: The robustness of the marginal likelihood estimator. *2014 European Control Conference, ECC 2014*, pages 2386–2391, 2014. doi: 10.1109/ECC.2014.6862629.

[24] Gianluigi Pillonetto, Minh Ha Quang, and Alessandro Chiuso. A new kernel-based approach for nonlinearsystem identification. *IEEE Transactions on Automatic Control*, 56(12):2825–2840, 2011. ISSN 00189286. doi: 10.1109/TAC.2011.2131830.

[25] Adrian E. Raftery, David Madigan, and Jennifer A. Hoeting. Bayesian Model Averaging for Linear Regression Models. *Journal of the American Statistical Association*, 92(437):179–191, 1997. ISSN 1537274X. doi: 10.1080/01621459.1997.10473615.

[26] Gareth O. Roberts and Jeffrey S. Rosenthal. Examples of adaptive MCMC. *Journal of Computational and Graphical Statistics*, 18(2):349–367, 2009. ISSN 10618600. doi: 10.1198/jcgs.2009.06134.

[27] Simo Särkkä. *Bayesian Filtering and Smoothing*. 2013. ISBN 9781107030657. doi: 10.1176/pn.39.24.00390022b.

[28] Matthias Seeger, Florian Steinke, and Koji Tsuda. Bayesian inference and optimal design in the sparse linear model. *Journal of Machine Learning Research*, 2:444–451, 2007. ISSN 15324435.

[29] C.E. Shannon. A Mathematical Theory of Communication. *Journal of the Franklin Institute*, 27(3), 1948. ISSN 00160032. doi: 10.1016/s0016-0032(23)90506-5.

[30] Jeremy G. Stoddard, James S. Welsh, and Håkan Hjalmarsson. EM-based hyperparameter optimization for regularized volterra kernel estimation. *IEEE Control Systems Letters*, 1(2):388–393, 2017. ISSN 24751456. doi: 10.1109/LCSYS.2017.2719766.

[31] Jing Tian and Kai Kuang Ma. A MCMC approach for bayesian super-resolution image reconstruction. *Proceedings - International Conference on Image Processing, ICIP*, 1(2):45–48, 2005. ISSN 15224880. doi: 10.1109/ICIP.2005.1529683.

[32] Runfeng Tian, Yuan Yang, Frans C.T. van der Helm, and Julius P.A. Dewald. A novel approach for modeling neural responses to joint perturbations using the NARMAX method and a hierarchical neural network. *Frontiers in Computational Neuroscience*, 12(December):1–8, 2018. ISSN 16625188. doi: 10.3389/fncom.2018.00096.

[33] Nelson J. Trujillo-Barreto, Eduardo Aubert-Vázquez, and Pedro A. Valdés-Sosa. Bayesian model averaging in EEG/MEG imaging. *NeuroImage*, 21(4):1300–1319, 2004. ISSN 10538119. doi: 10.1016/j.neuroimage.2003.11.008.

[34] Verdult Vincent Verhaegen, Michel. *Filtering and System Identification*, volume 53. 2019. ISBN 9788578110796. doi: 10.1017/CBO9781107415324.004.

[35] Martijn P. Vlaar, Teodoro Solis-Escalante, Alistair N. Vardy, Frans C.T. Van Der Helm, and Alfred C. Schouten. Quantifying nonlinear contributions to cortical responses evoked by continuous wrist manipulation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(5):481–491, 2017. ISSN 15344320. doi: 10.1109/TNSRE.2016.2579118.

[36] Martijn P. Vlaar, Georgios Birpoutsoukis, John Lataire, Maarten Schoukens, Alfred C. Schouten, Johan Schoukens, and Frans C.T. Van Der Helm. Modeling the Nonlinear Cortical Response in EEG Evoked by Wrist Joint Manipulation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(1):205–215, 2018. ISSN 15344320. doi: 10.1109/TNSRE.2017.2751650.

[37] Yang Bingzheng and Liu Xiaoyang. *Nonlinear System Identification.*, volume 4. 1986. ISBN 9781119943594. doi: 10.1016/s0005-1098(02)00239-x.