System Identification using Dynamic Expectation Maximization

From neuroscientific principle towards filtering and identification under the presence of correlated noise

Laurens Žnidaršič

Master of Science Thesis

Delft University of Technology

Department of BioMechanical Engineering
Delft Center for Systems and Control

System Identification using Dynamic Expectation Maximization

From neuroscientific principle towards filtering and identification under the presence of correlated noise

Master of Science Thesis

For the double degree of Master of Science in

Systems and Control

and

Mechanical Engineering

at Delft University of Technology

Laurens Žnidaršič

July 30, 2020

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Delft University of Technology
Delft Center for Systems and Control (dcsc)
and
Department of BioMechanical Engineering (BMechE)

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis entitled

System Identification using Dynamic Expectation Maximization

by

Laurens Žnidaršič

in partial fulfillment of the requirements for the degree of

Master of Science Systems and Control
and
Master of Science Mechanical Engineering

Dated: July 30, 2020

Supervisor(s):

prof. dr. ir. M. Wisse

dr. ir. P. Mohajerin Esfahani

Reader(s):

prof. dr. R. Babuska

A. A. Meera

Abstract

A fundamental task of intelligent and autonomous robots is to infer from observations the state of the world. This inference is generally achieved by employing a filter, which consists of a model and filtering law. Learning this model and filtering law from observations is another fundamental part of robotics, and is generally referred to as system identification.

Neuroscientist K.J. Friston has developed a relatively novel theory on biologically plausible human brain inference called the Free-Energy Principle. One of the theories within the Free-Energy Principle, namely that Dynamic Expectation Maximization (DEM), has been suggested as a novel method for filtering and system identification. This method is expected to outperform standard Expectation Maximization (EM) in terms of hidden state and parameter estimation in settings where noise is correlated. However, in order for this neuroscientific theory to be properly used for robot inference, two problems must first be solved.

The first of these problems is the fact that the theory is defined in the continuous-time domain, whereas data available for system identification is always discrete. In this thesis I will suggest three discrete-time interpretations for DEM-based system identification under the presence of coloured noise. The major difference between the three methods is the information that is embedded in the generalized signals: predictions, derivatives and past data.

The second problem is that the filtering method corresponding to the Free-Energy Principle depends on data which is not available: the derivative signals of measured in- and outputs. In this thesis I will describe two fundamentally different solutions to this feasibility issue: a numerical differentiator and a stable filter. Both of these solutions are shown to find an estimate for the unavailable data, but the former is shown to significantly outperform the latter.

The theory described in this thesis will be put in practice parallel to the research, by implementing it into a python toolbox for system identification. This toolbox can be used as a basis for further research and be approved along with it, until at some point it is ready to be used for real applications.

Using the toolbox, the DEM-based identification and filtering methods are tested though various numerical simulations and the results are compared with the EM method. Results show that with the implemented settings none of the suggested discrete-time filtering methods outperforms the conventional Kalman filter. The main cause of this inferior performance is shown to be instability in the filtering method. I make some suggestions for overcoming this problem. As a result of the inferior performance, the joint-performance of the suggested DEM-based parameter- and state- estimation methods also proves to be inferior in terms of parameter estimation accuracy.

However, results show that the theoretical parameter optima of the Free-Energy as determined from known hidden states are in fact close on the real parameters, and furthermore show to be invariant to noise correlation. This suggests that should the instability issue as some point be solved and a better means to approximate the theoretical optimum be found, the DEM-based methods might in fact outperform EM both in terms of hidden-state and parameter accuracy in settings with correlated noise.

Table of Contents

List of Figures

Glossary

List of Acronyms

EM	expectation maximization
DEM	dynamic expectation maximization
LTI	linear time invariant
SS	state space
CT	continuous-time
DT	discrete-time
FFT	fast Fourier transform
FE	Free-Energy
FEP	Free-Energy principle
KF	Kalman filter
MSE	mean squared error
ML	maximum-likelihood
LL	log-likelihood
PSD	power spectral density
GF	Generalized filter

List of Notations

a	Scalar variable $\in \mathbb{R}$
$\boldsymbol{a}$	Vector of variables $\in \mathbb{R}^n$ or $\mathbb{R}^{1\times n}$
$\mathbf{A}$	Matrix of variables $\in \mathbb{R}^{m\times n}$
$a(t)$	variable a at time t in continuous time
$a[k]$	variable a at time-step k in discrete time s.t. $k \times \Delta\mathrm{t} = t$
$\dot{a}$, $\ddot{a}$ and $a^{(i)}$	1^{st}, 2^{nd} and i^{th} time derivative of a.
a^i	a raised to the power of i
$\hat{a}$	Prior estimation of the unknown variable a
a'	Posterior estimation of the unknown variable a
a^*	Globally optimal solution of, i.e. $a^* := \arg\min_a J(a)$
$\tilde{a}$	Generalized a, i.e. $\tilde{a} := \begin{bmatrix}\dot{a} & \ddot{a} & \cdots & a^{(p)}\end{bmatrix}^\top$ (see ch. 4)
$\partial_a f(a)$ or $\frac{\partial f(a)}{\partial a}$	Partial derivative of $f(a, b)$ towards a
$\boldsymbol{a} \frown \mathcal{N}(\boldsymbol{m}, \mathbf{A})$	Normally distributed random vector $\boldsymbol{a}$ with mean $\boldsymbol{m}$ and covariance matrix $\mathbf{A}$

List of Variables

l, m and n	input-, output- and state dimensionality
p	Embedding order of generalized system
N	length of data-sequence s.t. $k \in \{\mathbb{Z} \vert 1 \leq k \leq N\}$
T	end time of data sequence s.t. $t \in \{\mathbb{R} \vert 0 \leq t \leq T\}$
$\boldsymbol{u}$	System input
$\boldsymbol{x}$	System hidden state
$\boldsymbol{y}$	System output
$\boldsymbol{w}$	System state noise
$\boldsymbol{z}$	System output noise
$\boldsymbol{\theta}$	Vector containing all unknown parameters of a real system
$\mathbf{A}_c$, $\mathbf{B}_c$, $\mathbf{C}_c$, $\mathbf{D}_c$,	Continuous time LTI SS matrices
$\mathbf{A}_d$, $\mathbf{B}_d$, $\mathbf{C}_d$, $\mathbf{D}_d$,	Discrete time LTI SS matrices
$\boldsymbol{\mathcal{D}}$	Shift operator (see ch. 4)

Acknowledgements

Firstly, I would like to thank both my supervisors, prof. dr. ir. Martijn Wisse and dr. ir. Peyman Mohajerin Esfahani. Thank you for pushing me to make the absolute most out of this research within the available time and for assisting me content-wise. Also, I very much enjoyed our intricate discussions and cannot begin to explain how valuable they have been to this thesis.

Secondly, Ajith: thank you for your help on understanding DEM and the Free-Enery Principle from the fundamentals to the tiny details. Our discussions, both those via e-mail and those in person have been of great value both for my own understanding of DEM and for this thesis.

Thirdly, I would like to thank my girlfriend. I realize that there have been days where I have not been able to fully shield you from my stress. Even stronger, circumstance lead to us being on facing sides of our dining table 40 hours a week. Therefore: Aletta, thank you for keeping a cool head and being my emotional anchor throughout of this research even though being literally at the front line of my frustration.

Finally, I would like to thank my parents. Thank you for continuously reminding me that I am still a student working towards a master's degree, which can only be acquired by handing in my thesis at some point. Without your gentle pressure I might very well have kept adding subjects to my research until the end of times.

Delft, University of Technology
July 30, 2020

Laurens Žnidaršič

“People think of education as something that they can finish”

-*Isaac Asimov*

Chapter 1

Introducton

In this introductory I will explain the motivation behind this research. Furthermore, I will state the sub-questions and the main question that this thesis addresses. The chapter is concluded with the outline of the document.

1-1 Context and motivation

We live in a world that is in an ongoing search for more intelligent and autonomous robots, at the purpose of outsourcing ever more work to robots such that we, the humans, can focus on the more important things in life, or just for the sake of pushing the limits of science and engineering.

At some point in the far future we will have created robots that are fully autonomous, let's call it the holy grail of robotics. The most obvious condition for considering a robot to be truly autonomous is that it must be able to perform tasks without any human intervention. This will include advanced control strategies that are robust, fault-tolerant, safe and energy-efficient, and so on. However, it will also include the robots ability to observe, interpret and understand itself and its surroundings.

In other words, the robot will, apart from a controller, need to include a filter. This filter will be based on a model, i.e. a belief on how the world produces observed data, and a filtering law, i.e. a mechanism which separates noise from information. This immediately raises the question on how the robot obtained this model and filtering law, which brings me to the last principle that any agent adhering to my definition of truly autonomous must include: model learning, or system identification, as it is more commonly referred to.

This thesis is on the latter two of the strategies. More specifically, it covers filtering and system identification in robotics in environments where the noise the world generates is not completely random, i.e. where subsequent noise samples are correlated. It has been shown that such types of noise are ubiquitous to many real-world processes, yet filtering and system-identification methods which can handle such noise are not.

Therefore, an important step towards the holy grail of robotics is a method for filtering and identification under the presence of coloured noise. In this thesis I will build on earlier work by world-renowned neuroscientist K.J. Friston: the theory of Dynamic Expectation Maximization (DEM). Ultimately I formulate a DEM-based filtering and identification method which should outperform the conventional method of Expectation Maximization (EM) in cases where systems are perturbed with correlated noise.

Parallel to performing this research, I will implement both EM and DEM to a python toolbox for system identification in robotics (SIR). The toolbox serves the dual purpose of providing results for this thesis and building the first foundations of a universal toolbox for system identification under the presence of any kind of noise and on any kind of system.

1-2 Research questions

Before arriving at such a method, It is important to have a thorough understanding on what the conventional EM method is, how it works and more importantly when it does not work, i.e. what are its limitations. Furthermore, a thorough understanding on the subject of random noise is needed, and more importantly what correlated random noise is, such that we can understand what causes the limitations.

> **Question 1:** Why is expectation maximization (EM) the state-of-the art method?

> **Question 2:** What are the theoretical principles which define EM?

> **Question 3:** What causes the limitations of EM?

When this knowledge has been established, the novel method can be considered. First, by considering the neuroscientific theory, and then on how it can be translated into an applicable system identification method.

> **Question 4:** What are the theoretical principles which define dynamic expectation maximization (DEM)?

> **Question 5:** How can DEM be translated into a method for filtering and system identification?

This translation will provide some feasibility issues which will then need to be solved. Furthermore, these performance of these solutions must be validated.

> **Question 6:** What feasibility problems arise when translating DEM to a filtering and system identification method?

> **Question 7:** How can these problems be solved?

> **Question 8:** Do the proposed solutions to the feasibility problems perform accurately?

Lastly the proposed method must be tested for filtering and identification performance under the presence of coloured noise, and compared with the state-of-the-art method

> **Question 9:** Does DEM outperform EM w.r.t. filtering under the presence correlated noise

> **Question 10:** Does DEM outperform EM w.r.t. identification under the presence correlated noise

Only after answering each and every one of these research questions, can we tend to the main question:

> **Main question** Do the DEM-based methods for filtering and system identification as suggested in this thesis outperform EM for systems perturbed with correlated noise?

1-3 Outline

Chapter 2 will concern the general concept of correlated random noise, providing a thorough theoretical foundation for understanding the limitations of the method discussed in chapter 3.

In chapter 3 I will discuss the conventional method for system identification: EM. More specifically, I will explain what theoretical principles drive the method, how it is defined mathematically and what it's limitations are, providing an answer to the research questions 1 and 2. Combining these answers with the knowledge from chapter 2 provides an answer to research question 3.

Chapter 4 will cover the subject of DEM as proposed by K.J. Friston and a systematic translation of the theory towards a DEM-based discrete-time system-identification method, which answers the research questions 4 and 5.

The method proposed in chapter 4 still includes a feasibility problem. In chapter 5 I will propose methods for solving this problems, thus answering research questions 6 and 7. Furthermore, I will numerically evaluate the performance of these solutions, such that research question 8 is answered.

In chapter 6 I will evaluate the performance of the DEM-based methods by comparing it with EM, with respect to both filtering and identification performance. At the end of this chapter I will have answered research questions 9 and 10.

Finally, chapter 7 will circle back to the main question, providing the conclusion of this thesis. Furthermore, it will provide a short summary of the main findings of this thesis, an overview of the research contributions of this thesis and recommendations for further research.

Chapter 2

Noise

In this chapter I will describe the general concept of noise and the different types of noise that are commonly assumed for filter design. Though fundamental to the problems of filtering and system identification, and specifically how the dynamic expectation maximization algorithm might outperform the expectation maximization algorithm, the chapter will not directly answer any of the research questions. It will, however, provide the necessary theoretical foundation of the answers that will be provided in subsequent chapters

2-1 Random noise signals

On a highest level of abstraction, any input signal which can not be directly controlled can formally be considered as noise. These noise signals can have any number of characteristics, which are determined by the way the noises are generated, but in most common scenarios the exact signal is unknown and includes some level of randomness.

White noise The most commonly assumed type of random noise is that where each subsequent sample is an independent random pick from a certain probability distribution. As a consequence, each sample is completely independent from the previous one, i.e. each sample is uncorrelated with previous samples. This specific property is generally referred to as White noise [1], which is proven to have a flat frequency spectrum and an impulse-autocorrelation function Figure 2-1.

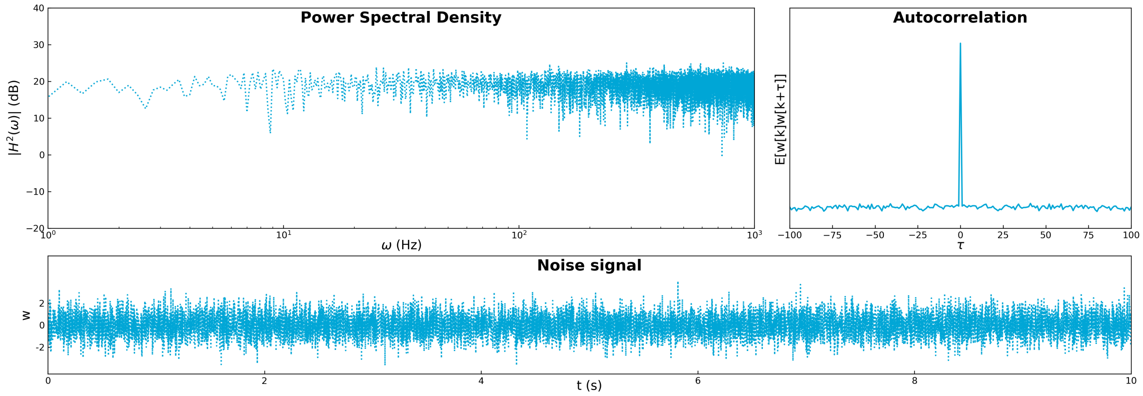

Figure 2-1: A White noise signal, its autocorrelation and PSD. The figure clearly depicts the flat frequency power spectrum and the impulse function autocorrelation of a White noise signal. $\Delta t = 10^{-3}$ s. $T = 10$ s. Source: github.com/lznidarsic/sir/

Distributions The most commonly assumed probability distribution for random noise is the Gaussian or Normal distribution, which has two sufficient statistics: the mean and the (co-)variance. The reason that this distribution is so commonly assumed, is mainly because it has been empirically shown that many real world process noise eventually converges towards something close to a Gaussian distribution. Since all other theory discussed in this thesis will be referring to signals drawn from a Gaussian distribution, I will not cover the topic of distributions explicitly.

2-2 Correlated noise signals

Though there are many real world noise processes for which White noise has been shown to be a good explanation, there are at least as many noises processes for which it is too simple of an explanation and thus inaccurate. These processes, which generally include some internal dynamics, are signals in which samples are not fully independent of one another, i.e. they are partly determined by the samples preceding them[1]. In other words, signals in which the samples are correlated through time.[2]

2-2-1 Coloured noise

A category of correlated noise signals that is common to the fields of signal processing and has been proven to be a good explanation for many real world process noises, is that of coloured noise [2–4]. The subcategorization of coloured noise is based on the slope of the spectral power spectral density (PSD) of the noise signal realizations and includes five colours[3], including White, as can be observed in Tab. 2-1.

Table 2-1: the colours of noise

Colour	**Slope**
Red	-20 dB/dec
Pink	-10 dB/dec
White	0 dB/dec
Violet	10 dB/dec
Blue	20 dB/dec

Only considering the PSD of these correlated noise categories provides very little insight into how such signals might appear, let alone how they can be generated. However, general linear systems theory states that the transformation between a slope of $0^{\text{dB}}/_{\text{dec}}$ and a slope of $-20^{\text{dB}}/_{\text{dec}}$ on any signal can be in fact be achieved by causal integration of the original signal [5]. In other words: causally integrating a White noise signal yields a Red noise signal[4].

[1] Or even those succeeding them, in which case the correlation is non-causal. It should be evident that such scenarios do not exist outside of simulation

[2] In Neuroscientific and Reinforcement Learning literature this kind of signals are said to not have the 'Markov Property', which is in fact equivalent

[3] The reason these noise types are categorized as colours, is because of how such noise signals would appear if they were signals of light. E.g. Violet noise emphasises high-frequencies, and the highest frequency of visible light appears violet

[4] Which is why Red noise is often referred to a Brown or Brownian noise, after Brownian or random walk motion [1]

A similar argument can be made for Violet noise, which is generated by differentiation of a White noise signal. Figure 2-3 and Figure 2-2 show the spectral and correlation data, the latter of which approximate the functions for numerical integration and differentiation respectively.

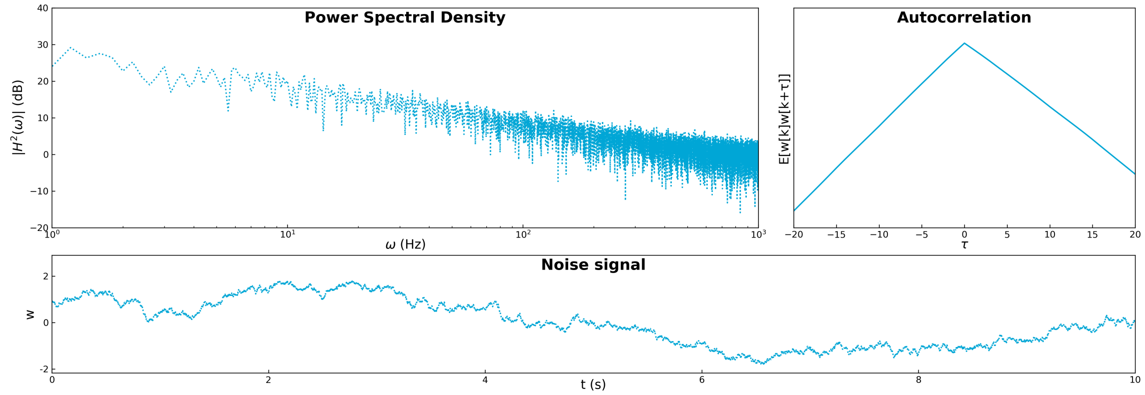

Figure 2-2: A Red noise signal, its autocorrelation and PSD. The figure clearly depicts the -20 $^{\mathrm{dB}}/_{\mathrm{dec}}$ power spectrum of the Red noise signal. The autocorrelation function approximates $-|\tau|$, but cannot be accurately estimated due to the estimation method relying on the wide-sense stationarity[5] property, which Red noise lacks. $\Delta t = 10^{-3}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

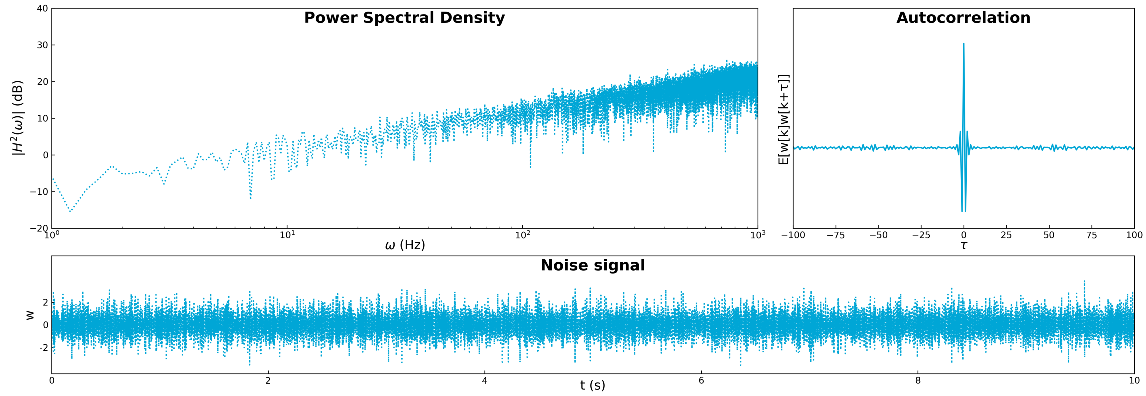

Figure 2-3: A Violet noise signal, its autocorrelation and PSD. The figure clearly depicts the 20 $^{\mathrm{dB}}/_{\mathrm{dec}}$ power spectrum of the Violet noise signal. The autocorrelation function approximates a finite-order central-difference filter, but the exact shape will depend on the method that was used for signal differentiation: in this case the noise was generated by frequency-domain operation, and therefore the shape is party driven by the Fast-Fourier transform algorithm . $\Delta t = 10^{-3}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

Even though linear systems theory provides very intuitive explanations for Red and Violet noise, it also immediately reveals that that such definitions cannot be stated for Pink and Blue noises, as there are no linear operations which yield relative slopes of $\pm 10^{\mathrm{dB}}/_{\mathrm{dec}}$ in the spectral power domain[6].

Therefore, the easiest method of generating Blue and Pink noise therefore, is by operating the White noise signal in the frequency domain directly, rather then in the time-domain. In

[6] The operations are in fact non-linear

other words, Blue noise can be generated by first applying a fast Fourier transform (FFT) on a White noise signal, then scaling the amplitude of each data point with the square root of its frequency, and finally applying an inverse fast Fourier transform. Pink noise can be generated by a similar procedure, with only difference being that the amplitude of each data point must be scaled with the inverse of the root of their frequency. Realizations of Pink and Blue noise, which have been generated by using the procedure as described, can be observed in Figure 2-4 and Figure 2-5. From the figures it becomes clear that the auto-correlation of Pink noise follows the shape of a negative root in both directions.

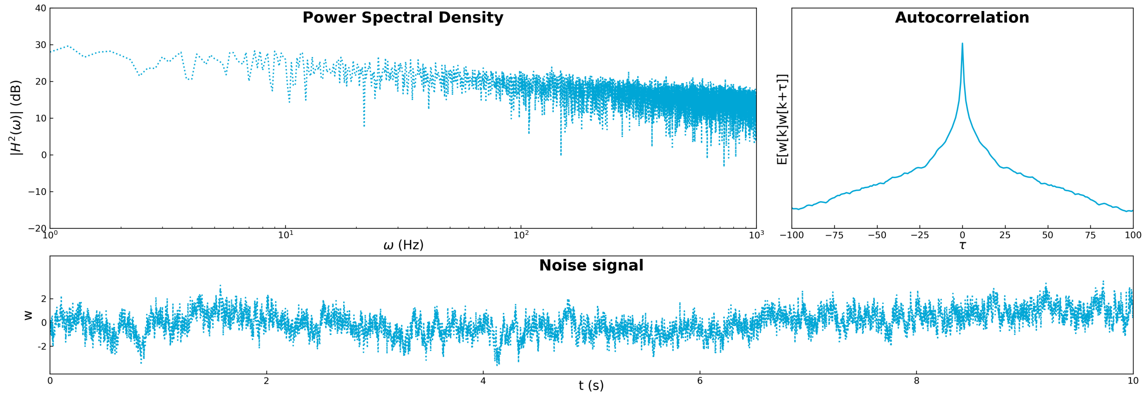

Figure 2-4: A Pink noise signal, its autocorrelation and PSD. The figure clearly depicts the -10 $^{\text{dB}}/_{\text{dec}}$ power spectrum of the Pink noise signal. The autocorrelation function approximates $-\sqrt{|\tau|}$, but cannot be accurately estimated due to the estimation method relying on the wide-sense stationarity[7] property, which Pink noise lacks $\Delta t = 10^{-3}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

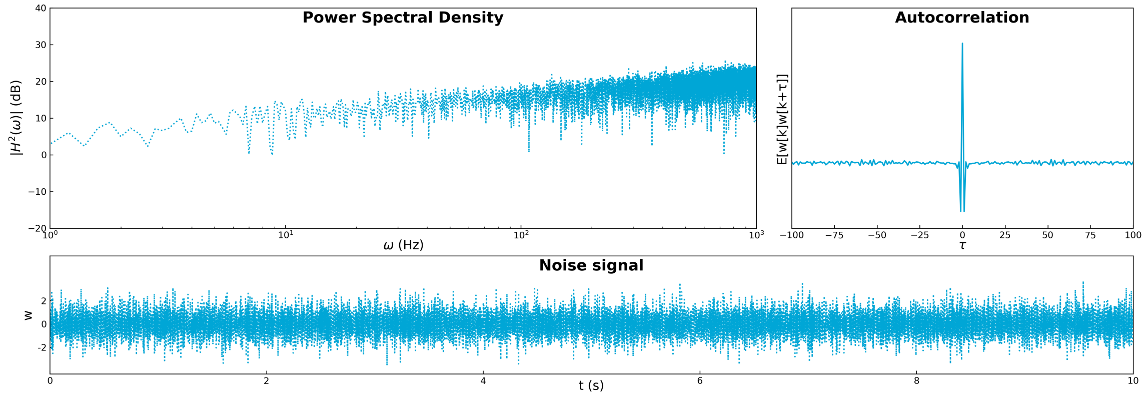

Figure 2-5: A Blue noise signal, its autocorrelation and PSD. The figure clearly depicts the 10 $^{\text{dB}}/_{\text{dec}}$ power spectrum of the Violet noise signal. The autocorrelation function approximates a non-linear finite-order central-difference filter. Intuitively, the non-linear difference function should be a square-root, but definitive proof for this statement lacks. $\Delta t = 10^{-3}$ s. T $= 10$ s. https://github.com/lznidarsic/sir/blob/master/demo_noise.py

2-2-2 Convolved noise

As described in the previous section, a portion of real-world noise can be explained by correlated noise that is simply dominated by high or low frequencies and in- or decreases linearly in the frequency domain. There are, however many other cases where the linear frequency response does not provide a sufficiently accurate explanation for the process noise. In such cases the noise can be modelled by White noise that has been convoluted by a linear or non-linear Kernel function. Mathematically, discrete-time convolution on a White noise signal brings:

$$\boldsymbol{y}[k] = \sum_{\tau=-\infty}^{\infty} \rho[k-\tau]\boldsymbol{x}[\tau]\Delta\mathrm{t} \tag{2-1}$$

with $\boldsymbol{x}$ a White noise signal, $\rho(k)$ the Kernel function and $\boldsymbol{y}$ the convoluted signal[8]. Even though the possibilities for the choice of these Kernel functions is endless, there are two that are most commonly discussed in the context of robotics and neuroscience: the Gaussian and Block kernel functions. I will discuss only these explicitly and will consider the other functions outside the scope of this thesis.

Gaussian-convolved noise Most commonly referred to in literature on the Free-Energy principle, which will be the main subject of thesis 4, is noise that has been convolved with a Gaussian kernel. The main reason for the broad adoption of such noise is the infinite differentiability of the Kernel function and consequently the noise signal itself. Mathematically, the discrete-time Gaussian Kernel function gives:

$$\rho[k] = \frac{1}{\Delta\mathrm{t}\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\frac{k^2}{\sigma^2}} \tag{2-2}$$

With σ the standard deviation of the Gaussian in number of sampling-time steps[9]. Note that the Δt term, which is a result of the discretization of the definition of the standard deviation, disappears when substituting Eq. (2-2) into Eq. (2-1). Figure 2-6 shows a realization of a Gaussian-convolved White noise signal. Note that the autocorrelation function is a Gaussian[10], and the frequency response, although less visible on a log-log scale, is once more a Gaussian.

[8] Note that in any practical application only a finite number of samples is available. Therefore the $-\infty$ and ∞ are generally replaced with 0 and N respectively. Not however, how this is not equivalent to making the convolution causal, $y[k_1]$ still depends on future samples of $x[k_1 > k]$

[9] In literature often referred to as Kernel width, although this is strictly incorrect as a Gaussian is infinitely wide and thus had no formal width.

[10] Though with a different width than the correlation filter

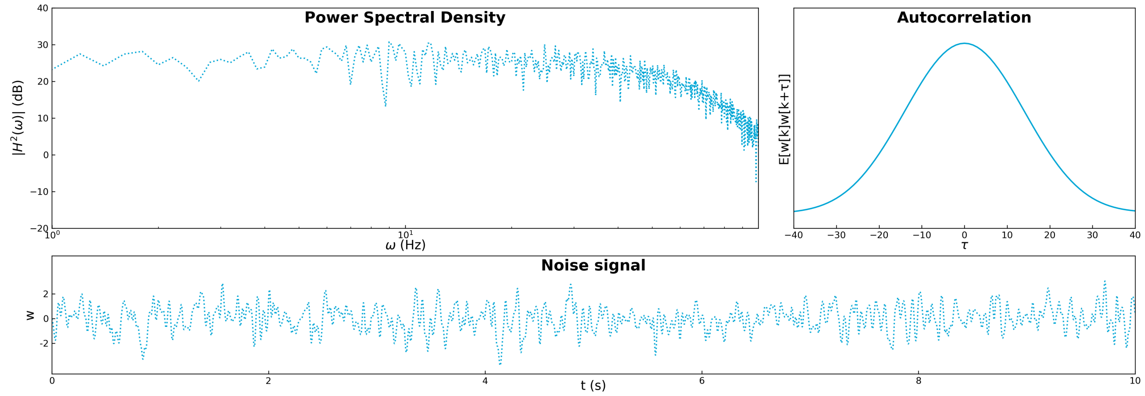

Figure 2-6: A Gaussian-convolved noise signal, its autocorrelation and PSD. The figure clearly depicts the Gaussian autocorrelation function of the Gaussian-convolved noise signal. Less clear due to the logarithmic axes, yet still present is the Gaussian-shaped power spectrum of the Gaussian-convolved noise signal. $\sigma = 10\Delta t$. $\Delta t = 10^{-3}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

Block-convolved noise Also often referred to as windowed averaging an common in literature on the Free-Energy principle as a signal smoother, is White noise that has been convolved with a Block-function kernel. Mathematically, the Block function can be constructed by addition of two step signals:

$$\rho[k] = \frac{1}{\sigma}\Big(u_s[k - \tfrac{\sigma}{2}] - u_s[k + \tfrac{\sigma}{2}]\Big) \tag{2-3}$$

where

$$u_s[k] = \begin{cases} 1 & \forall k \geq 0 \\ 0 & \forall k < 0 \end{cases} \tag{2-4}$$

with σ denoting the Kernel width.

Figure 2-7 shows a realization of a Block-convolved White noise signal. Note that the autocorrelation function is by approximation a triangle[11] function and the frequency response is a sinc[12] function.

[11] $R[k] = u_r[k - \frac{\sigma}{2}] - 2u_r[k] + u_r[k + \frac{\sigma}{2}]$ where $u_r[k] = k\Delta t \ \forall k \geq 0, \quad u_r[k] = 0 \ \forall k < 0$

[12] $\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$

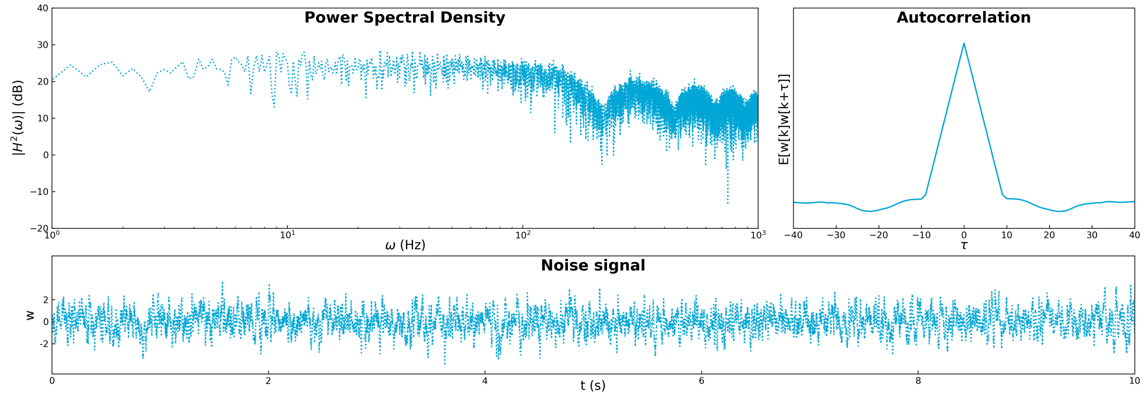

Figure 2-7: A Block-convolved noise signal, its autocorrelation and PSD. The figure depicts the sinc-function shaped power spectrum of the Block-convolved noise signal. The autocorrelation function approximates $-\sqrt{|\tau|}$ up until the width of the kernel, and 0 outside the kernel. $\sigma = 10\Delta\mathrm{t}$. $\Delta\mathrm{t} = 10^{-3}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

Coloured noise as convolved noise A final remark regarding convolved noise is that all coloured noise categories are in fact forms of convolved noise. Recall that Red noise as described in 2-2-1 is in fact integrated White noise. Integration itself can be represented as convolution with a step signal in negative time:

$$\rho[k] = 1 - u_s[k] = u_s[-k] \tag{2-5}$$

Similarly, recall Violet noise which is in fact differentiated White noise. Numerical differentiation can be can be represented as convolution with a non-causal 2^{nd} order difference filter:

$$\rho[k] = \frac{1}{2\Delta\mathrm{t}}(u_i[k+1] - 2u_i[k] + u_i[k-1]) \tag{2-6}$$

where $u_i[k]$ denotes the unit impulse function

$$u_i[k] = \begin{cases} 1 & \forall k = 0 \\ 0 & \forall k \neq 0 \end{cases} \tag{2-7}$$

Chapter 3

Expectation Maximization

In this chapter I explain why expectation maximization (EM) can be considered the state-of-the-art method for system identification in robotics. Furthermore, I describe the equations which define both the filter and the system identification method that determine EM, and the major limitation that the method suffers from. Ultimately, this chapter provides answers to research questions 1-3

3-1 State of the Art

The main goal of this chapter is to present the state of the art of system identification for robotics, such that it can later be compared with the novel approach proposed in the next chapter. However, before nosediving into the mathematical definitions and relations that define the expectation maximization algorithm, it is important to state why I am in fact considering this specific algorithm as the state of the art. There are in fact three major reasons why in the context of this thesis I consider EM the state of the art.

The first reason being its similarity to the novel method of dynamic expectation maximization (DEM), which is the main subject of this thesis. As will become clear in this and the next chapter, both methods are based on many of the same principles. Even stronger, in some of his literature [12], Friston describes DEM as an extension of EM.

The second reason being how widely EM is adopted. Initially, EM was described as a statistical principle by Dempster[1] et al. in [6]. However, soon after it was translated to a directly applicable algorithm for system identification by Shumway et al.in [7] it was recognized widely recognized and had been applied many times since.

The third reason being the fact that both EM and DEM can both be applied on- and off-line using the same equations. This latter point is very important to avoid conflict in parameter optima, and the reason that the dual application is important at all is to allow for robots to deal with changing parameters due to wear or temperature fluctuations, which are ubiquitous. A more detailed argument for these statements can be found in appendix A-3

[1] Which is the definition of EM that Friston refers to in his literature

3-2 Filtering

EM is a method for system identification which combines the state estimation of a Kalman filter (KF) with Likelihood Maximization to infer the parameters. In this section I will explain how the KF works, and why it is such a powerful method.

Optimal Filtering The main goal of any filtering method can either be to infer the hidden state, or to reject the noise on the measured output data. One fundamental principle of filtering is that these two goals are in fact mutually inclusive. For output noise rejection one needs an output estimate, which can only be constructed from the hidden state. Conversely, the hidden state estimate must be updated with the measured data, for which the filter needs an output estimate.

This becomes immediately clear when considering how such a filter is constructed. Consider a stable discrete-time LTI SS system:

$$\begin{aligned} \boldsymbol{x}[k+1] &= f(\boldsymbol{x}[k], \boldsymbol{u}[k]) = \mathbf{A}_d\boldsymbol{x}[k] + \mathbf{B}_d\boldsymbol{u}[k] + \boldsymbol{w}[k] \\ \boldsymbol{y}[k] &= h(\boldsymbol{x}[k], \boldsymbol{u}[k]) = \mathbf{C}_d\boldsymbol{x}[k] + \mathbf{D}_d\boldsymbol{u}[k] + \boldsymbol{z}[k] \end{aligned} \tag{3-1}$$

With $\boldsymbol{w} \frown \mathcal{N}(\mathbf{0}, \mathbf{Q})$ and $\boldsymbol{z} \frown \mathcal{N}(\mathbf{0}, \mathbf{R})$ both assumed White noise. Assuming that the parameters determining the system, i.e. the SS matrix entries, as well as those defining the noise, i.e. the covariance matrix entries are known, allows the construction of a simple filter of the form:

$$\begin{aligned} \hat{\boldsymbol{x}}[k] &= \mathbf{A}_d\hat{\boldsymbol{x}}[k-1] + \mathbf{B}_d\boldsymbol{u}[k-1] \\ \hat{\boldsymbol{y}}[k] &= \mathbf{C}_d\hat{\boldsymbol{x}}[k] + \mathbf{D}_d\boldsymbol{u}[k] \\ \hat{\boldsymbol{x}}'[k] &= \hat{\boldsymbol{x}}[k] + \mathbf{K}(\boldsymbol{y} - \hat{\boldsymbol{y}}[k]) \end{aligned} \tag{3-2}$$

with the SS matrices denoting a model that is in fact an exact copy of the system. $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}$ denote the predicted hidden state and output, whereas $\hat{\boldsymbol{y}}'$ and $\hat{\boldsymbol{y}}'$ denote the updated hidden state and output respectively.

From Eq. (3-2) it becomes clear that the filtering behaviour is completely determined by the updating gain matrix $\mathbf{K}$. Setting it very high will have the states follow the outputs exactly, but since the output was subject to noise, this will not yield the best estimate for the hidden state. Conversely, setting $\mathbf{K}$ low will yield a more smooth state estimate but might drive it away from the actual hidden state because of the accumulation of state noise.

This leads to the conclusion that there must be some $\mathbf{K}$ that is the optimal balance between rejecting the output noise whilst pulling the hidden state towards the measured data. This optimality can be defined by considering hidden state estimation as minimization of the distance between the unknown hidden state and the updated estimated hidden state, thus minimizing their mean squared error (MSE):

$$\text{MSE} = \frac{1}{N}\sum_{k=0}^{N} ||\boldsymbol{x}[k] - \hat{\boldsymbol{x}}[k]||^2 \tag{3-3}$$

From the realization that this statement is in fact equivalent to minimization of the trace of the covariance on the state error, it would makes sense to include information on the state- and output error covariances into the definition of this gain. Intuitively, increased covariance in the output error will decrease the gain and thus emphasize the predicted data. Conversely, increased covariance in the state error will increase the gain and thus emphasize the predicted data. Therefore, the best choice would be to choose the gain as **K** as:

$$\mathbf{K} = \mathbf{P}\mathbf{C}_d^\top \mathbf{S}^{-1} \tag{3-4}$$

with **P** and **S** the state- and output error covariances. Intuitively onw might think that **P** and **S** can be in fact be set to **Q** and **R** respectively, as the residual errors in the real system would be the noise, and would their covariance would be equal tothat of the noise. However, this will only be the case when the filter is somehow able to perfectly infer the hidden states for all time-steps, which can cannot be guaranteed[2] due to the non-invariant accumulation of state and output noise.

Kalman Filtering Therefore, an optimal filter which truly minimizes the criterion as defined in Eq. (3-3) must include some mechanism for estimating the state- and output error covariance which are dependent of but not equivalent to the noise covariances. A filter which includes such a mechanism, is the widely adopted Kalman filter [8]. It has been proven that for a LTI SS system, the optimal estimates of the state error covariance **P** and the output error covariance **S** are in fact the solution to the discrete-time algebraic Ricatti equation (DARE):

$$\mathbf{P} = \mathbf{A}\mathbf{P}\mathbf{A}^\top + (\mathbf{A}^\top\mathbf{P}\mathbf{C}^\top)(\mathbf{R} + \mathbf{C}\mathbf{P}\mathbf{C}^\top)^{-1}(\mathbf{C}\mathbf{P}\mathbf{A}) + \mathbf{Q} \tag{3-5}$$

which can be solved recursively by iterating until convergence:

$$\begin{aligned}
\hat{\mathbf{P}} &= \mathbf{A}_d\mathbf{P}'\mathbf{A}_d^\top + \mathbf{Q} \\
\hat{\mathbf{S}} &= \mathbf{C}_d\hat{\mathbf{P}}\mathbf{C}_d^\top + \mathbf{R} \\
\hat{\mathbf{K}} &= \mathbf{P}\mathbf{C}_d^\top\hat{\mathbf{S}}^{-1} \\
\mathbf{P}' &= (\mathbf{I} - \mathbf{K}\mathbf{C}_d)\hat{\mathbf{P}}
\end{aligned} \tag{3-6}$$

Note how solving the DARE provides, aside from the covariance estimates, the optimal Kalman gain. For linear time invariant systems, the DARE can be solved off-line prior to the filtering task. For non-linear, time-varying or on-line parameter estimation applications, Eq. (3-6) wil need to be run in parallel to the filter for continuous updating of the covariance estimates and gain, as it depends on $\mathbf{A}_d$ and $\mathbf{C}_d$.

3-3 Maximum Likelihood parameter estimation

In the previous section I discussed how for cases where a system is known and noise assumed to be White and of known distribution parameters, a hidden state can be optimally estimated as minimization of the euclidean distance between the actual and estimated hidden state data.

In this section I will complete the explanation of EM by showing how it combined the filtering properties of the KF with Likelihood Maximization to infer the system parameters for cases where they are not known.

[2] In fact, it is almost never the case

3-3-1 Likelihood

From a statistical mathematics point of view, uncorrelated data can be explained as a probability distribution around some mean, or expectation. Formally, the actual probability distribution can be time variant, non-linear and unknown. Truly inferring this probability distribution will therefore yield the necessity to run a very large number of simulations[3].

The probability distribution inference problem can be greatly reduced by assuming a fixed distribution function with known parameters[4] In fact, EM adopts the Laplace assumption [9], which states that this probability distribution can be assumed as a Gaussian. After this simplification, the problem of model parameter estimation reduces to finding those parameters that are the most likely explanation of both the observed and hidden data. This statement is is analogous to finding those parameters which maximally decrease the width of the probability distribution which describes the residual of the error between the estimated and actual data.

Furthermore, assuming that all residual errors are in fact independent of one another, which is generally referred to as the mean-field approximation [9], allows for definition of the joint probability distribution as the product of all the independent probability distributions. This finally brings for the joint density maximum-likelihood (ML):

$$\mathrm{ML} = \prod_{k=0}^{N} \frac{1}{||\mathbf{P}||\sqrt{2\pi}} e^{-\frac{1}{2}\boldsymbol{\varepsilon}_x^\top[k]\mathbf{P}^{-1}\boldsymbol{\varepsilon}_x[k]} \prod_{k=0}^{N} \frac{1}{||\mathbf{S}||\sqrt{2\pi}} e^{-\frac{1}{2}\boldsymbol{\varepsilon}_y^\top[k]\mathbf{S}^{-1}\boldsymbol{\varepsilon}_y[k]} \tag{3-7}$$

with

$$\begin{aligned} \boldsymbol{\varepsilon}_x[k] &:= \boldsymbol{x}[k] - \hat{\boldsymbol{x}}[k] \\ \boldsymbol{\varepsilon}_y[k] &:= \boldsymbol{y}[k] - \hat{\boldsymbol{y}}[k] \end{aligned} \tag{3-8}$$

Note how $\boldsymbol{\varepsilon}_x[k]$ depends on the hidden state $\boldsymbol{x}[k]$which is inherently unavailable. How EM overcomes this issue will become clear in the next section.

Furthermore, note how the expression of the ML is rather complex and subsequently that multiplication of different terms makes convexity of this function hard to infer. Therefore, using the fact that a log operation on function preserves convexity, allows for the definition of the log-likelihood (LL):

$$\mathrm{LL} = \sum_{k=0}^{N} -\frac{1}{2}\boldsymbol{\varepsilon}_x^\top[k]\mathbf{Q}^{-1}\boldsymbol{\varepsilon}_x[k] - \frac{1}{2}\boldsymbol{\varepsilon}_y^\top[k]\mathbf{R}^{-1}\boldsymbol{\varepsilon}_y[k] - \log(||\mathbf{Q}||) - \log(||\mathbf{R}||) - \log(2\pi) \tag{3-9}$$

Which is the cost function that EM minimizes. For a more comprehensive derivation of the LL from the ML, please refer to appendix A-5.

3-3-2 Maximization

In the previous section I stated that parameter estimation can be considered as maximization of the likelihood of the estimated data. This led to a cost function called the LL, which was simplified to the log-likelihood.

[3] Which is in fact the approach that Particle filtering methods adopt

[4] Estimating the parameters that determine this probability distribution, i.e. the hyper-parameters, is not part of the scope of this thesis.

In most cases, maximization of this cost is achieved by iterative gradient ascent on the full batch of data, which mathematically brings [10]:

$$\hat{\boldsymbol{\theta}}[i+1] = \hat{\boldsymbol{\theta}}[i] + \alpha \frac{\partial LL}{\partial \boldsymbol{\theta}} \tag{3-10}$$

with $\boldsymbol{\theta}$ the vector containing the system's parameters and α the step-size, a tunable hyperparameter.

On-line EM Alternatively, the same scheme can be applied in an on-line setting where the parameters are updated with each time step:

$$\hat{\boldsymbol{\theta}}[k+1] = \hat{\boldsymbol{\theta}}[k] + \alpha \frac{\partial \mathrm{LL}[k]}{\partial \boldsymbol{\theta}} \tag{3-11}$$

with $\mathrm{LL}[k]$ the LL evaluated at a each incoming data sample. Note that incrementally updating the model matrices yields a time-varying solution to the Ricatti equation, which implies that the error-covariance estimation and gain updating Eq. (3-6) will have to be run in parallel to the on-line parameter estimations scheme.

> **Note:** In some implementations of the expectation maximization algorithm, a 2nd order gradient ascent scheme[a] is adopted, which essentially replaces the scalar step-size with the inverse of the Hessian[b] of the LL towards θ. Even though these algorithms generally converge faster, the region of attraction in terms of initial parameters is much smaller for such an algorithm than for 1st gradient descent, as the former needs the Hessian to be positive definite, and the interval on which this can be guaranteed is generally tighter than that on which the function decreases, which is the only assumption that 1st order gradient descent needs. In my implementation, I therefore chose for 1st order gradient descent.
>
> [a] In literature commonly referred to as Gauss-Newton
>
> [b] 2nd order gradient

Algebraical approximation of the gradient The gradient, can be algebraically described by:

$$\begin{aligned}
\frac{\partial J}{\partial \boldsymbol{\theta}} &= \frac{1}{N}\sum_{k=0}^{N} \Big(\frac{\partial(\boldsymbol{x}[k]-\hat{\boldsymbol{x}}[k])}{\partial \boldsymbol{\theta}^\top}\Big)^\top \mathbf{P}^{-1}(\boldsymbol{x}[k]-\hat{\boldsymbol{x}}'[k]) + \Big(\frac{\partial(\boldsymbol{y}[k]-\hat{\boldsymbol{y}}[k])}{\partial \boldsymbol{\theta}^\top}\Big)^\top \mathbf{S}^{-1}(\boldsymbol{y}[k]-\hat{\boldsymbol{y}}'[k]) \\
&= \frac{1}{N}\sum_{k=0}^{N} -\Big(\frac{\partial \hat{\boldsymbol{x}}[k]}{\partial \boldsymbol{\theta}^\top}\Big)^\top \mathbf{P}^{-1}(\boldsymbol{x}[k]-\hat{\boldsymbol{x}}[k]) - \Big(\frac{\partial \hat{\boldsymbol{y}}[k]}{\partial \boldsymbol{\theta}^\top}\Big)^\top \mathbf{S}^{-1}(\boldsymbol{y}[k]-\hat{\boldsymbol{y}}'[k])
\end{aligned}$$

Note however, how this gradient depends on both the estimated hidden state data as well as the actual hidden state data, which is unknown.

This is were the KF enters the algorithm. We have two estimates for the hidden state data. One which depends only on the model and the parameters, and one which has been updated with the measured data. Therefore, an approximation of this likelood function can be achieved by adopting the KF, which immediately provides an estimate for the $\mathbf{P}$ and $\mathbf{S}$ matrices as well.

Therefore, the gradient can be approximated by

$$\frac{\partial J}{\partial \boldsymbol{\theta}} \approx \frac{1}{N}\sum_{k=0}^{N} -\Big(\frac{\partial \hat{\boldsymbol{x}}[k]}{\partial \boldsymbol{\theta}^\top}\Big)^\top \mathbf{P}^{-1}(\hat{\boldsymbol{x}}'[k]-\hat{\boldsymbol{x}}[k]) - \Big(\frac{\partial \hat{\boldsymbol{y}}[k]}{\partial \boldsymbol{\theta}^\top}\Big)^\top \mathbf{S}^{-1}(\boldsymbol{y}[k]-\hat{\boldsymbol{y}}'[k])$$

The gradients of the hidden state towards the parameters are found by :

$$\begin{aligned}\frac{\partial \hat{\boldsymbol{x}}[k]}{\partial \boldsymbol{\theta}} &= \mathbf{F}[k] + \mathbf{A}_d\mathbf{F}[k-1] + \mathbf{A}_d^2\mathbf{F}[k-2]... \\ \frac{\partial \hat{\boldsymbol{y}}[k]}{\partial \boldsymbol{\theta}} &= \mathbf{H}[k] + \mathbf{C}_d(\mathbf{F}[k] + \mathbf{A}_d\mathbf{F}[k-1] + \mathbf{A}_d^2\mathbf{F}[k-2]...)\end{aligned} \tag{3-12}$$

with $\mathbf{F}[k]$ the gradient of the state equations towards the parameters $\frac{\partial f}{\partial \boldsymbol{\theta}}$ and $\mathbf{H}[k]$the gradient of the measurement equations towards the parameters $\frac{\partial h}{\partial \boldsymbol{\theta}}$.

Note how, formally, the gradient will have to be traced back in time infinitely, which is not practically feasible. In any implementation, therefore, some finite gradient-embedding order is chosen as an extra tunable hyper-parameter. An extreme case are models where the parameters are are directly on the observable state, in which case the order can be set to one. In any other case the tuning of this parameter will have to be a balance between accuracy and computational efficiency

> **Remark:** Tracing the gradient of $\hat{\boldsymbol{x}}$ towards $\boldsymbol{\theta}$ back multiple steps for each sample becomes infeasible quickly as the order of the system or the number of samples increases. Therefore, in some implementations, the gradient is approximated numerically.
> Note, however that doing so will increase the problem of $\hat{\boldsymbol{x}}'$ not truly representing $\boldsymbol{x}$, and thus shift the actual optimum of the cost function further away from the theoretical optimum[a]. In other words, numerical gradient approximations will yield a more efficient algorithm at the cost of parameter estimation performance.
>
> [a] The real parameters

3-4 Answers to research questions

To conclude this chapter I briefly circle back to the research questions as formulated in the introduction to answer them based on the content of this chapter.

Why is EM the state-of-the art method? EM is directly applicable in both the off- and online stages of robot learning. Furthermore, EM is widely adopted, provides optimal performance given certain circumstances and is very similar to DEM in terms of cost and minimization.

What are the theoretical principles which define EM? EM combines the hidden state estimations of a KF with a maximum-likelihood estimation of the system parameters. The minimization of the likelihood is achieved by gradient descent using either an algebraical or numerical gradient approximation. Estimated parameters can either be updated iteratively on a batch of collected data or incrementally on new data, i.e. EM can be applied both in an off-line and on-line setting.

What are the limitations of EM? Only under strict assumptions can EM perform optimally. One of the assumptions is the uncorrelated of noise assumption. Violation of this assumption seriously deteriorates the filtering accuracy of the identified model and filter. As discussed in chapter 2, there is but one type of noise for which this property holds, namely White noise. There are many real world systems for which the noise processes can not be accurately modelled by White noise. Consequently there are many systems which can not be accurately identified using EM. Naturally, there are more limitations, but these are not relevant for the comparison with DEM as it will be made in this thesis[5].

[5] Further limitations, such as lack of input estimation must be considered at a later stage in this research

Chapter 4

Dynamic Expectation Maximization

In this chapter I introduce the Generalized Filtering (GF) [11] and dynamic expectation maximization (DEM) [12] methods as described from the Free-Energy principle (FEP) [13] by K.J. Friston. In order to translate the theory into an applicable method for system identification, I propose a discrete-time interpretation of GF and DEM. Ultimately, this chapter provides answers to research questions 4 and 5.

4-1 The Free-Energy Principle

In the first part of this chapter, where I explain the theoretical principles underlying dynamic expectation maximization, I will adopt the continuous-time definitions as stated by Friston in his literature. Furthermore I will adopt the state-space formulations as discussed in [14] and [15] (both yet to be published). Later in the chapter, I will propose a discrete-time definition of the algorithm, which will be more in line with the rest of the theory described in this thesis.

4-1-1 Generalized systems

Central to the Free-Energy principle, which is the broader set of theories which includes dynamic expectation maximization is the concept of generalized systems. It rests upon the fact that, depending on the nature of the signals, there might be information embedded in the time derivatives of the input and noise signals .

One way of accessing the information embedded in these higher order signal derivatives, is by considering not only the 1^{st} order hidden state derivative of the system, but rather considering all its higher order derivatives up to a certain order p, the embedding order. The dynamics of these higher order derivative signals are driven by copies of the system itself and the derivatives of the input and noise signals. This can be easily shown by considering a continuous-time LTI SS system, perturbed with state noise $\boldsymbol{w} \frown \mathcal{N}(\mathbf{0}, \mathbf{Q})$ and output noise

$\boldsymbol{z} \frown \mathcal{N}(\mathbf{0}, \mathbf{R})$, both of which do not necessarily conform to the white noise assumption. Taking the time derivatives on both sides up to order p brings:

$$\begin{aligned}
\dot{\boldsymbol{x}}(t) &= \mathbf{A}_c\boldsymbol{x}(t) + \mathbf{B}_c\boldsymbol{u}(t) + \boldsymbol{w}(t)\\
\ddot{\boldsymbol{x}}(t) &= \mathbf{A}_c\dot{\boldsymbol{x}}(t) + \mathbf{B}_c\dot{\boldsymbol{u}}(t) + \dot{\boldsymbol{w}}(t)\\
&\vdots\\
\boldsymbol{x}^{(p+1)}(t) &= \mathbf{A}_c\boldsymbol{x}^{(p)}(t) + \mathbf{B}_c\boldsymbol{u}^{(p)}(t) + \boldsymbol{w}^{(p)}(t)
\end{aligned}$$

and

$$\begin{aligned}
\boldsymbol{y}(t) &= \mathbf{C}_c\boldsymbol{x}(t) + \mathbf{D}_c\boldsymbol{u}(t) + \boldsymbol{z}(t)\\
\dot{\boldsymbol{y}}(t) &= \mathbf{C}_c\dot{\boldsymbol{x}}(t) + \mathbf{D}_c\dot{\boldsymbol{u}}(t) + \dot{\boldsymbol{z}}(t)\\
&\vdots\\
\boldsymbol{y}^{(p)}(t) &= \mathbf{C}_c\boldsymbol{x}^{(p)}(t) + \mathbf{D}_c\boldsymbol{u}^{(p)}(t) + \boldsymbol{z}^{(p)}(t)
\end{aligned} \tag{4-1}$$

Considering the union of the variables and their higher order derivatives, i.e. by generalizing the state, input, noise and output and consequently the dynamics underlying them, brings:

$$\begin{aligned}
\begin{bmatrix} \dot{\boldsymbol{x}}(t)\\ \ddot{\boldsymbol{x}}(t)\\ \vdots\\ \boldsymbol{x}^{(p+1)}(t) \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_c & & & \\ & \mathbf{A}_c & & \\ & & \ddots & \\ & & & \mathbf{A}_c \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t)\\ \dot{\boldsymbol{x}}(t)\\ \vdots\\ \boldsymbol{x}^{(p)}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_c & & & \\ & \mathbf{B}_c & & \\ & & \ddots & \\ & & & \mathbf{B}_c \end{bmatrix} \begin{bmatrix} \boldsymbol{u}(t)\\ \dot{\boldsymbol{u}}(t)\\ \vdots\\ \boldsymbol{u}^{(p)}(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{w}(t)\\ \dot{\boldsymbol{w}}(t)\\ \vdots\\ \boldsymbol{w}^{(p)}(t) \end{bmatrix}\\
\begin{bmatrix} \boldsymbol{y}(t)\\ \dot{\boldsymbol{y}}(t)\\ \vdots\\ \boldsymbol{y}^{(p)}(t) \end{bmatrix} &= \begin{bmatrix} \mathbf{C}_c & & & \\ & \mathbf{C}_c & & \\ & & \ddots & \\ & & & \mathbf{C}_c \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t)\\ \dot{\boldsymbol{x}}(t)\\ \vdots\\ \boldsymbol{x}^{(p)}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{D}_c & & & \\ & \mathbf{D}_c & & \\ & & \ddots & \\ & & & \mathbf{D}_c \end{bmatrix} \begin{bmatrix} \boldsymbol{u}(t)\\ \dot{\boldsymbol{u}}(t)\\ \vdots\\ \boldsymbol{u}^{(p)}(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{z}(t)\\ \dot{\boldsymbol{z}}(t)\\ \vdots\\ \boldsymbol{z}^{(p)}(t) \end{bmatrix}
\end{aligned} \tag{4-2}$$

which can be written in a more concise format as

$$\begin{aligned}
\dot{\tilde{\boldsymbol{x}}}(t) &= \tilde{\mathbf{A}}_c\tilde{\boldsymbol{x}}(t) + \tilde{\mathbf{B}}_c\tilde{\boldsymbol{u}}(t) + \tilde{\boldsymbol{w}}(t)\\
\tilde{\boldsymbol{y}}(t) &= \tilde{\mathbf{C}}_c\tilde{\boldsymbol{x}}(t) + \tilde{\mathbf{D}}_c\tilde{\boldsymbol{u}}(t) + \tilde{\boldsymbol{z}}(t)
\end{aligned} \tag{4-3}$$

Note how all generalized SS matrices are block-diagonal, and thus there is no coupling between the different embedding layers, and thus evolution of the signals is independent.

4-1-2 Generalized filtering

Directly following the concept of generalized systems comes that of generalized filtering, i.e. inferring the evolution of the generalized hidden state in the presence of unknown generalized noise from the evolution of the measured in- and output signals.

One way of inferring this generalized hidden state is by employing a model of the generalized system, which is for now assumed to be an exact copy of the actual generalized system as described in Eq. (4-3):

$$\begin{aligned}
\dot{\hat{\tilde{\boldsymbol{x}}}}(t) &= \hat{\tilde{\mathbf{A}}}_c\hat{\tilde{\boldsymbol{x}}}(t) + \hat{\tilde{\mathbf{B}}}_c\tilde{\boldsymbol{u}}(t)\\
\hat{\tilde{\boldsymbol{y}}}(t) &= \hat{\tilde{\mathbf{C}}}_c\hat{\tilde{\boldsymbol{x}}}(t) + \hat{\tilde{\mathbf{D}}}_c\tilde{\boldsymbol{u}}(t)
\end{aligned} \tag{4-4}$$

Considering the diagonal structure of the generalised system matrices as defined in Eq. (4-2), it becomes immediately clear that each layer in the estimated generalized hidden state evolves completely independent from all other others. This implies that only when the initial conditions of the generalized hidden state and the complete evolution of the generalized input and noise signals are known and consistent, can we expect consistence in the evolution of the estimated generalized hidden state. A generalized variable is consistent if the signal in each consecutive layer of a generalized variable is exactly the derivative of the signal in the preceding layer

Since the noise signals are not known, there will not be consistence between the embedding layers. However, assuming that the generalized input is known, the inconsistency of the estimated hidden state can is driven only by the generalized noise. It is therefore that it is exactly this inconsistency that we aim to minimize in order to infer the generalized hidden state.

Considering the fact that in each subsequent layer of the generalized hidden state is the derivative of the current layer, allows for the definition of the layer-shift-up operator

$$\boldsymbol{\mathcal{D}} := \mathbf{U}_p \otimes \mathbf{I}_n \tag{4-5}$$

with $\mathbf{U}_p$ a $p \times p$ upper-shift matrix[1] and $\mathbf{I}_n$ an $n \times n$[2] Identity matrix, such that

$$\dot{\hat{\tilde{\boldsymbol{x}}}}'(t) = \boldsymbol{\mathcal{D}}\hat{\tilde{\boldsymbol{x}}}(t) \tag{4-6}$$

This method for generalized hidden state derivative approximation leads to the definition of the internal consistency error, i.e. the how much any layer does NOT represent the derivative of the preceding one:

$$\begin{aligned} \boldsymbol{\varepsilon}_{\tilde{\boldsymbol{x}}}(t) : &= \dot{\hat{\tilde{\boldsymbol{x}}}}'(t) - \dot{\hat{\tilde{\boldsymbol{x}}}}(t) \\ &= \boldsymbol{\mathcal{D}}\hat{\tilde{\boldsymbol{x}}}(t) - \hat{\tilde{\mathbf{A}}}_c\hat{\tilde{\boldsymbol{x}}}(t) - \hat{\tilde{\mathbf{B}}}_c\tilde{\boldsymbol{u}}(t) \end{aligned} \tag{4-7}$$

Furthermore, assuming the generalized system output is fully available for measurement[3], leads to the definition of the generalized prediction error.

$$\begin{aligned} \boldsymbol{\varepsilon}_{\tilde{\boldsymbol{y}}}(t) : &= \tilde{\boldsymbol{y}}(t) - \hat{\tilde{\boldsymbol{y}}}(t) \\ &= \tilde{\boldsymbol{y}}(t) - \hat{\tilde{\mathbf{C}}}_c\hat{\tilde{\boldsymbol{x}}}(t) - \hat{\tilde{\mathbf{D}}}_c\tilde{\boldsymbol{u}}(t) \end{aligned} \tag{4-8}$$

Free-Energy Then, the Free-Energy principle considers filtering as minimization of the width of the joint probability distribution that describes both the error in internal consistency and the error in output prediction. Furthermore, assuming one more that the errors independent[4] and are normally distributed[5] leads to the formulation of the variational Free-Energy, which is in fact a log-likelihood of the form[6]:

$$\mathcal{F}(t) := U(t) + W_\theta(t) \tag{4-9}$$

[1] Sparse matrix with 1's on the super-diagonal

[2] Recall n is the dimensionality of the state-space, i.e. the length of $\boldsymbol{x}$

[3] Of course in many applications it is not. The next chapter will concern methods for estimating generalized signals

[4] Mean-field approximation

[5] Laplace assumption

[6] Please refer to appendix A-5 once more for the derivation of the log-likelihood.

with the internal energy

$$U(t) := -\frac{1}{2}\boldsymbol{\varepsilon}_{\tilde{x}}^{\top}(t)\tilde{\mathbf{\Pi}}_{w}\boldsymbol{\varepsilon}_{\tilde{x}}(t) - \frac{1}{2}\boldsymbol{\varepsilon}_{\tilde{y}}^{\top}(t)\tilde{\mathbf{\Pi}}_{z}\boldsymbol{\varepsilon}_{\tilde{y}}(t) + \log(\tilde{\mathbf{\Pi}}_{w}) + \log(\tilde{\mathbf{\Pi}}_{z}) \tag{4-10}$$

denoting the likelihood of the generalized hidden state estimation. $W_{\theta}(t)$ is the mean field term which concerns the uncertainty of the parameters. For now the parameters can be assumed as known, as the mean-field term can be assumed as zero. I will discuss this term more in-depth in the next section.

Noise precision matrices The matrices $\tilde{\mathbf{\Pi}}_{w}$ and $\tilde{\mathbf{\Pi}}_{z}$ are constructed such that they contain the inverses of the covariances $\mathbf{Q}$ and $\mathbf{R}$ of the White noise that preceded the correlation filter, and information on how this covariance is expected to scale in the deeper layers of embedding:

$$\begin{aligned}\tilde{\mathbf{\Pi}}_{w} &:= \mathbf{S}^{-1} \otimes \mathbf{Q}^{-1} \\ \tilde{\mathbf{\Pi}}_{z} &:= \mathbf{S}^{-1} \otimes \mathbf{R}^{-1}\end{aligned} \tag{4-11}$$

with $\mathbf{S}$ the temporal correlation matrix which includes te information from the (assumed) autocorrelation function [16]:

$$\mathbf{S} := \begin{bmatrix} E[\rho(t)\rho(t)] & E[\rho(t)\dot{\rho}(t)] & \cdots & E[\rho(t)\rho^{(p)}(t)] \\ E[\dot{\rho}(t)\rho(t)] & E[\dot{\rho}(t)\dot{\rho}(t)] & \cdots & E[\dot{\rho}(t)\rho^{(p)}(t)] \\ \vdots & & \ddots & \\ E[\rho^{(p)}(t)\rho(t)] & E[\rho^{(p)}(t)\dot{\rho}(t)] & \cdots & E[\rho^{(p)}(t)\rho^{(p)}(t)] \end{bmatrix} \tag{4-12}$$

where $\rho(t)$ as stated in chapter 2 can formally be any kind of auto-correlation, but in Friston's literature it is most commonly assumed as a Gaussian-convolved noise, see ch. 2 or [16].

Minimization As stated, generalized filtering is then minimization of the FE using gradient descent, such that the generalized hidden state remains consistent, i.e.:

$$\dot{\hat{\tilde{\boldsymbol{x}}}}(t) = \boldsymbol{\mathcal{D}}\hat{\tilde{\boldsymbol{x}}}(t) - \alpha_{\tilde{x}}\frac{\partial \mathcal{F}(t)}{\partial \tilde{\boldsymbol{x}}} \tag{4-13}$$

with $\alpha_{\tilde{x}}$ the gradient descent step size, a tunable hyper-parameter.

4-1-3 Parameter estimation

In the previous section I explained that filtering under the Free-Energy principle is in fact minimization of a function called the variational Free-Energy. In this section, I will build on that statement, by showing how a similar principle can be used for estimating unknown parameters underlying a system, i.e. how parameter estimation is also a form of free energy minimization[7].

In the previous chapter I explained how for expectation maximization, the parameters are estimated as a maximization of the likelihood of the estimated data. For parameter estimation under the Free-Energy principle using dynamic expectation maximization, this statement remains unchanged[8].

[7] In fact, according to Friston all forms of inference, including estimating unknown inputs and selecting optimal control action, are a form of Free-Energy minimization

[8] Aside from the fact that the FE is in fact negative Likelihood which is thus minimized rather than maximized

However, the actual variational Free-Energy is more complex than what was stated in Eq. (4-9). In fact, the former definition is a simplification of the full definition of the Free-Energy for cases where the parameters are known and certain, as some terms that define the Free-Energy in fact drop when the parameters are certain.

Free-Energy This happens because, subsequent to the estimated generalized hidden states and output, the parameters are modelled as a probability distribution as well, which allows for estimation of not only the parameters, but also their certainty[9]. Consequently, the Free-Energy brings:

$$\begin{aligned}\bar{\mathcal{F}} :&= \bar{U} + \bar{W}_{\tilde{x}} \\ &= \int_0^t U(t)\mathrm{d}t + \int_0^t W_{\tilde{x}}(t)\mathrm{d}t \end{aligned} \tag{4-14}$$

with $U(t)$ as defined in Eq. (4-10) and

$$W_{\tilde{x}}(t) = -\mathrm{tr}\Big(\mathbf{\Pi}_{\tilde{x}}^{-1}(t)\Big(\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}^\top(t)}{\partial \tilde{\boldsymbol{x}}}\tilde{\mathbf{\Pi}}_w\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}(t)}{\partial \tilde{\boldsymbol{x}}} + \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}^\top(t)}{\partial \tilde{\boldsymbol{x}}}\tilde{\mathbf{\Pi}}_z\frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}(t)}{\partial \tilde{\boldsymbol{x}}}\Big)\Big) \tag{4-15}$$

correcting the parameter estimates for the uncertainty in the generalized hidden state estimates, as according to the mean-field theory [9].

Recall that I mentioned in the previous section how the variational Free-Energy also contains a mean-field term which is a consequence of the mean-field approximation[10] [9] and corrects the generalized hidden state estimates for the uncertainty in the parameter estimates. Since now the parameters are unknown, it cannot be assumed that this term drops to zero. Therefore, it is defined as:

$$W_{\theta}(t) = -\mathrm{tr}\Big(\mathbf{\Pi}_{\theta}^{-1}(t)\Big(\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}^\top(t)}{\partial \boldsymbol{\theta}}\tilde{\mathbf{\Pi}}_w\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}(t)}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}^\top(t)}{\partial \boldsymbol{\theta}}\tilde{\mathbf{\Pi}}_z\frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}(t)}{\partial \boldsymbol{\theta}}\Big)\Big) \tag{4-16}$$

State and parameters precision Matrices Lastly, note how both of these terms depend on the precision of $\tilde{\boldsymbol{x}}$ and $\boldsymbol{\theta}$ of the generalized hidden state and parameters respectively. As proposed by Friston, these precisions can be approximated as the second order gradient of the internal energy towards the corresponding variables. For the precision of $\tilde{\boldsymbol{x}}$ this brings:

$$\begin{aligned}\mathbf{\Pi}_{\tilde{x}}(t) &= \frac{\partial^2 \bar{U}}{\partial \tilde{\boldsymbol{x}}^2} \\ &= \int_0^t \frac{\partial^2 U(t)}{\partial \tilde{\boldsymbol{x}}^2}\mathrm{d}t \\ &= \int_0^t \Big(\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}^\top(t)}{\partial \tilde{\boldsymbol{x}}}\tilde{\mathbf{\Pi}}_w\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}(t)}{\partial \tilde{\boldsymbol{x}}} + \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}^\top(t)}{\partial \tilde{\boldsymbol{x}}}\tilde{\mathbf{\Pi}}_z\frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}(t)}{\partial \tilde{\boldsymbol{x}}}\Big)\mathrm{d}t \end{aligned} \tag{4-17}$$

$$\tag{4-18}$$

[9] In fact, Friston goes further to state that even the hyper-parameters, i.e. the parameters that determine the noise and its correlation, can be modelled by a probability distribution rather than a fixed number. Estimation of the noise parameters will not be considered within the scope of this thesis.

[10] Which raises the question why EM, which is also built upon the mean-field approximation, does not contain these terms

and for the precision of $\boldsymbol{\theta}$ this brings:

$$\begin{aligned}\mathbf{\Pi}_\theta(t) &= \frac{\partial^2 \bar{U}}{\partial \boldsymbol{\theta}^2} \\ &= \int_0^t \frac{\partial^2 U(t)}{\partial \boldsymbol{\theta}^2} \mathrm{dt} \\ &= \int_0^t \Big(\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}^\top(t)}{\partial \boldsymbol{\theta}} \tilde{\mathbf{\Pi}}_w \frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}(t)}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}^\top(t)}{\partial \boldsymbol{\theta}} \tilde{\mathbf{\Pi}}_z \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}(t)}{\partial \boldsymbol{\theta}} \Big) \mathrm{dt} \end{aligned} \tag{4-19}$$

Note how, as time evolves, the values in the precision matrices are expected to grow due to the integration. Consequently, note that the mean field terms, which include the inverse of these precision matrices, are thus expected to die out as time evolves.

Parameter update Now that we have fully defined the Free-Energy as it is minimized for the parameter estimation step. The actual minimization is once more performed via 1st order gradient descent, which is formally defined as

$$\dot{\boldsymbol{\theta}}(t) = -\alpha_\theta \frac{\partial \bar{\mathcal{F}}}{\partial \boldsymbol{\theta}} \tag{4-20}$$

4-1-4 Priors

For completeness it must be mentioned that there is in fact one[11] last term in the Free-Energy which is referred to as the prior expectation term. It is defined as a log-likelihood of the estimated parameters as opposed to their prior expectation:

$$\Xi := \boldsymbol{\varepsilon}_\theta^\top \mathbf{P}_\theta \boldsymbol{\varepsilon}_\theta \tag{4-21}$$

where

$$\varepsilon_\theta := \boldsymbol{\eta}_\theta - \hat{\boldsymbol{\theta}} \tag{4-22}$$

with $\boldsymbol{\eta}_\theta$ containing the prior parameter values, and $\mathbf{P}_\theta$ the certainty of the prior parameter values. Setting the values in P to high values indicate a large certainty in the prior estimates for the parameters. Note that by inverting this logic, the term can formally be removed by just assuming complete uncertainty in the prior expectation, which is analogous to setting the values in $\mathbf{P}_\theta$ to zero.

The argumentation against the use of priors as described in Eq. (4-21) can be formulated by comparing it with the standard way that prior information is included, namely as initial estimates. Note that placing the prior information in the cost function inherently shifts the optimum of the cost function, whilst just choosing the prior as an initial estimate does not. It can be argued that when including the prior into the cost function, the new optimum will lie somewhere between the prior and the initial optimum. This will increase accuracy of the final estimate if only if the prior was in fact a better estimate than the minimum of the cost function, but in such a case one might have wondered why system identification was necessary in the first place.

[11] Actually two, considering the hyper-parameter estimation contains a prior as well

4-2 Discretization

Any application of the theory explained in the previous section, be it for robot learning or system identification in general, will inevitably deal with discrete samples. This makes the continuous-time interpretation of the world, which is at the foundation of the free-energy principle as presented in the previous section, though very suitable for analysis of continuous systems, rather incompatible for real applications.

Thus, it makes sense to consider the world as a discrete-time generative process of data, i.a. to consider discrete-time systems. Therefore, in this section I will propose a discrete-time interpretation of all principles discussed in the previous section, which together form a complete discrete-time interpretation of DEM.

4-2-1 Embedded signals and their discrete-time interpretation

Preceding the formulation of a discrete-time version of dynamic expectation maximization is the interpretation of discrete-time generalized systems, and more fundamentally discrete-time embedded signals. Considering the latter, there is one major design choice which drives the definition of discrete-time DEM: what information is embedded in the deeper layers of the generalized signals. In this section I will propose three options regarding this choice and briefly motivate their pro's and cons.

Firstly the question on what information is hidden in the deeper layer of embedding: do they contain the 1^{st} -, 2^{nd} -, 3^{rd} -, etc. order derivative signals, or their one-, two-, three, etc. step ahead predictions, or even one-, two-, three, etc. step back predictions. I will refer to these three different approaches as embedded derivatives (DEM-ED), embedded predictions (DEM-EP) and embedded history (DEM-EH). Mathematically these generalized signals would bring, for an exemplary signal $\phi[k]$:

$$\begin{aligned}
\tilde{\phi}_p[k] &:= \begin{bmatrix}\phi[k] & \phi[k+1] & \phi[k+2] & \cdots & \phi[k+p]\end{bmatrix}^\top \\
\tilde{\phi}_d[k] &:= \begin{bmatrix}\phi[k] & \dot{\phi}[k] & \ddot{\phi}[k] & \cdots & \phi^{(p)}[k]\end{bmatrix}^\top \\
\tilde{\phi}_h[k] &:= \begin{bmatrix}\phi[k] & \phi[k-1] & \phi[k-2] & \cdots & \phi[k-p]\end{bmatrix}^\top
\end{aligned} \tag{4-23}$$

with subscripts p, d and h referring to DEM with embedded predictions (DEM-EP), derivatives (DEM-ED) and history (DEM-EH) respectively.

As will become clear in the remainder of this chapter, each of the three proposed approaches introduces inherent changes to how the equations as introduced in the previous section are formulated. It is hard to predict how these changes will affect the behaviour and performance of DEM, which led to the decision to elaborate and implement all three. Aside from their behaviour however, there are practical advantages and disadvantages to each of the approaches.

DEM-EP relies on future data, which in an off-line system identification setting can be easily obtained by simply shifting the signals 1, 2, 3, etc. time-steps back. However, when System identification has finished the model that has been identified will likely be employed in an on-line filtering setting. Then, the future data will not be readily available and will thus have to be estimated using some external method. As will become clear as will become clear in section 4-2-3, DEM-EP yields a change in precision matrices $\tilde{\mathbf{\Pi}}_w$ and $\tilde{\mathbf{\Pi}}_z$.

DEM-ED relies on derivative data, which even in an off-line system identification setting must be estimated using some external method as it is not readily available. In an off-line setting this method does not have to be causal; it can for example be a central-difference estimator. However, when System identification has finished and the model that has been identified will be employed in an on-line filtering setting, the method will have to be causal due to the unavailability of future data. It remains unclear whether the final filtering performance will be better when the non-causal method is used both for system identification and filtering, or when the identification is done non-causally and the filtering is done causally. As will become clear as will become clear in section 4-2-3, DEM-ED yields a change in the shift matrix $\boldsymbol{\mathcal{D}}$.

DEM-EH relies on past data, which means that both in system identification and filtering applications there is no need for an external method for generalized signal estimation. However, as will become clear in section 4-2-3, DEM-ED yields a change in both the shift matrix $\boldsymbol{\mathcal{D}}$ and the precision matrices $\tilde{\mathbf{\Pi}}_w$ and $\tilde{\mathbf{\Pi}}_z$, which means that this approach is most dissimilar to the original formulation of DEM and thus behaviour is hardest to predict.

4-2-2 Generalized systems and their discrete-time interpretation

Any interpretation of discrete-time embedded signals as defined in the previous section allows for the discrete-time interpretation of discrete-time generalized systems. Consider a discrete-time LTI SS process:

$$\begin{aligned} \boldsymbol{x}[k+1] &= \mathbf{A}_d\boldsymbol{x}[k] + \mathbf{B}_d\boldsymbol{u}[k] + \boldsymbol{w}[k] \\ \boldsymbol{y}[k] &= \mathbf{C}_d\boldsymbol{x}[k] + \mathbf{D}_d\boldsymbol{u}[k] + \boldsymbol{z}[k] \end{aligned} \tag{4-24}$$

Then, combining th definition of a discrete-time system with my definition the embedded signals as stated in the previous section leads to my definition of a discrete-time generalized system:

$$\begin{aligned} \tilde{\boldsymbol{x}}[k+1] &= \tilde{\mathbf{A}}_d\tilde{\boldsymbol{x}}[k] + \tilde{\mathbf{B}}_d\tilde{\boldsymbol{u}}[k] + \tilde{\boldsymbol{w}}[k] \\ \tilde{\boldsymbol{y}}[k] &= \tilde{\mathbf{C}}_d\tilde{\boldsymbol{x}}[k] + \tilde{\mathbf{D}}_d\tilde{\boldsymbol{u}}[k] + \tilde{\boldsymbol{z}}[k] \end{aligned} \tag{4-25}$$

with generalized system matrices:

$$\begin{aligned} \tilde{\mathbf{A}}_d &:= \mathbf{I}_p \otimes \mathbf{A}_d \\ \tilde{\mathbf{B}}_d &:= \mathbf{I}_p \otimes \mathbf{B}_d \\ \tilde{\mathbf{C}}_d &:= \mathbf{I}_p \otimes \mathbf{C}_d \\ \tilde{\mathbf{D}}_d &:= \mathbf{I}_p \otimes \mathbf{D}_d \end{aligned} \tag{4-26}$$

and p denoting the embedding order[12]. The definition of the signals $\tilde{\boldsymbol{u}}$, $\tilde{\boldsymbol{x}}$, $\tilde{\boldsymbol{y}}$, $\tilde{\boldsymbol{w}}$ and $\tilde{\boldsymbol{v}}$ is according to Eq. (4-23) and depends on the embedded information setting.

[12] Note the slight abuse of notation, where p is used both as an indication of a generalized signal with embedded predictions, and as the embedding order (parameter determining the length of the generalized signals)

4-2-3 Generalized Filtering

Building upon the proposed definitions of the three types of discrete-time embedded signals and generalized systems, I will propose three methods for estimating the generalized hidden state based on the method of generalized filtering as discussed in section 4-1-2.

Generalized model At the foundation of the proposed generalized filter is still the generative model, which is for now assumed again as an exact copy of the system:

$$\begin{aligned}\hat{\tilde{\boldsymbol{x}}}[k] &= \hat{\tilde{\mathbf{A}}}_d\hat{\tilde{\boldsymbol{x}}}[k] + \hat{\tilde{\mathbf{B}}}_d\tilde{\boldsymbol{u}}[k]\\ \hat{\tilde{\boldsymbol{y}}}[k] &= \hat{\tilde{\mathbf{C}}}_d\hat{\tilde{\boldsymbol{x}}}[k] + \hat{\tilde{\mathbf{D}}}_d\tilde{\boldsymbol{u}}[k]\end{aligned} \tag{4-27}$$

Shift-matrices For generalized signals with embedded predictions the upper shift matrix as proposed in Eq. (4-5) yields a forward shift, i.e. one-step ahead prediction, which is in accordance with the model:

$$\boldsymbol{\mathcal{D}}_p := \boldsymbol{\mathcal{D}} \tag{4-28}$$

with $\boldsymbol{\mathcal{D}}$ as in Eq. (4-5).

The first real differences between the my discrete-time (DT) generalized filtering approach and the one discussed by Friston in [11, 12] emerge in the shift matrices for the remaining two proposed approaches. Note that simply shifting the entries of $\hat{\tilde{\boldsymbol{x}}}$ up for generalized signals with embedded derivatives or embedded history will yield derivative and 1-step-back predictions respectively, whereas the model still predicts a one-step-ahead prediction. These two realizations of the generalized signals therefore call for a different shift matrix.

For the generalized signals with embedded history, it can be reasoned that if an upper-shift matrix yields a backwards prediction, then a lower-shift matrix will yield a forwards prediction. Therefore, the shift matrix $\boldsymbol{\mathcal{D}}$ must be redefined as the 1-step-down operator:

$$\boldsymbol{\mathcal{D}}_h := \mathbf{L}_p \otimes \mathbf{I}_n \tag{4-29}$$

with $\mathbf{L}_p$ a $p \times p$ lower-shift matrix[13] and $\mathbf{I}_n$ an $n \times n$ Identity matrix, such that

$$\hat{\tilde{\boldsymbol{x}}}'[k+1] = \boldsymbol{\mathcal{D}}_h\hat{\tilde{\boldsymbol{x}}}[k] \tag{4-30}$$

Proof of this statement can be found in appendix A-6.

For the generalized signals with embedded derivatives, it can be reasoned that if a upper-shift matrix yields a derivative prediction, then the translation from derivative predictions to one-step-ahead predictions yields a discretization step. Therefore, using the exact discretization approach, the shift matrix $\boldsymbol{\mathcal{D}}$ must be redefined as the discretization operator:

$$\boldsymbol{\mathcal{D}}_d := e^{\boldsymbol{\mathcal{D}}_p \mathrm{dt}} \tag{4-31}$$

with $\boldsymbol{\mathcal{D}}_p$ the upper-shift matrix used for the generalized signals with embedded predictions.

[13] Sparse matrix with 1's on the sub-diagonal

Generalized errors The redefinition of the shift matrices as proposed in the previous paragraph allow me to define the discrete-time version of the internal consistency- and output prediction error for the generalized system with embedded predictions:

$$\begin{aligned}\boldsymbol{\varepsilon}_{\tilde{\boldsymbol{x}}}[k] &: = \boldsymbol{\mathcal{D}}_p \hat{\tilde{\boldsymbol{x}}}_p[k] - \hat{\tilde{\mathbf{A}}}_d \hat{\tilde{\boldsymbol{x}}}_p[k] - \hat{\tilde{\mathbf{B}}}_d \tilde{\boldsymbol{u}}_p[k] \\ \boldsymbol{\varepsilon}_{\tilde{\boldsymbol{y}}}[k] &: = \tilde{\boldsymbol{y}}_p[k] - \hat{\tilde{\mathbf{C}}}_c \hat{\tilde{\boldsymbol{x}}}_p[k] - \hat{\tilde{\mathbf{D}}}_c \tilde{\boldsymbol{u}}_p[k]\end{aligned} \tag{4-32}$$

Similar statements can be written for the generalized errors with embedded derivatives and those with embedded history, but since

Free-energy With the definition of the internal consistency errors, the Free-Energy used for filtering can be redefined. However, note that the Free-Energy (FE) as defined in Eq. (4-9) is instantaneous. Therefore, the discrete-time statement is equal to the continuous-time definition as in
u

$$\mathcal{F}[k] := U[k] + W_\theta[k] \tag{4-33}$$

with

$$U[k] := -\frac{1}{2}\boldsymbol{\varepsilon}_{\tilde{x}}^\top[k]\tilde{\mathbf{\Pi}}_w \boldsymbol{\varepsilon}_{\tilde{x}}[k] - \frac{1}{2}\boldsymbol{\varepsilon}_{\tilde{y}}^\top[k]\tilde{\mathbf{\Pi}}_z \boldsymbol{\varepsilon}_{\tilde{y}}[k] + \log(\tilde{\mathbf{\Pi}}_w) + \log(\tilde{\mathbf{\Pi}}_z) \tag{4-34}$$

and the mean field term $W_\theta[k]$, for now, again zero as I assumed no parameter uncertainty.

Noise precision matrices The definitions of the precision matrices, including the temporal correlation, remains unchanged for the embedded derivatives case.

However, for the other two cases, i.e. embedded prediction and embedded history, the temporal correlation matrix underlying these matrices does change, as it must now contain information on how subsequent samples are correlated, rather then how samples and their derivatives are correlated. For both DEM-EP and DEM-EH this brings:[14]:

$$\mathbf{S} := \begin{bmatrix} E[\rho[0]\rho[0]] & E[\rho[0]\rho[1]] & \cdots & E[\rho[0]\rho[p]] \\ E[\rho[1]\rho[0]] & E[\rho[1]\rho[1]] & \cdots & E[\rho[1]\rho[p]] \\ \vdots & \vdots & \ddots & \vdots \\ E[\rho[p]\rho[0]] & E[\rho[p]\rho[1]] & \cdots & E[\rho[p]\rho[p]] \end{bmatrix} \tag{4-35}$$

with for the Gaussian-convolved noise case $\rho[k] = e^{-\frac{1}{2}\frac{k^2}{\sigma^2}}$

Filtering Finally, the filtering law remains unchanged, and thus brings gradient descent on the Free-Energy such that internal consistency is enforced:

$$\hat{\tilde{\boldsymbol{x}}}[k+1] = \boldsymbol{\mathcal{D}}\hat{\tilde{\boldsymbol{x}}}[k] - \alpha_{\tilde{x}}\frac{\partial \mathcal{F}[k]}{\partial \tilde{\boldsymbol{x}}} \tag{4-36}$$

[14] Strictly speaking, for the embedded history case, the positive numbers in the corellation matrix should be negative. However, since, the Gaussian corellation filter is symmetric, this does not yield any change in the matrix entries.

4-2-4 Parameter estimation

Following the same principles as derived in the previous section, I will now redefine the equations for parameter estimation using DEM.

Free-Energy Consequent to the discrete-time definition of the free-energy function in the previous section is the one that is used for parameter estimation. It contains an integral, which in discrete time can be approximated by:

$$\begin{aligned}\bar{\mathcal{F}} :=& \bar{U} + \bar{W}_{\tilde{x}} \\ =& \sum_{k=0}^{N} U[k]\Delta\mathrm{t} + \sum_{k=0}^{N} \bar{W}_{\tilde{x}}[k]\Delta\mathrm{t}\end{aligned} \tag{4-37}$$

with $U[k]$ as defined in Eq. (4-10)

Mean-Field terms Since the mean field terms are instantaneous, they can be directly translated to the discrete-time definition by substituting (t) with $[k]$. Furthermore, the assumption of a LTI SS system and model allows for the further simplification of the terms:

$$\begin{aligned}W_{\tilde{x}}[k] :=& -\operatorname{tr}\Big(\mathbf{\Pi}_{\theta}^{-1}[k]\Big(\frac{\partial\boldsymbol{\varepsilon}_{\tilde{x}}^{\top}[k]}{\partial\tilde{\boldsymbol{x}}}\tilde{\mathbf{\Pi}}_{w}\frac{\partial\boldsymbol{\varepsilon}_{\tilde{x}}[k]}{\partial\tilde{\boldsymbol{x}}} + \frac{\partial\boldsymbol{\varepsilon}_{\tilde{y}}^{\top}[k]}{\partial\tilde{\boldsymbol{x}}}\tilde{\mathbf{\Pi}}_{z}\frac{\partial\boldsymbol{\varepsilon}_{\tilde{y}}[k]}{\partial\tilde{\boldsymbol{x}}}\Big)\Big) \\ =& -\operatorname{tr}\Big(\mathbf{\Pi}_{\theta}^{-1}[k]\Big((\boldsymbol{\mathcal{D}}_{i} - \hat{\hat{\mathbf{A}}}_{d})^{\top}\tilde{\mathbf{\Pi}}_{w}(\boldsymbol{\mathcal{D}}_{i} - \hat{\hat{\mathbf{A}}}_{d}) + (-\hat{\hat{\mathbf{C}}}_{d})^{\top}\tilde{\mathbf{\Pi}}_{z}(-\hat{\hat{\mathbf{C}}}_{d})\Big)\Big) \\ W_{\theta}[k] :=& -\operatorname{tr}\Big(\mathbf{\Pi}_{\tilde{x}}^{-1}[k]\Big(\frac{\partial\boldsymbol{\varepsilon}_{\tilde{x}}^{\top}([k]}{\partial\boldsymbol{\theta}}\tilde{\mathbf{\Pi}}_{w}\frac{\partial\boldsymbol{\varepsilon}_{\tilde{x}}[k]}{\partial\boldsymbol{\theta}} + \frac{\partial\boldsymbol{\varepsilon}_{\tilde{y}}^{\top}([k]}{\partial\boldsymbol{\theta}}\tilde{\mathbf{\Pi}}_{z}\frac{\partial\boldsymbol{\varepsilon}_{\tilde{y}}[k]}{\partial\boldsymbol{\theta}}\Big)\Big) \\ =& -\operatorname{tr}\Big(\mathbf{\Pi}_{\theta}^{-1}[k]\Big((-\hat{\hat{\mathbf{F}}}[k])^{\top}\tilde{\mathbf{\Pi}}_{w}(-\hat{\hat{\mathbf{F}}}[k]) + (-\hat{\hat{\mathbf{H}}}[k])^{\top}\tilde{\mathbf{\Pi}}_{z}(-\hat{\hat{\mathbf{H}}}[k])\Big)\Big)\end{aligned} \tag{4-38}$$

with $\tilde{\mathbf{F}}[k]$ and $\tilde{\mathbf{H}}[k]$ the generalized counterparts of the transition- and measurement-function to parameter gradients as defined in Eq. (3-12).

State- and parameter precision Matrices Furthermore, the precision matrices that are needed for the mean-field terms, and which involve an integral, can be approximated by a sum over the samples. Furthermore, once more assuming LTI SS system and model allows for the further simplification of precisions:

$$\begin{aligned}
\mathbf{\Pi}_{\tilde{x}}[k] :=& \frac{\partial \bar{U}}{\partial \tilde{\boldsymbol{x}}} \\
=& \sum_{i=0}^{k} \Big(\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}^{\top}[i]}{\partial \tilde{\boldsymbol{x}}} \tilde{\mathbf{\Pi}}_w \frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}[i]}{\partial \tilde{\boldsymbol{x}}} + \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}^{\top}[i]}{\partial \tilde{\boldsymbol{x}}} \tilde{\mathbf{\Pi}}_z \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}[i]}{\partial \tilde{\boldsymbol{x}}} \Big) \Delta \mathrm{t} \\
=& \sum_{i=0}^{k} \Big((\boldsymbol{\mathcal{D}}_i - \hat{\hat{\mathbf{A}}}_d)^{\top} \tilde{\mathbf{\Pi}}_w (\boldsymbol{\mathcal{D}}_i - \hat{\hat{\mathbf{A}}}_d) + (-\hat{\hat{\mathbf{C}}}_d)^{\top} \tilde{\mathbf{\Pi}}_z (-\hat{\hat{\mathbf{C}}}_d) \Big) \Delta \mathrm{t} \\
=& k \Delta \mathrm{t} \Big((\boldsymbol{\mathcal{D}}_i - \hat{\hat{\mathbf{A}}}_d)^{\top} \tilde{\mathbf{\Pi}}_w (\boldsymbol{\mathcal{D}}_i - \hat{\hat{\mathbf{A}}}_d) + (-\hat{\hat{\mathbf{C}}}_d)^{\top} \tilde{\mathbf{\Pi}}_z (-\hat{\hat{\mathbf{C}}}_d) \Big) \\
\mathbf{\Pi}_{\theta}[k] :=& \frac{\partial \bar{U}}{\partial \boldsymbol{\theta}} \\
=& \sum_{i=0}^{k} \Big(\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}^{\top}([i]}{\partial \boldsymbol{\theta}} \tilde{\mathbf{\Pi}}_w \frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}[i]}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}^{\top}([i]}{\partial \boldsymbol{\theta}} \tilde{\mathbf{\Pi}}_z \frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}[i]}{\partial \boldsymbol{\theta}} \Big) \Delta \mathrm{t} \\
=& \sum_{i=0}^{k} \Big((-\hat{\hat{\mathbf{F}}}[i])^{\top} \tilde{\mathbf{\Pi}}_w (-\hat{\hat{\mathbf{F}}}[i]) + (-\hat{\hat{\mathbf{H}}}[i])^{\top} \tilde{\mathbf{\Pi}}_z (-\hat{\hat{\mathbf{H}}}[i]) \Big) \Big) \Delta \mathrm{t}
\end{aligned} \tag{4-39}$$

Note how the precision terms grow as time evolves, due to summation of positive definite matrices. Consequently, their inverse, which drives the mean-field terms decreases. As a result, the mean-field terms die out as time evolves.

The parameter update Now that we have fully defined the Free-Energy as it is minimized for the parameter estimation step. The actual minimization is once more performed via 1^{st} order gradient descent, which is formally defined as

$$\boldsymbol{\theta}[i+1] = \boldsymbol{\theta}[i] - \alpha_{\theta} \frac{\partial \bar{\mathcal{F}}}{\partial \boldsymbol{\theta}} \tag{4-40}$$

The on-line parameter update Similar to expectation maximization (EM), DEM also includes the possibility of updating the parameters on-line, which brings

$$\boldsymbol{\theta}[k+1] = \boldsymbol{\theta}[k] - \alpha_{\theta} \frac{\partial \bar{\mathcal{F}}[k]}{\partial \boldsymbol{\theta}} \tag{4-41}$$

with $\bar{\mathcal{F}}[k]$ the FE as stated in Eq. (4-37), but evaluated at a single data sample.

The parameter gradient Note how the parameter updating scheme relies on the calculation of the gradient from the Free-Energy towards the parameters. This gradient can be algebraically calculated by

$$\begin{aligned}
\frac{\partial \bar{\mathcal{F}}}{\partial \boldsymbol{\theta}} &= \frac{\partial \bar{U}}{\partial \boldsymbol{\theta}} + \frac{\partial \bar{W}_{\tilde{x}}}{\partial \boldsymbol{\theta}} \\
&= \frac{1}{N} \sum_{k=0}^{N} \frac{\partial U[k]}{\partial \boldsymbol{\theta}} \Delta \mathrm{t} + \frac{1}{N} \sum_{k=0}^{N} \frac{\partial W_{\tilde{x}}[k]}{\partial \boldsymbol{\theta}} \Delta \mathrm{t}
\end{aligned} \tag{4-42}$$

with

$$\begin{aligned}\frac{\partial U[k]}{\partial \boldsymbol{\theta}} &= -\frac{1}{2}\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}^{\top}[k]}{\partial \boldsymbol{\theta}}\tilde{\mathbf{\Pi}}_w \boldsymbol{\varepsilon}_{\tilde{x}}[k] \\ -\frac{1}{2}\frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}^{\top}[k]}{\partial \boldsymbol{\theta}}\tilde{\mathbf{\Pi}}_z \boldsymbol{\varepsilon}_{\tilde{y}}[k] &= -\frac{1}{2}\frac{\partial \tilde{\boldsymbol{x}}^{\top}[k]}{\partial \boldsymbol{\theta}}\tilde{\mathbf{\Pi}}_w \boldsymbol{\varepsilon}_{\tilde{x}}[k] - \frac{1}{2}\frac{\partial \tilde{\boldsymbol{y}}^{\top}[k]}{\partial \boldsymbol{\theta}}\tilde{\mathbf{\Pi}}_z \boldsymbol{\varepsilon}_{\tilde{y}}[k]\end{aligned} \tag{4-43}$$

and the mean-field term gradient to be evaluated per-hidden state-variable:

$$\begin{aligned}\frac{\partial W_{\tilde{x}}[k]}{\partial \theta_i} &= -\mathrm{tr}\Big(\mathbf{\Pi}_{\theta}^{-1}[k]\Big(\frac{\partial^2 \boldsymbol{\varepsilon}_{\tilde{x}}^{\top}[k]}{\partial \tilde{\boldsymbol{x}}\partial \theta_i}\tilde{\mathbf{\Pi}}_w\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}[k]}{\partial \tilde{\boldsymbol{x}}} + \frac{\partial^2 \boldsymbol{\varepsilon}_{\tilde{y}}^{\top}[k]}{\partial \tilde{\boldsymbol{x}}\partial \theta_i}\tilde{\mathbf{\Pi}}_z\frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}[k]}{\partial \tilde{\boldsymbol{x}}}\Big)\Big) \\ &= -\mathrm{tr}\Big(\mathbf{\Pi}_{\theta}^{-1}[k]\Big(\frac{\partial^2 \hat{\tilde{\boldsymbol{x}}}^{\top}[k]}{\partial \tilde{\boldsymbol{x}}\partial \theta_i}\tilde{\mathbf{\Pi}}_w\frac{\partial \boldsymbol{\varepsilon}_{\tilde{x}}[k]}{\partial \tilde{\boldsymbol{x}}} + \frac{\partial^2 \hat{\tilde{\boldsymbol{y}}}^{\top}[k]}{\partial \tilde{\boldsymbol{x}}\partial \theta_i}\tilde{\mathbf{\Pi}}_z\frac{\partial \boldsymbol{\varepsilon}_{\tilde{y}}[k]}{\partial \tilde{\boldsymbol{x}}}\Big)\Big)\end{aligned} \tag{4-44}$$

Though theoretically relevant, it remains unclear whether tracing the gradient towards $\boldsymbol{\theta}$ further than one step is necessary within DEM. In his SPM-toolbox [17], Friston does not, which simply leaves: $\frac{\partial \hat{x}[k]}{\partial \boldsymbol{\theta}} = \mathbf{F}[k]$ and $\frac{\partial \hat{y}[k]}{\partial \boldsymbol{\theta}} = \mathbf{H}[k]$. Even stronger, in many of the demos that the SPM-toolbox includes, Friston simply approximates the gradients numerically.

4-3 Answers to research questions

To conclude this chapter I briefly circle back to the research questions as formulated in the introduction to answer them based on the content of this chapter.

What is the theoretical principle which defines DEM? DEM is part of the Free-Energy principle. Consequently it assumes systems are perturbed with correlated noise, which introduces the need for generalization of systems, i.e. considering the information in higher order derivatives of the signals. Furthermore, the FEP considers all types of inference, i.e. both that of parameters and that of hidden states, as minimization of a function called the Free-Energy, which is essentially the joint negative log-likelihood (LL) of the estimated generalized hidden state's internal consistency in the generalized output prediction accuracy.

How can DEM be translated into a method for filtering and system identification? The major translation step from DEM as proposed by Friston to something directly applicable as a system identification method is a discrete-time interpretation.

Chapter 5

Generalized signal estimation

In this chapter I introduce the feasibility limitation that is inherent to two of the three formulations of dynamic expectation maximization (DEM) as proposed in the previous chapter. Furthermore, I propose two different solutions for overcoming the feasibility problem and validate their performance with numerical simulations. Ultimately, this chapter provides answers to research questions 6, 7 and 8

5-1 The feasibility limitation

Central to filtering and system identification under the Free-Energy principle (FEP) is the assumption that the derivatives of a systems in- and outputs can be obtained. However, in most real world applications these signals are inherently unobtainable, posing serious limitations to the feasibility of any real world application of DEM.

Therefore, I propose two methods for estimating generalized signals with embedded derivatives. Subsequently I extend these methods to a discrete-time formulation, such that the methods can be used directly in conjunction with the formulations of DEM as proposed in the previous chapter.

5-2 Numerical signal differentiation

Following a local linearity assumption and zero order hold, a first-order backward-Euler approximation can be performed to numerically estimate the derivative of any signal ϕ based on the current and previous data-point:

$$\dot{\phi}[k] \approx \frac{1}{\Delta \mathrm{t}}(\phi[k] - \phi[k-1]) \tag{5-1}$$

Similarly, this procedure can be repeated using three data-points to estimate the second order derivative:

$$\begin{aligned}\ddot{\phi}[k] &\approx \frac{1}{\Delta \mathrm{t}}(\dot{\phi}[k] - \dot{\phi}[k-1]) \\ &\approx \frac{1}{\Delta \mathrm{t}}(\frac{1}{\Delta \mathrm{t}}(\phi[k] - \phi[k-1]) - \frac{1}{\Delta \mathrm{t}}(\phi[k-1] - \phi[k-2])) \\ &= \frac{1}{\Delta \mathrm{t}^2}(\phi[k] - 2\phi[k-1] + \phi[k-2]) \end{aligned} \tag{5-2}$$

Repeating this procedure p times allows for estimation of all the entries of the generalized signal $\tilde{\boldsymbol{\phi}}_d[k]$ of order p:

$$\tilde{\boldsymbol{\phi}}[k] := \begin{bmatrix} \phi[k] & \dot{\phi}[k] & \ddot{\phi}[k] & \cdots & \phi^{(p)}[k] \end{bmatrix}^\top \tag{5-3}$$

Vectorization of the system of equations that follows the repetitive application of Eq. (5-1) and Eq. (5-2) brings:

$$\tilde{\boldsymbol{\phi}}[k] \approx \boldsymbol{\Delta}^{-1} \boldsymbol{\mathcal{P}}_p \mathbf{I}_{\pm} \tilde{\boldsymbol{\phi}}_h[k] \tag{5-4}$$

with $\boldsymbol{\mathcal{P}}_p$ a Pascal matrix of order p, $\boldsymbol{\Delta} := \mathrm{diag}(\Delta\mathrm{t},\ \Delta\mathrm{t}^2,\ ...,\ \Delta\mathrm{t}^p)$ and $\mathbf{I}_{\pm}$ an identity matrix with 1 on uneven rows and -1 on even rows. Furthermore

$$\tilde{\boldsymbol{\phi}}_h[k] := \begin{bmatrix} \phi[k] & \phi[k-1] & \cdots & \phi[k-p] \end{bmatrix}^\top \tag{5-5}$$

is the generalized signal with embedded history, which can be directly inferred from past data. For a more detailed derivation of these equations please refer to appendix A-1.

5-3 Dynamical filter signal differentiation

In this section I propose a second, fundamentally different, approach for generalized signal estimation, which is adopted from the field of adaptive control and thoroughly discussed in [18].

It states that for a continuous-time signal $\phi(t)$ the derivative can be inferred by considering the signal in the frequency domain. Recall from chapter 2 that the p^{th}-order derivative in the time-domain yields a slope-transform in the frequency domain, i.e.:

$$\mathcal{L}(\phi^{(p)}(t)) = s^p \mathcal{L}(\phi(t)) \tag{5-6}$$

with $\mathcal{L}(\cdot)$ the Laplace transform and s the Laplace variable. Note that in terms of a bode plot of the operation, differentiation is equivalent to placing a zero at $s = 0$.

Furthermore, linear systems theory states that any operator $\phi(t)$ whose Laplace transform has a numerator that is of higher order than its denominator[1], is inherently unstable. Therefore, the adaptive control approach includes a $(p+1)^{\text{th}}$-order filter into the the derivative estimation scheme, which flattens the slope of the frequency domain response, thus rendering a stable differentiator operation. In the Laplace domain, this filter can be considered as:

$$\Lambda(s) = \frac{\lambda^{p+1}}{(s+\lambda)^{p+1}}$$

[1] i.e. has more zeros than poles

with λ the cut-off frequency of the filter and p the order of the derivative. The full differentiation operation, including the stable filter, then brings:

$$\begin{aligned} H(s) &= \Lambda(s)s^p \\ &= \frac{\lambda^{p+1}s^p}{(s+\lambda)^{p+1}} \end{aligned} \tag{5-7}$$

A translation step from this transfer function into a dynamical filter that can be directly used for signal differentiation, yields a transfer function to LTI SS mapping of $H(s)$.

Even more so, by choosing a specific realization, the controllable canonical form, it turns out that the state of this dynamical filter already contains the filtered input signal in its top layer, the 1^{st} order derivative in the second layer, the 2^{nd} order derivative in the third layer and so on, up to a scaling factor. Mathematically, this realization brings a continuous-time LTI SS system:

$$\begin{aligned} \dot{\boldsymbol{x}}_\phi(t) &= \mathbf{A}_\phi \boldsymbol{x}_\phi(t) + \mathbf{B}_\phi \phi(t) \\ \tilde{\boldsymbol{\phi}}_d(t) &= \mathbf{C}_\phi \boldsymbol{x}_\phi(t) + \mathbf{D}_\phi \phi(t) \end{aligned}$$

where $\mathbf{A}_\phi$ and $\mathbf{B}_\phi$ follow from realization theory, $\mathbf{C}_\phi = \lambda^{p+1}\mathbf{I}_{p\times p+1}$ and $\mathbf{D}_\phi = \mathbf{0}$. For a more detailed description of the controllable canonical state-space realization and how it can be inferred from the transfer function, please refer to appendix A-2.

The stable dynamical filter can then be discretized using theory from the field of digital control [19], which states that exact discretization[2] yields:

$$\begin{aligned} \mathbf{A}_d &= e^{\mathbf{A}_\phi \Delta \mathrm{t}} \\ \mathbf{B}_d &= \mathbf{A}_\phi^{-1}(\mathbf{A}_d - \mathbf{I})\mathbf{B}_\phi \end{aligned} \tag{5-8}$$

which can then be used in the discrete-time filter:

$$\begin{aligned} \boldsymbol{x}_\phi[k+1] &= \mathbf{A}_d \boldsymbol{x}_\phi[k] + \mathbf{B}_d \phi[k] \\ \tilde{\boldsymbol{\phi}}_d[k] &= \mathbf{C}_\phi \boldsymbol{x}_\phi[k] \end{aligned} \tag{5-9}$$

> **Note:** Choosing the cut-off frequency λ very large will yield a negligible phase shift and thus very accurate estimation. However, doing so will also yield large spikes when the initial embedded signal is unknown and will increase the sensitivity to discretization error.

5-4 Signal predictions estimation

The approaches in the previous two sections provide methods for estimating generalized signals which involve embedded derivatives. However, another interpretation of DEM that was proposed in the previous chapter involved embedded predictions rather than embedded derivatives. Therefore, the approaches as introduces in the previous sections must be extended such that the estimated generalized signals do not contain embedded derivatives, but embedded predictions.

[2] Still assuming zero-order hold and local linearity, so not really exact, just more accurate than Euler

This can be easily realized by applying the inverse procedure of numerical derivative estimation, i.e. by assuming local linearity and numerically integrating the embedded signal using forward Euler, such that for a signal $\phi[k]$, given $\phi[k]$ and $\dot{\phi}[k]$:

$$\phi[k+1] \approx \phi[k] + \Delta\mathrm{t}\dot{\phi}[k] \tag{5-10}$$

Subsequently,

$$\dot{\phi}[k+1] \approx \dot{\phi}[k] + \Delta\mathrm{t}\ddot{\phi}[k] \tag{5-11}$$

and thus

$$\begin{aligned}\phi[k+2] &\approx \phi[k+1] + \Delta\mathrm{t}\dot{\phi}[k+1] \\ &\approx \phi[k] + \Delta\mathrm{t}\dot{\phi}[k] + \Delta\mathrm{t}(\dot{\phi}[k] + \Delta\mathrm{t}\ddot{\phi}[k]) \\ &\approx \phi[k] + 2\Delta\mathrm{t}\dot{\phi}[k] + \Delta\mathrm{t}^2\ddot{\phi}[k]\end{aligned} \tag{5-12}$$

Repeating this procedure and vectorizing the equations yields:

$$\tilde{\boldsymbol{\phi}}_p[k] \approx \boldsymbol{\mathcal{P}}_p \boldsymbol{\Delta} \tilde{\boldsymbol{\phi}}_d[k] \tag{5-13}$$

with $\tilde{\boldsymbol{\phi}}_d[k]$ as defined in Eq. (5-3). For a more detailed derivation of these equations please refer to appendix A-4.

Numerical discrete-time generalized signal estimation Combining the approach as proposed in this section with the first approach I proposed for embedded derivative estimation, yields a method which can be used directly to infer a generalized signal with embedded predictions from a historical sequence of data. In other words, combining equations Eq. (5-4) and Eq. (5-13) yields:

$$\begin{aligned}\tilde{\boldsymbol{\phi}}_p[k] &\approx \boldsymbol{\mathcal{P}}_p \boldsymbol{\Delta} \boldsymbol{\Delta}^{-1} \boldsymbol{\mathcal{P}}_p \mathbf{I}_{\pm} \boldsymbol{\phi}_h[k] \\ &\approx \boldsymbol{\Delta} \boldsymbol{\mathcal{P}}_p \boldsymbol{\mathcal{P}}_p \mathbf{I}_{\pm} \boldsymbol{\phi}_h[k]\end{aligned} \tag{5-14}$$

Stable discrete-time generalized signal estimation Combining the approach as proposed in this section with the second approach I proposed for embedded derivative estimation, yields a method which can be used to infer a generalized signal with embedded predictions from a dynamical filter which stabily estimates the embedded derivatives of a generalized signal. Shifting from embedded derivatives to embedded predictions does not induce changes for the dynamics of the filter itself, only for the observation model. In other words, combining equations Eq. (5-9) and Eq. (5-13) yields:

$$\begin{aligned}\tilde{\boldsymbol{\phi}}_p[k] &\approx \boldsymbol{\mathcal{P}}_p \boldsymbol{\Delta} \Big(\mathbf{C}_{\tilde{\phi}} \tilde{\boldsymbol{x}}_{\phi}[k] + \mathbf{D}_{\tilde{\phi}} \phi[k] \Big) \\ &= \boldsymbol{\mathcal{P}}_p \boldsymbol{\Delta} \mathbf{C}_{\tilde{\phi}} \tilde{\boldsymbol{x}}_{\phi}[k]\end{aligned} \tag{5-15}$$

where $\tilde{\boldsymbol{x}}_{\phi}[k]$ is obtained from the dynamical filter as defined in Eq. (5-9).

5-5 Results

In order to validate the performance of the two methods for derivative estimation as proposed in this chapter, numerical simulations are performed on both the methods.

The simulation involves estimation of the 1^{st} , 2^{nd} and 3^{rd} order derivatives of a signal with tractable derivatives, namely $\phi(t) = \sin 2t^2$. Choosing such a signal allows for the comparison between the estimated derivative signals and the true derivative signals, as the derivatives of sinusoids[3] can be analytically determined up to an infinite order.

Finally, the relative performance of the two methods is compared. Their relative comparison in conjunction with the more practical pros and cons of the two methods is the foundation on which my final proposition for generalized signal estimation is based.

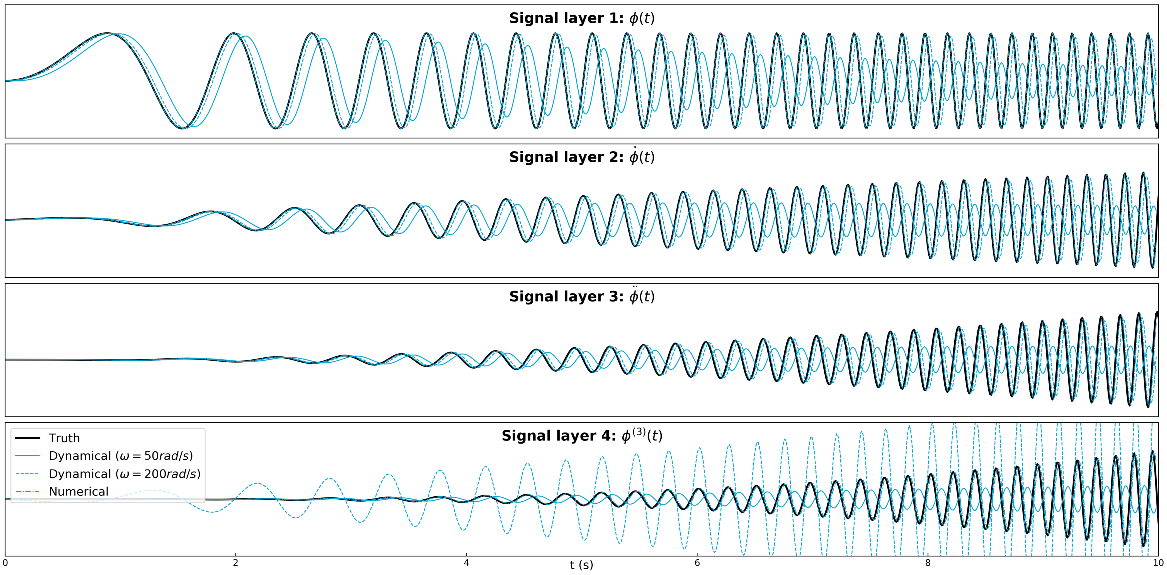

Figure 5-1: with two different cut-off frequency settings. From this figure it can be observed that the dynamical filters exert dynamical behaviour, and that the numerical filter tracks the signals more accurately than both dynamical filters. $\phi(t) = \sin(2t^2)$ T = 10 s. Source: github.com/lznidarsic/sir/

Most notable in this figure is the clearly present behaviour of the dynamical filter, which causes lag and an amplitude decrease when signal approaches the cut-off frequency. It would then be expected that increasing the cut-off frequency will yield more accurate signal estimation performance. However, as can be observed in the figure denoting the fourth embedding layer, the filter with a cut-off frequency of $200Hz$ deviates strongly from the true signal. This instability can be explained by the build-up of round-off errors which root in the discretization.

Considering the numerical filter, no errors can be visually observed, except upon very close inspection of the figure depicting the fourth embedding layer. It must be noted, however, that when the signal would have contained a certain degree of roughness, e.g. if it were perturbed with a level of Pink, White, Blue or Violet noise, the numerical filter would turn to instability in the deeper layers.

Conclusion Based on the results as discussed in this section, and the fact that the stable filter relies on a tuning parameter which may or may not destroy the essential information, or might even add wrong information to the embedded data, I propose to use the numerical differentiator for on-line DEM. Furthermore, I propose to use the numerical differentiator in combination with the numerical predictor for on-line DEM with embedded predictions.

[3] Along with cosinusoids, gaussians, etc.

> **Remark:** It might be that at some point in further research the need emerges to include compatibility with White noise into DEM. In this case, it might prove to be necessary to include a kind of filter which can estimate generalized signals perturbed with white noise. Then, the dynamical filter as proposed in this chapter might prove to be a solution.

5-6 Answers to research questions

To conclude this chapter I briefly circle back to the research questions as formulated in the introduction to answer them based on the content of this chapter.

What feasibility problems arise when translating DEM to a filtering and system identification method? For two of the three proposed methods, any implementation will rely on data that is not readily available, namely the higher order derivative signals or future predictions of the input and the output signals.

How can these problems be solved? Two different methods for estimating these derivative signals are proposed, the first being a simple causal numerical differentiating scheme, which is efficient but inaccurate. The second is a stable filter with a tunable cut-off frequency, which is, in theory, less prone to instability, at the cost of computational efficiency.

Do the proposed solutions to the feasibility perform accurately? Simulations showed that both solutions find some estimate of the generalized signals. However, the stable approach showed, contrary to expectation, to be more sensitive to discretization errors. Furthermore, performance of the stable filter depends on tuning. Therefore I propose to use the numerical estimation method.

Chapter 6

Results

In this chapter I will discuss the system, model, parametrization and signal properties used for the numerical simulations in this chapter. These are then performed on the methods for filtering and system identification as proposed in the previous chapters. The results of the numerical simulations are discussed. Ultimately, this chapter provides answers to research questions 9 and 10

6-1 Dataset, simulation and parametrization

This chapter will include numerical simulations of all the methods on data obtained from a 2^{nd} order spring-mass damper system, excited with a persistently exciting input signal and two different types of noise. All data will be from a simulation with $T = 10$ and $\Delta\mathrm{t} = 10^{-2}$, i.e. $N = 1001$.

System As mentioned, the system is a 2^{nd} order spring-mass damper system, from which only the position is available for measurement. The system has been discretized preceding the simulation using forward Euler, i.e. the simulation data is from an actual discrete-time system:

$$\begin{aligned} \boldsymbol{x}[k+1] &= \mathbf{A}_d\boldsymbol{x}[k] + \mathbf{B}_d\boldsymbol{u}[k] + \boldsymbol{w}[k] \\ \boldsymbol{y}[k] &= \mathbf{C}_d\boldsymbol{x}[k] + \mathbf{D}_d\boldsymbol{u}[k] + \boldsymbol{z}[k] \end{aligned} \tag{6-1}$$

Following the Euler discretization, the parametrization brings:

$$\mathbf{A}_d(\boldsymbol{\theta}) = \begin{bmatrix} 1 & \Delta\mathrm{t} \\ -\Delta\mathrm{t}\theta_1 & 1-\Delta\mathrm{t}\theta_2 \end{bmatrix}, \quad \mathbf{B}_d(\boldsymbol{\theta}) = \begin{bmatrix} 0 \\ \Delta\mathrm{t}\theta_3 \end{bmatrix}, \quad \mathbf{C}_d(\boldsymbol{\theta}) = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad \mathbf{D}_d(\boldsymbol{\theta}) = \begin{bmatrix} 0 \end{bmatrix} \tag{6-2}$$

with $\boldsymbol{\theta} = \begin{bmatrix} k/m & d/m & 1/m \end{bmatrix}^\top$, k the spring constant, d the damping coefficient and m the mass.

Model It is common for state-space identification methods to parametrize all entries of the SS model[1]. However, as the SS representation of systems is not unique, doing so will not necessarily bring a final model estimate which represents the system one-to-one, making

[1] and thus assume no structure in the matrices

hidden state estimation performance hard to evaluate. A solution would be to constrain structure in the model before estimation[2] [3]. For the scope of this thesis I will assume the model structure to be the equal to the system's structure. In other words, the model will be an exact copy of the system as presented in Eq. (6-1) and Eq. (6-2), the parameters $\boldsymbol{\theta} = [\theta_1,\ \theta_2,\ \theta_3]^\top$ of which are to be estimated. Doing so will also allow direct evaluation of the parameter-estimation performance by comparison of the estimated parameters with the real system's parameters.

Noise The first type of noise will be Gaussian-convolved white noise with a very thin kernel width of $\sigma = 1\Delta\text{t}$ (0.01 s), such that the realization can be considered as white noise[4]. The second type of noise will be Gaussian-convolved white noise with a wide kernel width of $\sigma = 10\Delta\text{t}$ (0.1 s), such that the realization can be considered as Gaussian-convolved white noise. The white noise preceding both the Gaussian filters will have $\mathbf{Q} = 0.1\mathbf{I}$, $\mathbf{R} = 0.2$. The near-White noise and corellated noise-sequences can be observed in Figure 6-1.

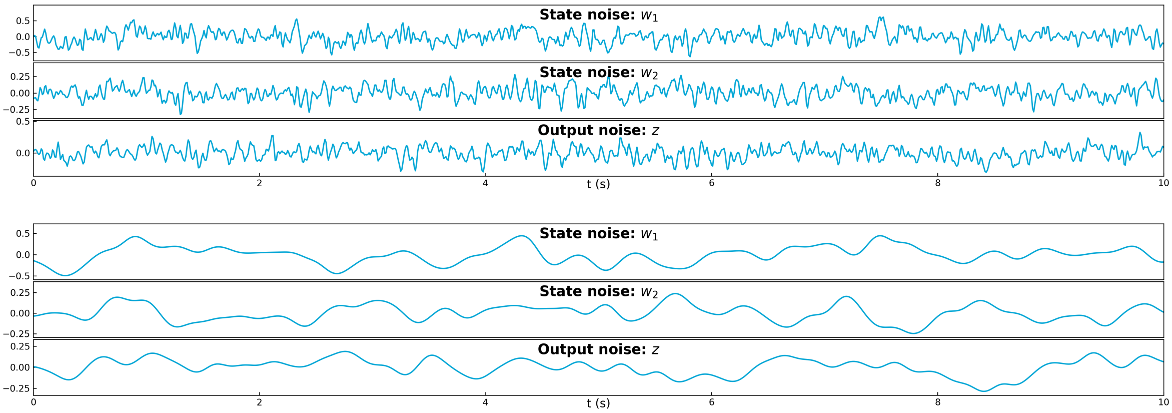

Figure 6-1: The two different noise signals . Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. $\Delta\text{t} = 10^{-2}$ s. $\text{T} = 10$ s. Source: github.com/lznidarsic/sir/

Input To ensure there is enough information in the collected data to properly identify the system, the data must be collected whilst the system is excited with an input signal following the persistence of excitation property [**?**, 18, 20]. For the simulation data used in this thesis, the input signal is a sum of sinusoids as can be observed in figure Figure 6-2.

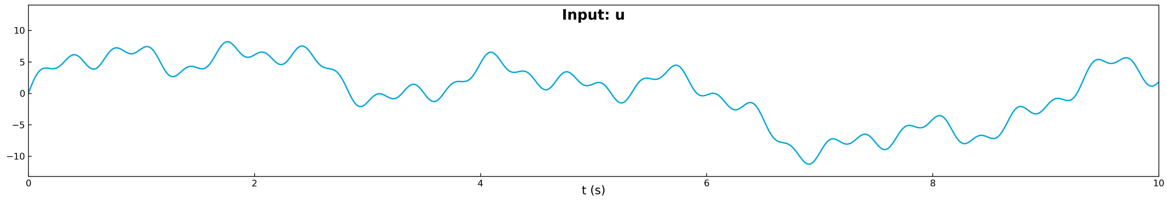

Figure 6-2: Persistently exciting input signal , sum of sinusoids. $\Delta\text{t} = 10^{-2}$ s. $\text{T} = 10$ s. Source: github.com/lznidarsic/sir/

[2] If the structure is unknown, the simplification can also be achieved via canonical realizations. Canonical realizations for 2^{nd} order systems contain four parameters, and thus for a true canonical realization the model would have had to be parametrized in the 0-entry of the **B**-matrix as well.

[3] Another way to compare the model estimate with the system would be via similarity transforms

[4] Using actual white noise introduces problems regarding the temporal correlation matrix

6-2 Filtering

Underlying the performance and convergence behaviour of the system identification methods lies the performance of the filters which they employ for state estimation. As a first test, all system parameters are assumed known, and the filtering methods are used for hidden state estimation on the dataset as described in the previous section. The filtering results for both noise settings can be observed in Figure 6-3 to Figure 6-6 respectively.

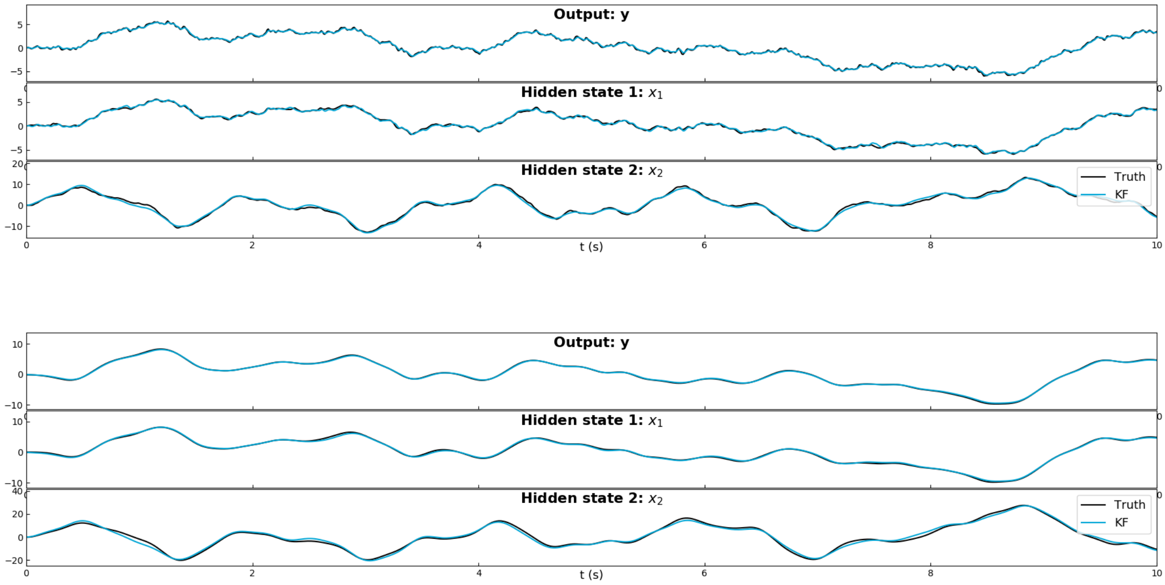

Figure 6-3: Results of the Kalman filter method calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. $\Delta t = 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

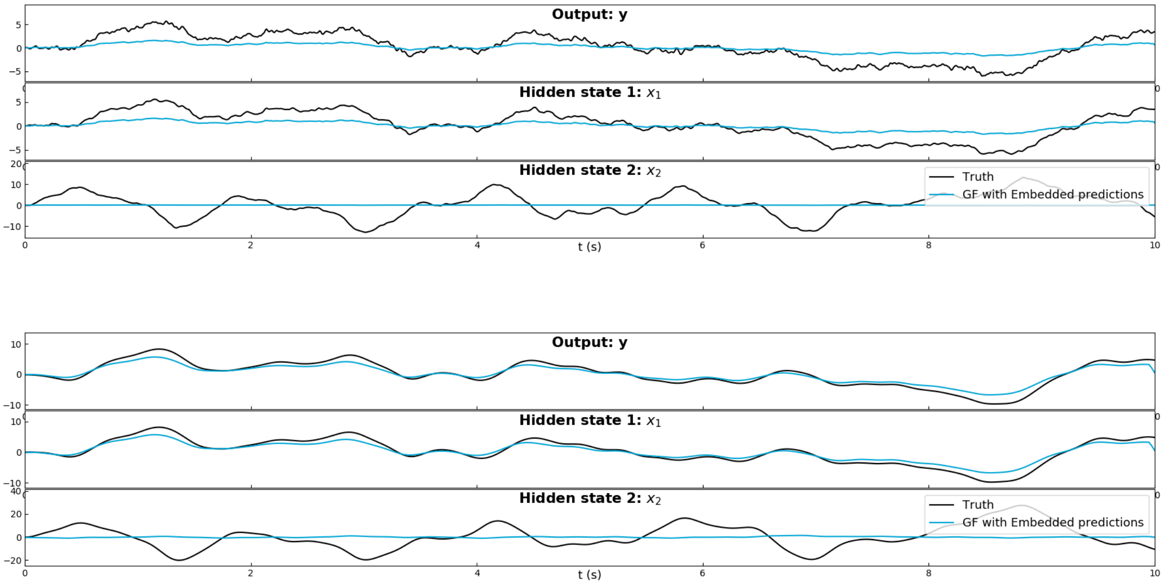

Figure 6-4: Results of the Generalized Filter with embedded predictions method calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

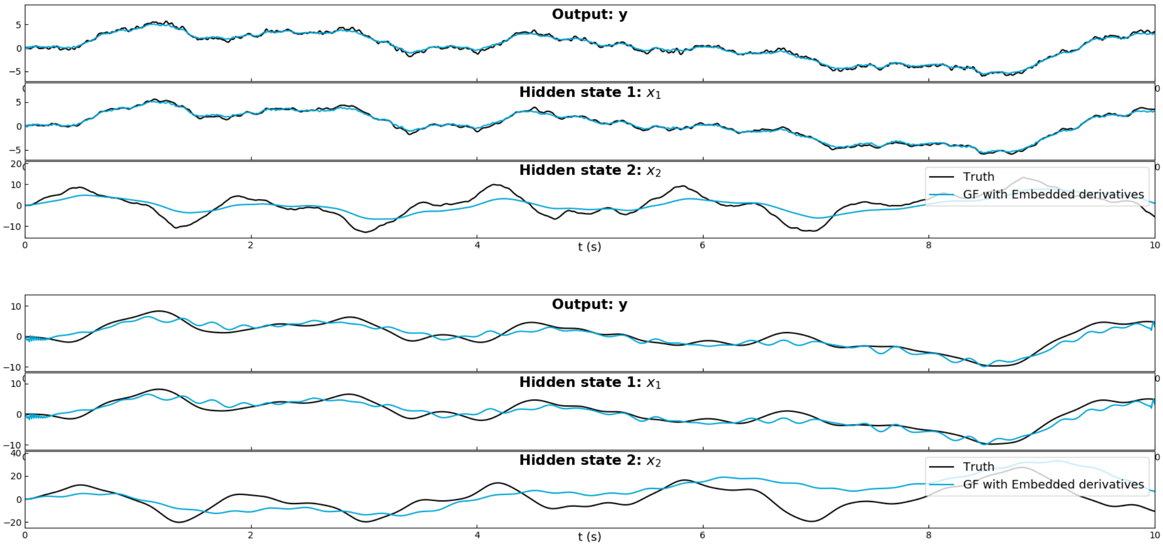

Figure 6-5: Results of the Generalized Filter with embedded derivatives method calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

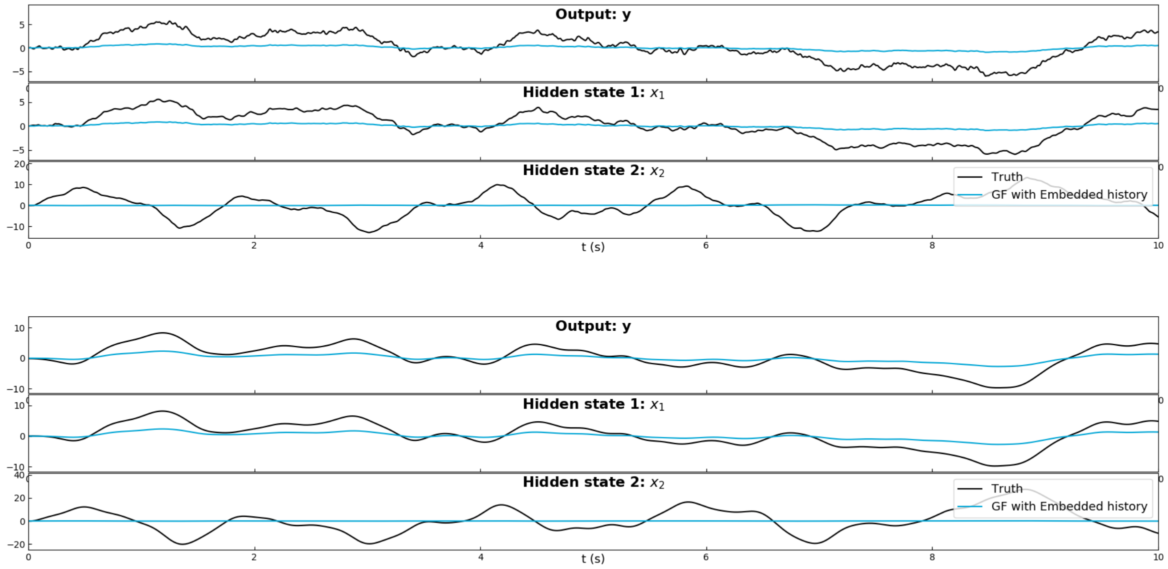

Figure 6-6: Results of the Generalized Filter with embedded history method calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. $\Delta \mathsf{t} = 10^{-2}$ s. $\mathsf{T} = 10$ s. Source: github.com/lznidarsic/sir/

The conclusion that can immediately be drawn from the figures is that in their current setting the filtering performance[5] of all three Generalized Filters is inferior to that of the conventional Kalman filter, both for noise that is almost white and for noise that is clearly correlated.

It must be noted that the filtering performance of all generalized filters could have been majorly improved by increasing the updating gain $\alpha_{\tilde{x}}$. However, increasing the gain further than the values that have been used for the depicted simulation yielded instability. Identifying the source of this instability and subsequently finding a means to overcome it, will highly increase the performance of all generalized filters and will yield a better comparison between the behaviour of the filters.

A possible candidate for the source of instability are the internal consistency mismatch in initial generalized input, hidden state and output due to a wrongly constructed $\boldsymbol{\mathcal{D}}$ matrix. Another possible candidate is the temporal correlation matrix $\mathbf{S}$, which would explain the behaviour of the DEM-EP and the DEM-EH methods.

A possible way to improve the performance of the generalized filters, as suggested to me by my supervisor P. Mohajerin Esfahani, would be to use a generalized Kalman updating scheme rather than gradient-descent Free-Energy minimization. Together we worked out that the coupling between different embedding layers would can then be achieved via the updating gain, which will be partly determined by the precision matrices and thus include te temporal correlation model. This last statement is particularly important to keep the ability to filter under the presence of corellated noise, as without coupling there would be no added value in generalization.

[5] Measured in MSE between actual hidden state and estimated hidden state

6-3 Parameter-cost optima

In this section the theoretical parameter estimation performance of the proposed methods will be tested by means of cost function evaluation on the system and dataset as proposed in the first section of this chapter. The performance will be evaluated as the distance between the parameters that are at the minimum of the cost-function[6], and the real parameters, i.e. the parameters of the real system.

6-3-1 Cost optima with known hidden states

Before the parameter estimation methods can be properly tested in conjunction with their corresponding filtering methods, their stand-alone performance must be evaluated. This can be achieved by assuming the (generalized) hidden states $(\mathbf{x}, \tilde{\mathbf{x}})$ as known, and thus evaluating the cost functions only in terms of a 1-step prediction from this known hidden state.

For each method two types of cost functions will be evaluated. The first being the theoretical cost, where errors are obtained in terms of the distance between the predicted (generalized) hidden state and the real one:

$$\begin{aligned} \boldsymbol{\varepsilon}_{x,\text{theoretical}}[k] &= \boldsymbol{x}[k+1] - \mathbf{A}\boldsymbol{x}[k] - \mathbf{B}\boldsymbol{u}[k] \\ \boldsymbol{\varepsilon}_{\tilde{x},\text{theoretical}}[k] &= \tilde{\boldsymbol{x}}[k+1] - \tilde{\mathbf{A}}\tilde{\boldsymbol{x}}[k] - \tilde{\mathbf{B}}\tilde{\boldsymbol{u}}[k] \end{aligned} \tag{6-3}$$

The second being the actual cost function that the method minimizes, where errors are obtained in terms of the distance between the predicted (generalized) hidden state and the estimate that the filter provides.

$$\begin{aligned} \boldsymbol{\varepsilon}_{x,\text{practical}}[k] &= \mathbf{K}(\mathbf{y}[k+1] - \mathbf{C}(\mathbf{A}\boldsymbol{x}[k] - \mathbf{B}\boldsymbol{u}[k])) \\ \boldsymbol{\varepsilon}_{\tilde{x},\text{practical}}[k] &= \boldsymbol{\mathcal{D}}\tilde{\boldsymbol{x}}[k] - \tilde{\mathbf{A}}\tilde{\boldsymbol{x}}[k] - \tilde{\mathbf{B}}\tilde{\boldsymbol{u}}[k] \end{aligned} \tag{6-4}$$

with $\boldsymbol{\varepsilon}_{\tilde{x}}$ implemented for all three flavour of DEM, and their shift matrices $\boldsymbol{\mathcal{D}}$ and embedded signals $\tilde{\boldsymbol{x}}$ according to the theory[7] as described in chapter 4. With respect to the DEM-based methods, the difference between the two costs can be considered a measure of much error the $\boldsymbol{\mathcal{D}}$ operator introduces. The cost functions of all proposed methods and expectation maximization (EM) in both the noise settings can be observed in figure Figure 6-7 to Figure 6-10.

> **Remark:** Since there are three parameters in the system, the full cost function cannot be properly depicted. Rather, the costs are evaluated per-parameter, with the other parameters set to their real parameter values[a]. If then for all three parameters the cost minimum is on the real parameter, it can be concluded that at least some local optimum of the method is on the real system parameters.
>
> [a] i.e. I am evaluating projections of the cost functions on planes spanned by the real parameters

[6] i.e. the parameter the algorithm will converge towards

[7] in this section the subscripts p, d and h were omitted to denote that the equation would be equivalent for all three cases.

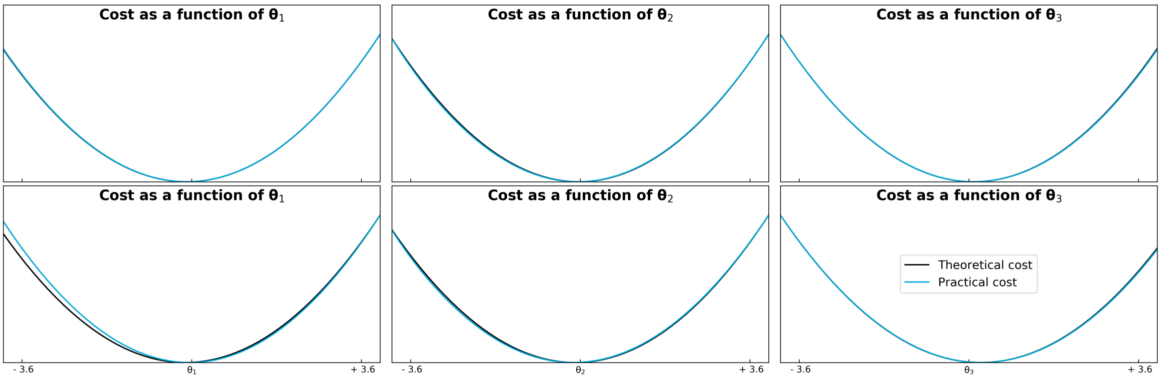

Figure 6-7: EM: The negative log-likelihood cost function with known hidden states calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top row of figures: near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom row of figures: convolved noise-signal signal, kernel width $\sigma = 0.1$ s. From the figure it can be observed that noise correlation shifts both the theoretical and practical optima away from the real parameters. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

From Figure 6-7 it can be observed that, given known hidden states, the EM method has its theoretical optimum on the real parameters for both noise cases and its practical optimum very close to, if not exactly on, the real parameters as well. This is not a surprise, as the expected failure of the EM algorithm will arise as a result of poor filtering performance rather than a shifted parameter optimum. Consequently, given some filter that is able to accurately estimate the hidden states under the presence of coloured noise, the EM parameter estimation step would still perform close-to optimal.

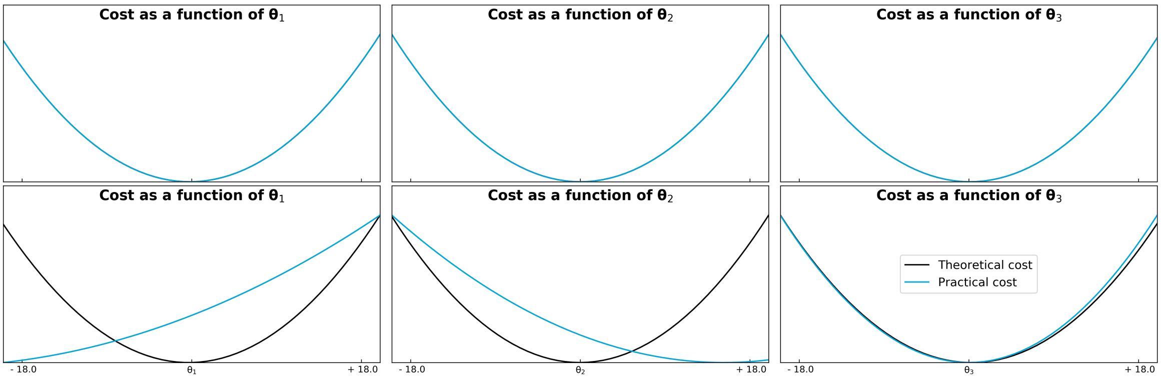

Figure 6-8: DEM: The Free-Energy with embedded predictions cost function with known hidden states calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. Δt $= 10^{-2}$ s. From the figure it can be observed that though noise correlation shifts the practical optimum away from the real parameters, the theoretical optimum remains on the real parameters. T $= 10$ s. `github.com/lznidarsic/sir/demo_parameter_estimation_cost.py`

From Figure 6-8 it can be observed that, given known hidden states, the DEM method with embedded predictions has both its theoretical and practical optimum on the real parameters for the near-white noise case. In other words, if its corresponding filtering method is able to accurately infer the hidden states[8] given known parameters, and furthermore the accuracy of the hidden state estimates increases with increasing parameter accuracy[9], then this method would be able to infer the real parameters of the system.

However, the bottom row of Figure 6-8 clearly shows that, though the theoretical optimum of the cost function remains on the real parameters, the practical does not. This implies that the parameter estimation performance of the method is not invariant to noise correlation. In fact, the negative influence of the noise correlation on the parameter estimation accuracy is larger for DEM-EP than for regular EM. The fact that the error increases with increasing correlation suggests an error in the Temporal Corellation Matrix that I defined for the generalized signals with embedded derivatives.

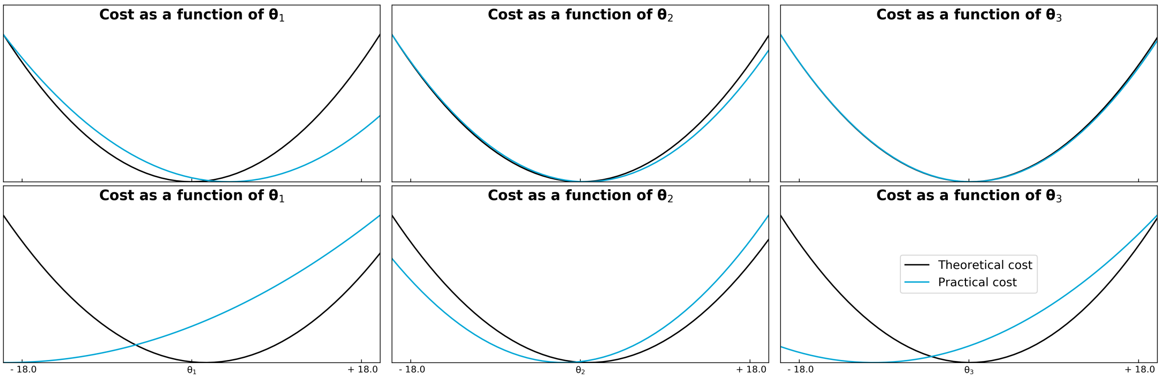

Figure 6-9: DEM: The Free-Energy with embedded derivatives cost function with known hidden states calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. From the figure it can be observed that the theoretical optimum is on the real parameters but the practical optimum is not, and that the error increases with increased correlation. $\Delta t = 10^{-2}$ s. $T = 10$ s. Source: github.com/lznidarsic/sir/

From Figure 6-9 it can be observed that, given known hidden states, the DEM method with embedded derivatives has its theoretical optima on the real parameters for both noise cases, but its practical optima are not. The accuracy of the practical cost deteriorates with increasing correlation.

[8] which I showed in the previous section that unfortunately it cannot

[9] again, I showed that unfortunately it does not

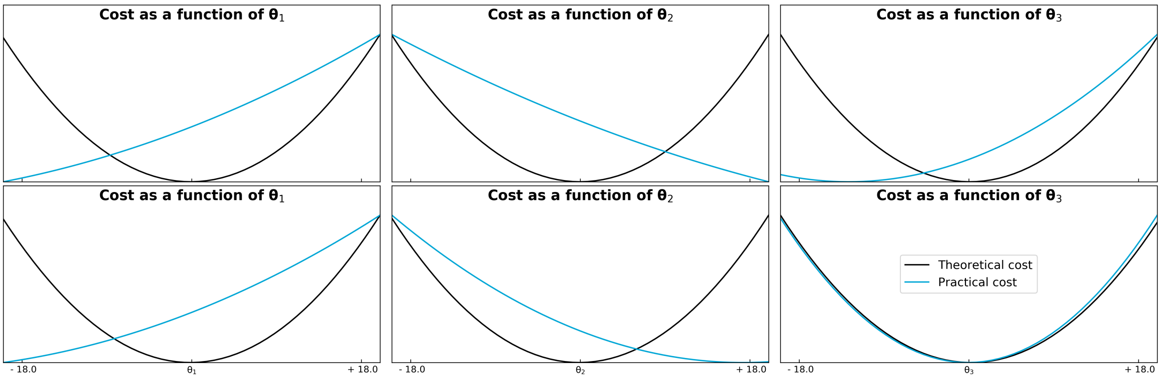

Figure 6-10: DEM: The Free-Energy with embedded history cost function with known hidden states calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

From Figure 6-10 it can be observed that, given known hidden states, the DEM method with embedded derivatives has its theoretical optima on the real parameters for both noise cases, but its practical optima are not. There is no noticeable difference in accuracy of the practical cost with increasing correlation.

The results presented in Figure 6-10 and Figure 6-10 suggest that using shift operators within the Free-Energy for parameter estimation introduces errors which seriously deteriorate the performance of DEM.

A possible workaround would be to simply replace the $\boldsymbol{\mathcal{D}}\hat{\tilde{\boldsymbol{x}}}[k-1]$ terms with $\hat{\tilde{\boldsymbol{x}}}[k]$ such that the internal consistency error yields:

$$\boldsymbol{\varepsilon}_{\tilde{x}}[k] = \hat{\tilde{\boldsymbol{x}}}[k] - \hat{\tilde{\mathbf{A}}}\hat{\tilde{\boldsymbol{x}}}[k-1] - \hat{\tilde{\mathbf{B}}}\hat{\tilde{\boldsymbol{u}}}[k-1]$$

Of course, this approach will still not solve the problem when the filters don't provide accurate estimates for the hidden states. Also note that changing the Free-Energy as suggested will bring the definition of DEM closer to EM, especially when combined with my suggestion for improving the filters via a Kalman scheme. In that case, DEM will essentially be EM with generalized coordinates.

6-3-2 Optima with unknown hidden states

Now that the independent performance of both the parameter estimation- and filtering methods have been established, their joint performance must be evaluated. This can be achieved by considering once more the cost functions as dependent on the unknown parameters, but now the hidden states are also unknown and are thus estimated using corresponding filters that each method includes.

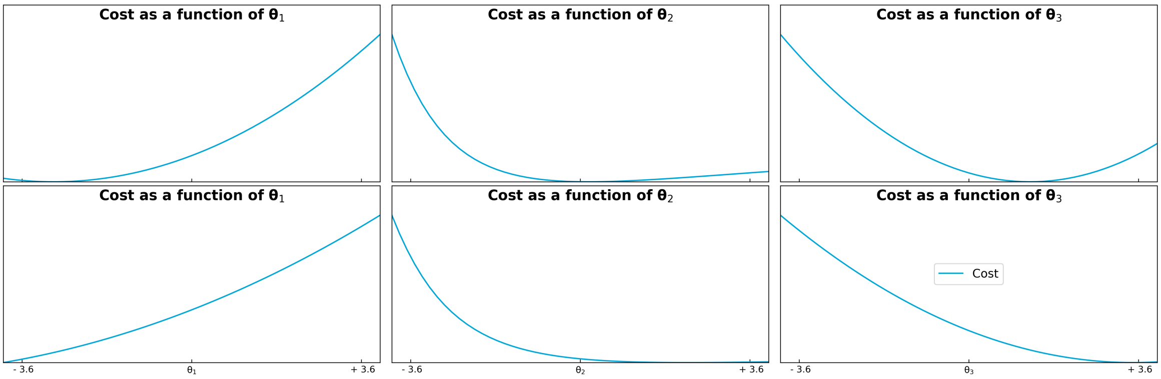

Figure 6-11: EM: The negative log-likelihood cost function with unknown hidden states calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top row of figures: near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom row of figures: convolved noise-signal signal, kernel width $\sigma = 0.1$ s. $\Delta t = 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

From Figure 6-11 it can be observed that the EM method has practical optimum close to the real parameters [10]. However, note that the optima are much closer to the real parameters in the near-White noise as compared to the correlated noise case, providing proof to the statement that noise correlation deteriorates the performance of EM.

[10] note that the scale of the parameter-space x-axis is much smaller for EM than in the DEM-based methods. Decreasing the parameter interval was necessary for EM due to instability for parameter values which rendered the model unstable.

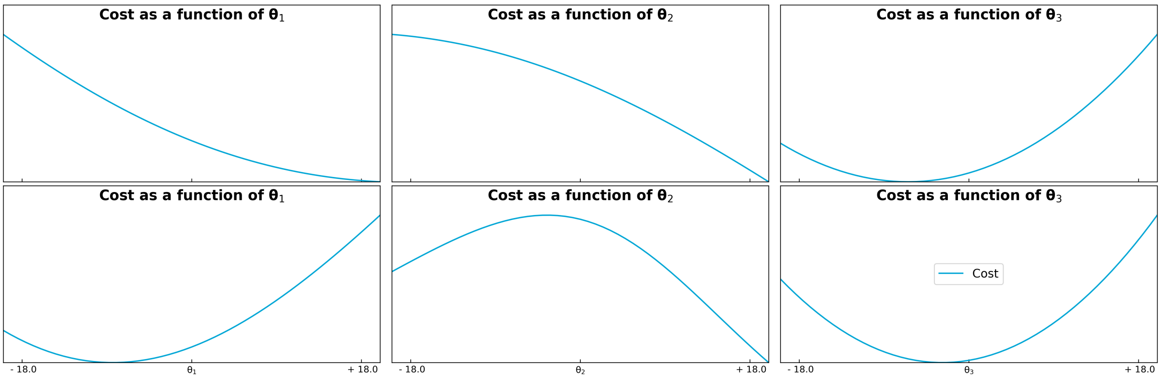

Figure 6-12: DEM with embedded predictions: The Free-Energy cost function with unknown hidden states calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

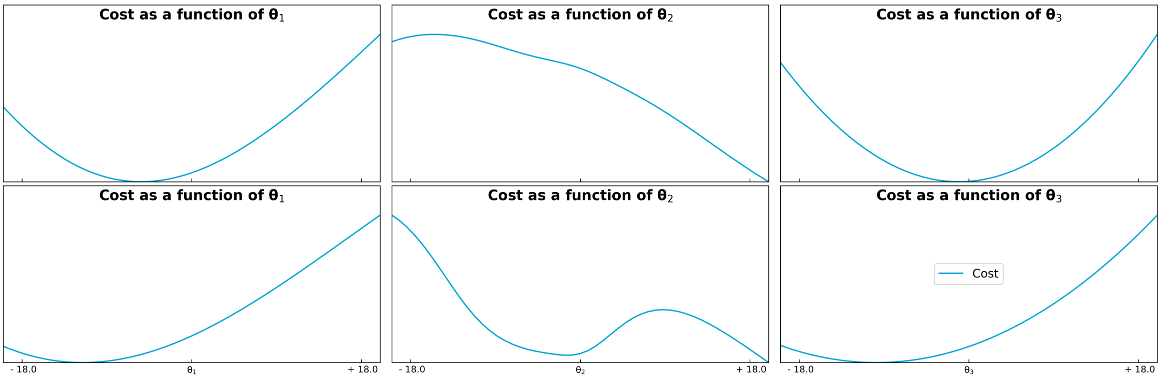

Figure 6-13: DEM with embedded derivatives: The Free-Energy cost function with unknown hidden states calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

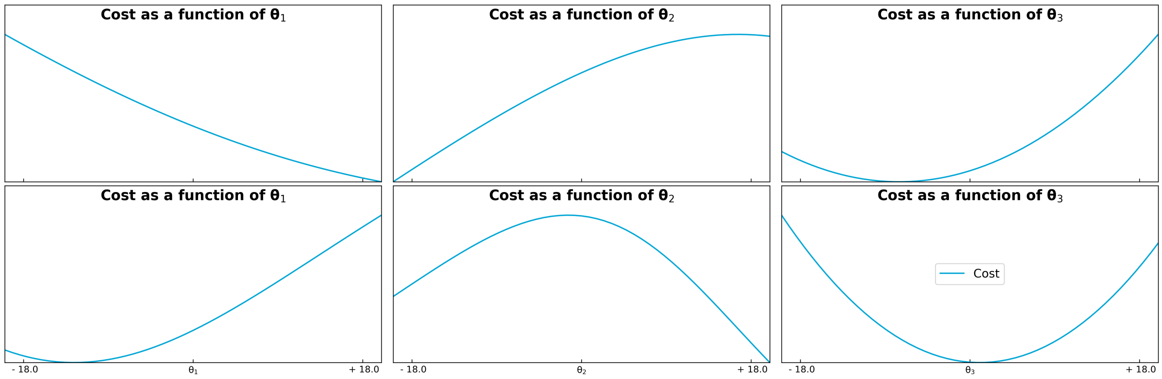

Figure 6-14: DEM with embedded history: The Free-Energy cost function with unknown hidden states calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

From Figure 6-12, Figure 6-13 and Figure 6-14 it can be observed that none of the proposed DEM-based methods have their optima close to the real parameters. Even stronger, some of the cost functions have an don't appear to have an optimum at all, which would render the methods unstable. Note however, as both the parameter and state- estimation methods do not perform adequately individually, it was to be expected that their joint performance would not be accurate. Nonetheless, the results presented in this section provide additional proof to the conclusion that DEM in its current implementation does not outperform EM in a correlated noise setting.

6-4 System Identification

In this section I will provide numerical simulations of all proposed methods, both in an off-line and an on-line setting. Note that these do not add directly to the conclusions made in this thesis. However, as a proof-of concept of all the methods described in this thesis as well as additional evidence to the statements made in previous sections, it is important to observe the behaviour of the algorithms from a practical perspective. In other words, to show that when applying the methods to a real (simulated) system identification problem, they behave as theory predicted.

6-4-1 Offline system Identification

Figure Figure 6-15 to Figure 6-18 depict the numerical simulation results for an off-line system identification setting. The final hidden state estimates are not shown, as for all the DEM-based methods the parameters values tend to infinity, rendering the final hidden state estimates as invalid numbers. This result was to be expected based on the (local) non-convexity of the cost functions as discussed in the previous section. The simulations do show that the performance of the EM-method deteriorates as noise correlation increases.

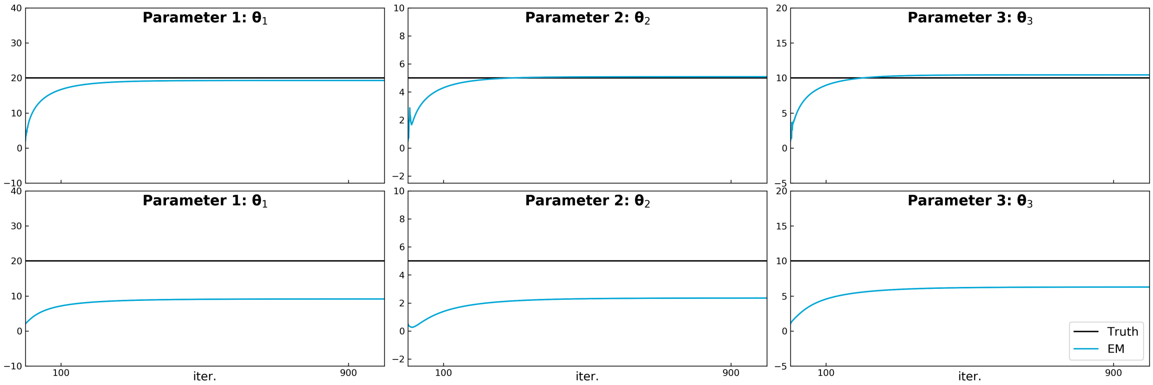

Figure 6-15: EM: Offline system identification results calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top row: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom row: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. This figure shows that noise correlation deteriorates the performance of EM. $\Delta t = 10^{-2}$ s. $T = 10$ s. github.com/lznidarsic/sir/demo_offline_parameter_estimation.py

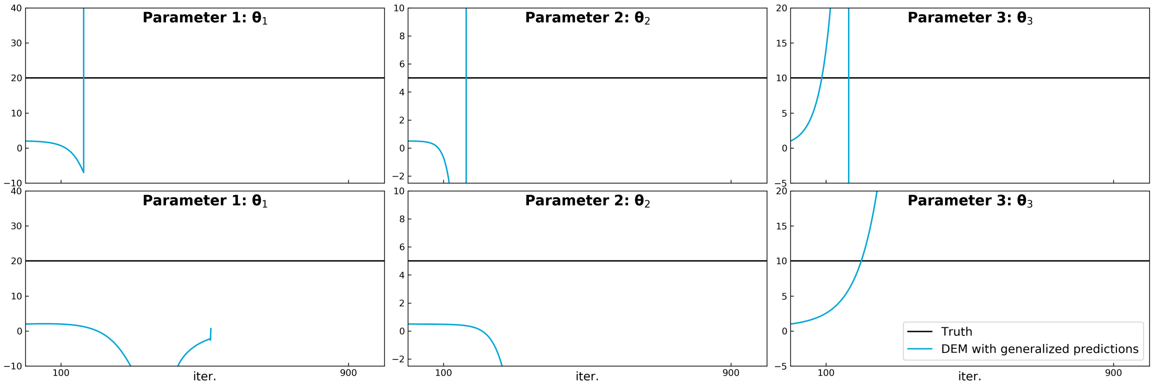

Figure 6-16: DEM with embedded predictions: offline system identification results calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top row: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom row: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. This figure shows that DEM-EP is unstable in its current setting. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

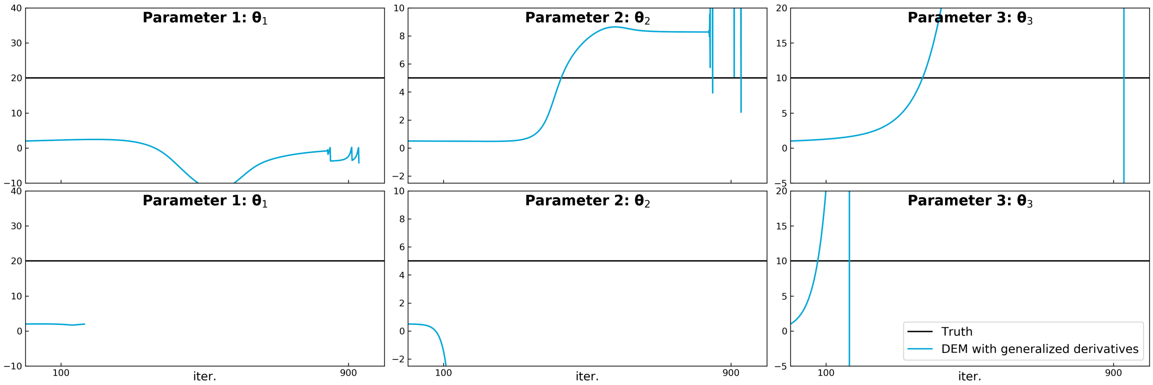

Figure 6-17: DEM with embedded derivatives: offline system identification results calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top row: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom row: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. This figure shows that DEM-ED is unstable in its current setting. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

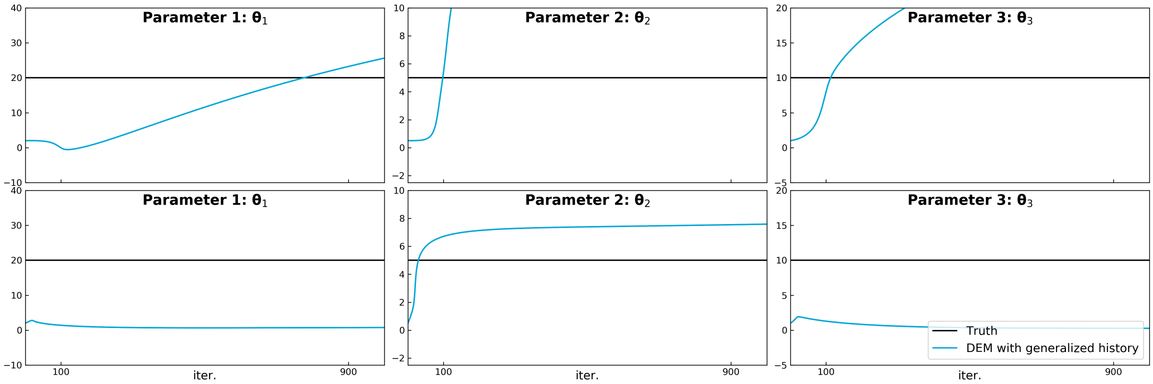

Figure 6-18: DEM with embedded history: offline system identification results calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top row: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom row: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. This figure shows that DEM-EH is unstable in its current setting. Δt $= 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

6-4-2 Online system Identification

Figure Figure 6-15 to Figure 6-18 depict the numerical simulation results for an on-line system identification setting. For all the DEM-based methods the parameters values tend to infinity. This result was to be expected based on the (local) non-convexity of the cost functions as discussed in the previous section. The simulations do show that the performance of the EM-method deteriorates as noise correlation increases.

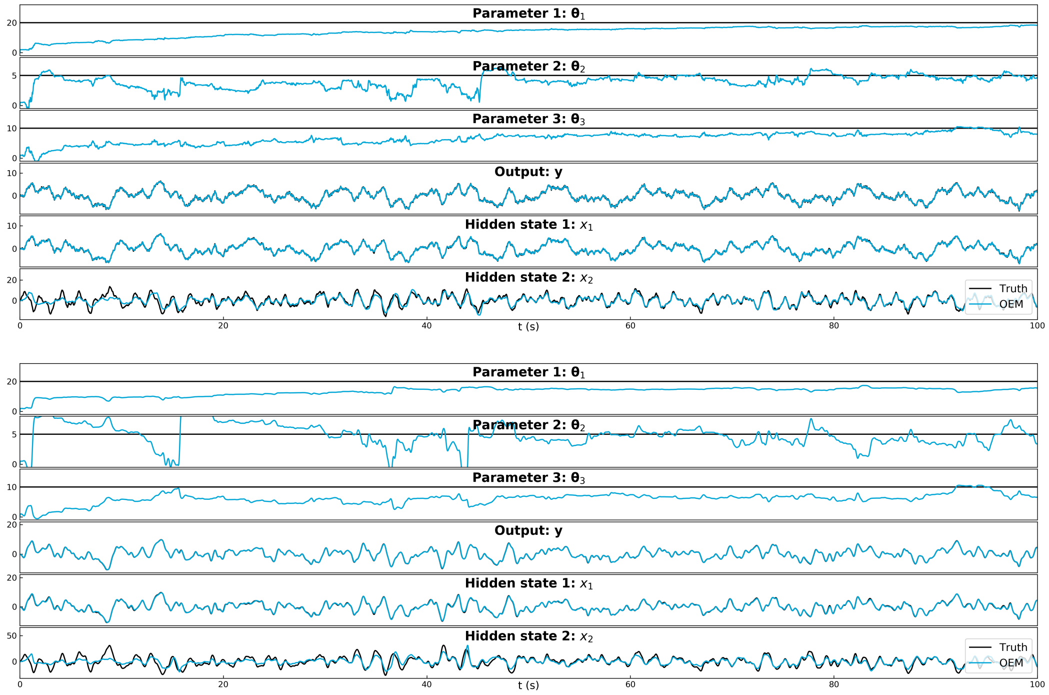

Figure 6-19: OEM: On-line system identification results calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. This figure shows that noise correlation deteriorates the performance of EM. $\Delta t = 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

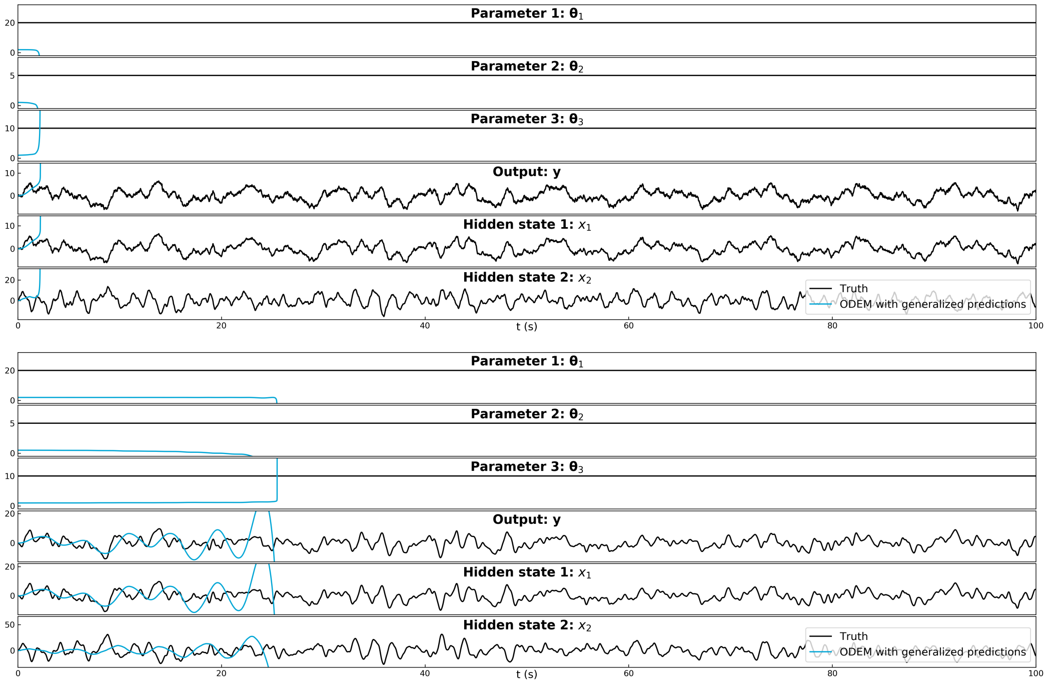

Figure 6-20: ODEM with embedded predictions: on-line system identification results calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. This figure shows that DEM-EP is unstable in its current setting. $\Delta \text{t} = 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

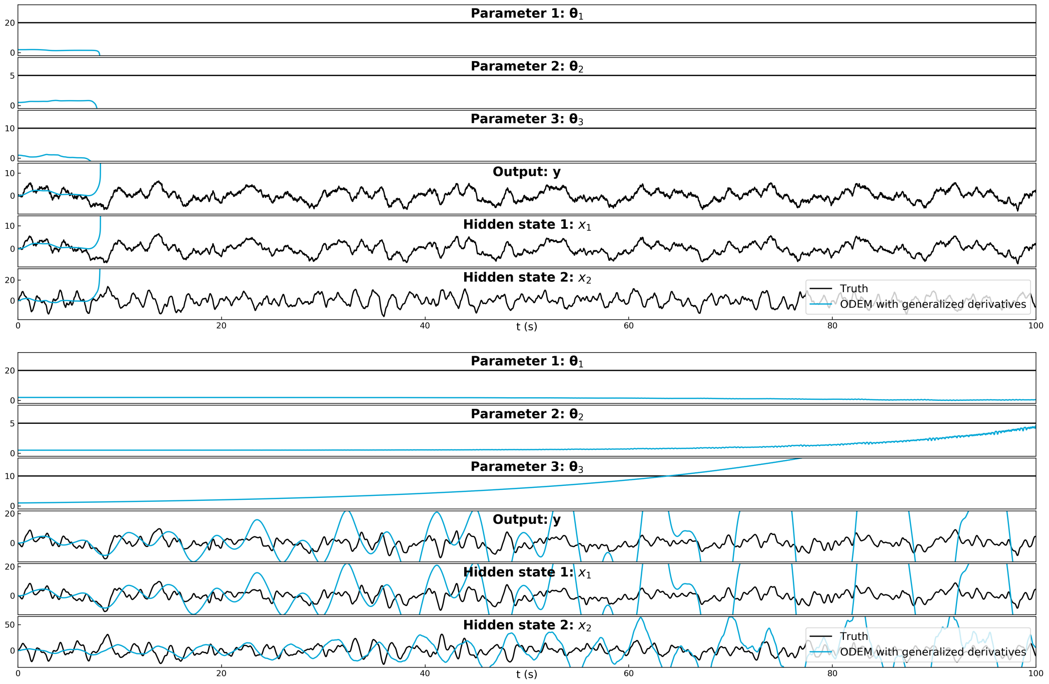

Figure 6-21: ODEM with embedded derivatives: on-line system identification results calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. This figure shows that DEM-ED is unstable in its current setting. $\Delta \text{t} = 10^{-2}$ s. T $= 10$ s. Source: github.com/lznidarsic/sir/

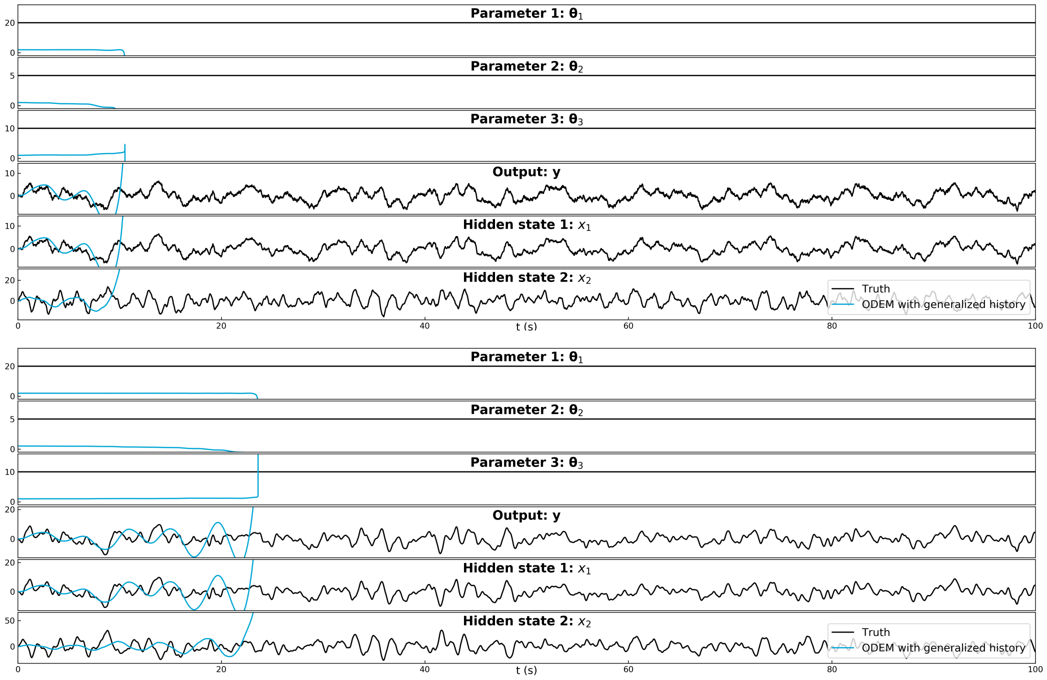

Figure 6-22: ODEM with embedded history: on-line system identification results calculated from an exemplary 2^{nd} order mass-spring-damper system perturbed with the two different noise signals. Top figure: Near-White noise signal, kernel width $\sigma = 0.01$ s. Bottom figure: Convolved noise-signal signal, kernel width $\sigma = 0.1$ s. This figure shows that DEM-EH is unstable in its current setting. $\Delta \mathrm{t} = 10^{-2}$ s. $\mathrm{T} = 10$ s. Source: github.com/lznidarsic/sir/

6-5 Answers to research questions

To conclude this chapter I briefly circle back to the research questions as formulated in the introduction to answer them based on the content of this chapter.

Does DEM outperform EM w.r.t. filtering under the presence correlated noise I showed that none of the proposed DEM-based filtering methods in their current setting outperform the conventional Kalman filter in terms of hidden state estimation. I suggested to replace the shift matrices with a Kalman update as a possible way to improve the state-estimation performance.

Does DEM outperform EM w.r.t. identification under the presence correlated noise I showed that none of the proposed DEM-based parameter estimation methods in their current setting outperform EM in terms of parameter estimation. I suggested to replace the shift matrices with the filtered state estimate as a possible way to improve the parameter-estimation performance.

Chapter 7

Conclusion

In this chapter I will provide a short summary of the main findings of this thesis, based on the answers of the research questions. I will then address the main research question. This chapter will conclude with an overview of the research contributions of this thesis and recommendations for further research

7-1 Summary

In this thesis I proposed three discrete-time methods for system identification under the presence of coloured noise, based on the continuous-time definition of the DEM-algorithm as proposed by neuroscientist K.J Friston.

The major translation steps from theoretical principle to feasible system identification methods involved discretization and overcoming the feasibility issues due to dependency of unavailable data. For the latter, two solutions were considered of which the numerical differentiator proved the best candidate.

I showed that none of the proposed DEM-based filtering methods in their current setting outperform the conventional Kalman filter in terms of hidden state estimation, due to instability issues. Should these instability issues be solved, then the filtering methods might outperform the Kalman filter when noise is correlated. I suggested to replace the shift matrices with a Kalman update as a possible way to improve the state-estimation performance.

I showed that none of the proposed DEM-based parameter estimation methods in their current setting outperform expectation maximization (EM) in terms of parameter estimation, but this can be mainly attributed to the poor performance of the filters. The theoretical optima of the Free-Energy cost functions proved to be invariant to noise correlation, providing strong evidence that when a different method for approximating the theoretical optima is included and a way to improve filtering performance is found, DEM might outperform EM in terms of parameter estimation when noise is correlated. I suggested to replace the shift matrices with the filtered state estimate as a possible way to improve the parameter-estimation performance.

7-2 Answer to the main question

Do the DEM-based methods for filtering and system identification as suggested in this thesis outperform EM for systems perturbed with correlated noise? No. In the current interpretations that were proposed in this thesis the filters suffer from instability issue which seriously deteriorate the hidden state estimation performance. As a result, the DEM-based system identification methods, which heavily rely on the hidden states estimated by the filters, cannot accurately identify the system. However, I have shown that the optima of the Free-Energy cost function as dependent on the parameters is invariant to noise correlation, which implies that, should further research find a means of overcoming the stability issues of the generalized filters and a better way to approximate the theoretical optimum for parameter estimation, there is evidence that DEM might outperform EM.

7-3 Contributions

A brief overview of the major contributions of this thesis:

- Showed that noise correlation deteriorates performance of EM
- Provided three discrete-time system identification methods based on DEM
- Implemented EM and my interpretations of DEM
- Laid the foundations of a system identification toolbox around EM and DEM in python which includes:

  various filters

  model structure (works both for linear and non-linear state-space systems)

  noise generator for correlated noise in various flavours

  derivative observers for generalized signal estimation

  various demos showing on all topics discussed in this thesis
- Validated the performance of DEM in terms of parameter estimation and filtering
- Provided evidence suggesting that performance of DEM in its current implementation is not invariant to noise correlation
- Provided suggestions on how the performance of the both the filters and the parameter estimation can be improved

7-4 Recommendations

In this section I will describe recommendations for further research. Some of these will build on findings of this thesis, and others will concern subjects that I considered outside the scope of this thesis.

Improving state-estimation performance The major step in improving the performance of the DEM-based system identification methods as discussed in this thesis, will be achieved by improving the performance of the filters. More specifically, finding a means such that the state-estimation gain $\alpha_{\tilde{x}}$ can be increased without causing instability. Doing so, will highly increase the hidden-state estimation accuracy of the filters.

Furthermore, if further research brings that the reason for the instability is inherent to how the filters are currently defined, it might be necessary to adopt a fundamentally different kind of filtering scheme. A possible candidate would be to replace the $\boldsymbol{\mathcal{D}}\hat{\tilde{x}}$ in $\hat{\tilde{x}}[k+1] = \boldsymbol{\mathcal{D}}\hat{\tilde{x}} - \partial_{\tilde{x}}\mathcal{F}$ with a more accurate prediction estimate, i.e. one calculated with the model or even a Kalman filter.

The latter might even be used without Free-Energy minimization if it can be proven that a generalized Kalman filter will directly provide the necessary coupling[1] to ensure the information in the deeper embedding layers is utilized. If it turns out that a generalized Kalman filter will yield no coupling between embedding layers, it might be necessary to combine it with Free-Energy minimization.

Improving parameter-estimation performance From the results presented in this thesis it has become evident that the parameter estimation scheme, even with sufficiently performing filters, will not converge towards the real parameters. The results did show, however, that this bad performance can be attributed to how the hidden state error is approximated. DEM as proposed by Friston relies heavily on the concept of shift operators to approximate the hidden state error, but the results of this thesis have shown that in a discrete-time setting these are a bad estimator for the generalized hidden state.

A possible workaround would be to simply replace the $\boldsymbol{\mathcal{D}}\hat{\tilde{\boldsymbol{x}}}[k]$ terms with $\hat{\tilde{\boldsymbol{x}}}[k+1]$ such that the internal consistency error yields:

$$\boldsymbol{\varepsilon}_{\tilde{x}}[k] = \hat{\tilde{\boldsymbol{x}}}[k+1] - \hat{\tilde{\mathbf{A}}}\hat{\tilde{\boldsymbol{x}}}[k] - \hat{\tilde{\mathbf{B}}}\hat{\tilde{\boldsymbol{u}}}[k]$$

Of course, this approach will still not solve the problem when the filters don't provide accurate estimates for the hidden states. Also note that changing the Free-Energy as suggested will bring the definition of DEM closer to EM, especially when combined with my suggestion for improving the filters via a Kalman scheme. In that case, DEM will essentially be EM with generalized coordinates. Nonetheless, working such a scheme out and evaluating its performance might be an interesting topic for further study.

The mean-field terms To this point it remains unclear what influence the mean-field terms have on the performance of the state- and parameter estimation. The main questions considering the terms are: does omitting the mean-field terms cause a shift in state- and parameter optima, if so, does this shift de- or increase the estimation accuracy[2], and if not, why are they there at all? The results of this thesis yielded insufficient insight in this matter, but they are fundamental to the Friston-defined DEM, and thus a better understanding in their functionality is desired.

[1] via the Kalman gain, which relies on the inverse precisions as discussed in this thesis

[2] i.e. in what direction is the shift? Towards, or away from the real optima

Include hyper-parameter estimation into EM and DEM, and find a means to validate its performance Both EM as defined by Dempster in [6] and DEM as defined by Friston in [12] include mechanisms for estimating the parameters which determine the noise, i.e. the covariance matrices $\mathbf{Q}$ and $\mathbf{R}$. For a better foundation of the argument of autonomous robotics it will be very valuable to evaluate how inclusion of these mechanisms influences the parameter- and state-estimation performance of DEM and EM.

Input estimation To keep the scope of this thesis somewhat compact I decided to not consider the input-estimation part of DEM as defined by Friston. I do however recommend a follow-up study in which the influence of input uncertainty on the state- and parameter estimation performance is evaluated. I firmly believe that without inputs one cannot estimate the model parameters, and without the model one cannot estimate the inputs, i.e. the number of degrees of freedom in the estimation exceeds the number of constraints, but I invite anyone to challenge me on this point of view.

Auto-tuning of $\boldsymbol{\alpha}_{\tilde{\boldsymbol{x}}}$ and $\boldsymbol{\alpha}_{\boldsymbol{\theta}}$ The fact that DEM relies on scaling factors that have to be manually tuned and do influence the performance of at least the state-estimation methods, is a very strong argument against the superiority of DEM. the method for the use of autonomous robots. A much more elegant solution, which would bring much more value to its application within autonomous agents, would be a method which does not rely on tunable scaling factors. I would therefore recommend to include either some heuristic trial-and-error based auto-tuning[3] of the scaling factors, or to trade-in the 1^{st} order gradient descent for a 2^{nd} order gradient descent which, though sensitive to non-convexity[4], does not rely on tuning and subsequently converges much faster.

Priors From my perspective the inclusion of Priors on the parameters, states, hyper-parameters or inputs defeats the purpose of mechanisms designed for estimating these variables. If the prior is correct, then why perform estimation at all, if it is wrong, then inclusion will shift the optima of the cost function towards this wrong value and thus deteriorate the performance of the estimation schemes. Rather, I would suggest using priors as initial estimates. I do however recommend an open debate and a thorough review on the meaning of the priors, as it might very well be that my argumentation is simply too blunt.

[3] e.g. prior to estimation, increase α to the maximum value that still renders stable results

[4] Can be solved by global optimization, i.e. multi-starting from (pesudo-)random initial estimates, which would strictly be necessary anyway

Appendix A

Detailed background on selected topics

A-1 Numerical approximation of generalized signals with embeded-ded derivatives

Given a causal sequence of data of a certain signal ϕ of length p, i.e. a generalized signal with embedded history [1] :

$$\tilde{\boldsymbol{\phi}}_h[k] = \begin{bmatrix}\phi[k] & \phi[k-1] & \phi[k-2] & \phi[k-3] & \cdots & \phi[k-N]\end{bmatrix}^\top$$

which we want to use to approximate the generalized signals with embedded derivatives of order p:

$$\tilde{\boldsymbol{\phi}}_d[k] = \begin{bmatrix}\phi[k] & \dot{\phi}[k] & \ddot{\phi}[k] & \cdots & \phi^{(p)}[k]\end{bmatrix}^\top$$

The derivative approximation is performed using 1st order backward Euler differentiation via:

$$\begin{aligned}
\dot{\phi}[k] &\approx \frac{1}{\Delta \mathrm{t}}(\phi[k] - \phi[k-1]) \\
\ddot{\phi}[k] &\approx \frac{1}{\Delta \mathrm{t}}(\dot{\phi}[k] - \dot{\phi}[k-1]) \\
&\approx \frac{1}{\Delta \mathrm{t}^2}(\phi[k] - 2\phi[k-1] + \phi[k-2]) \\
\phi^{(3)}[k] &\approx \frac{1}{\Delta \mathrm{t}}(\ddot{\phi}[k] - \ddot{\phi}[k-1]) \\
&\approx \frac{1}{\Delta \mathrm{t}^2}(\phi[k] - 3\phi[k-1] + 3\phi[k-2] - \phi[k-3]) \\
\phi^{(4)}[k] &\approx \frac{1}{\Delta \mathrm{t}}(\phi^{(3)}[k] - \phi^{(3)}[k-1]) \\
&\approx \frac{1}{\Delta \mathrm{t}^2}(\phi[k] - 4\phi[k-1] + 6\phi[k-2] - 4\phi[k-3] + \phi[k-4])
\end{aligned}$$

Repeating this procedure of substitution and expansion up to the order p, allows for the estimation of all the entries of the generalized signal. Then, vectorization of the derivative estimates such that it describes the generalized signal, yields the following matrix equation:

$$\tilde{\boldsymbol{\phi}}_d[k] \approx \boldsymbol{\Delta}^{-1}\boldsymbol{\mathcal{P}}_p \mathbf{I}_\pm \tilde{\boldsymbol{\phi}}_h[k]$$

with $\boldsymbol{\Delta} := \mathrm{diag}(\Delta \mathrm{t},\ \Delta \mathrm{t}^2,\ ...,\ \Delta \mathrm{t}^p)$, $\mathbf{I}_\pm$ an identity matrix with 1 on uneven rows and -1 on even rows and $\boldsymbol{\mathcal{P}}_p$ a Pascal matrix, which for an exemplary embedding order of $p = 6$ yields:

$$\boldsymbol{\mathcal{P}}_6 = \begin{bmatrix}
1 & & & & & & \\
1 & 1 & & & & & \\
1 & 2 & 1 & & & & \\
1 & 3 & 3 & 1 & & & \\
1 & 4 & 6 & 4 & 1 & & \\
1 & 5 & 10 & 10 & 5 & 1 & \\
1 & 6 & 15 & 20 & 15 & 6 & 1
\end{bmatrix}$$

[1] i.e. the only generalized signal that is directly measurable

A-2 Estimation of generalized signals with embedded derivatives using a stable filter

Given a stable a strictly proper transfer function of the form:

$$\frac{Y(s)}{U(s)} = \frac{b_{n+1}s^n + b_n s^{n-1} + ... + b_3 s^2 + b_2 s + b_1}{s^{n+1} + a_{n+1}s^n + a_n s^{n-1} + ... + a_3 s^2 + a_2 s + a_1} \tag{A-1}$$

Then, realization theory states that the coefficients of this transfer function can be directly used in a SISO controllable canonical state-space realization[2]:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \begin{bmatrix} & 1 & & \\ & & \ddots & \\ & & & 1 \\ -a_1 & -a_2 & \cdots & -a_{n+1} \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} b_1 & b_2 & \cdots & b_{n+1} \end{bmatrix} \boldsymbol{x}(t) \end{aligned} \tag{A-2}$$

Then, considering the transfer function of my filtered differentiator, the exponents can be expanded:

$$\begin{aligned} H(s) &= \frac{\lambda^{n+1}s^n}{(s+\lambda)^{n+1}} \\ &= \frac{\lambda^{n+1}s^n}{(s^{n+1} + c_{n+1}\lambda s^n + c_n\lambda^2 s^{n-1} + c_{n-1}\lambda^3 s^{n-2} + \ ... \ + c_2\lambda^n s + c_1\lambda^{n+1}} \end{aligned} \tag{A-3}$$

where the constants c_1, c_2, etc. are the 2^{nd} to $(n+1)^{\text{th}}$ entries of the last row of a $(n+1)^{\text{th}}$ order Pascal matrix. Then, substituting for all $a_{n+1} = c_{n+1}\lambda$, $a_n = c_n\lambda^2$ ... $a_1 = c_1\lambda^{n+1}$ and $b_{n+1} = \lambda^{n+1}$, $b_i = 0 \ \forall i \neq n+1$ and using Eq. (A-2) leads to the following realization of of the stable differentiator:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \begin{bmatrix} & 1 & & \\ & & \ddots & \\ & & & 1 \\ -c_1\lambda^{n+1} & -c_2\lambda^n & \cdots & -c_{n+1}\lambda \end{bmatrix} \boldsymbol{x}(t) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} 0 & 0 & \cdots & \lambda^{n+1} \end{bmatrix} \boldsymbol{x}(t) \end{aligned} \tag{A-4}$$

where $y(t) = \hat{u}^{(n)}(t)$, i.e. the filtered estimate of the n^{th} order derivative of u. Now, note how the state-to-output mapping states that the output signal, which is our n^{th} order derivative estimate of the input signal, is in fact the scaled $(n+1)^{\text{th}}$ entry of the hidden state. Also note how the state-transition matrix states that all entries in the layers of $\boldsymbol{x}$ are in fact obtained by in integration of their superseding layers. In other words, if the last entry of $\boldsymbol{x}$ is a scaled estimate of the n^{th} order derivative of u, then the second to last entry is a scaled estimate of the $(n-1)^{\text{th}}$ order derivative of u, and so on. Thus, the estimate of the full generalized signal can be obtained simply by retrieving the full scaled hidden state:

$$\hat{\tilde{\boldsymbol{u}}}(t) = \begin{bmatrix} \lambda^{n+1} & & & \\ & \lambda^{n+1} & & \\ & & \ddots & \\ & & & \lambda^{n+1} \end{bmatrix} \boldsymbol{x}(t) \tag{A-5}$$

[2] the same holds for the observable canonical realization

A-3 Off- and online learning

In the context of autonomous robotics, where system identification can be considered as an agent learning the behaviour of its environment, the learning process can be divided into two stages.

The first being the learning stage, in which the agent is allowed to freely control its input and is in a safe environment where it cannot do any harm. With these permissions, the agent is able to drive it's states through the full space of its dynamics, such that it is able to collect sufficient information to be able to infer the system parameters. This is analogous to stating that during this stage, the system is allowed to feed an input signal adhering to the persistence of excitation property[3] [20,21], and is thus not constricted to the control strategy. Furthermore, during this stage, the parameters can be updated iteratively after inferring the hidden states on the full batch of data, i.e. the agent can employ an off-line learning method. Such methods are generally less prone to local minima and are more efficient in terms of data.

When the learning stage is over, the robot will have learned its dynamics and can thus proceed to executing the task it was designed for. However, it generally occurs that during its lifetime, parameters slowly change due to temperature fluctuations, wear, changes in pressure or humidity. Therefore, an autonomous robot should continue its learning process parallel to task execution, such that it can update the model and filter for changing circumstances. In other words, the robot should employ an on-line learning method. It is of paramount importance that this on-line method is be built on the same principles as the off-line learning method, to avoid a shift in optima.

[3] A necessary condition for parameter convergence

A-4 Numerical approximation of generalized signals with embedded predictions

Following a very similar procedure as presented in the previous appendix, any (estimated) generalized signal with embedded derivatives:

$$\tilde{\boldsymbol{\phi}}_d[k] = \begin{bmatrix} \phi[k] & \dot{\phi}[k] & \ddot{\phi}[k] & \cdots & \phi^{(p)}[k] \end{bmatrix}^\top$$

can be translated to a generalized signal with embedded predictions of the form:.

$$\tilde{\boldsymbol{\phi}}_p[k] = \begin{bmatrix} \phi[k] & \phi[k+1] & \phi[k+2] & \cdots & \phi[k+p] \end{bmatrix}^\top$$

In fact, this translation step can be once again be achieved via Euler's method, only now instead of differentiation, the translation yields integration and we use the forward method rather than the backwards method:

$$\begin{aligned}
\phi[k+1] &\approx \phi[k] + \Delta\mathrm{t}\dot{\phi}[k] \\
\dot{\phi}[k+1] &\approx \dot{\phi}[k] + \Delta\mathrm{t}\ddot{\phi}[k] \\
\phi[k+2] &\approx \phi[k+1] + \Delta\mathrm{t}\dot{\phi}[k+1] \\
&\approx \phi[k] + 2\Delta\mathrm{t}\dot{\phi}[k] + \Delta\mathrm{t}^2\ddot{\phi}[k] \\
\dot{\phi}[k+2] &\approx \dot{\phi}[k] + 2\Delta\mathrm{t}\ddot{\phi}[k] + \Delta\mathrm{t}^2\phi^{(3)}[k] \\
\phi[k+3] &\approx \phi[k+2] + \Delta\mathrm{t}\dot{\phi}[k+2] \\
&\approx \phi[k] + 3\Delta\mathrm{t}\dot{\phi}[k] + 3\Delta\mathrm{t}^2\ddot{\phi}[k] + 3\Delta\mathrm{t}^3\phi^{(3)}[k] \\
\dot{\phi}[k+3] &\approx \dot{\phi}[k] + 3\Delta\mathrm{t}\ddot{\phi}[k] + 3\Delta\mathrm{t}^2\phi^{(3)}[k] + \Delta\mathrm{t}^3\phi^{(4)}[k] \\
\phi[k+4] &\approx \phi[k+3] + \Delta\mathrm{t}\dot{\phi}[k+3] \\
&\approx \phi[k] + 4\Delta\mathrm{t}\dot{\phi}[k] + 6\Delta\mathrm{t}^2\ddot{\phi}[k] + 4\Delta\mathrm{t}^3\phi^{(3)}[k] + \Delta\mathrm{t}^4\phi^{(4)}[k]
\end{aligned}$$

Repeating this procedure of substitution and expansion up to the order p, allows for the estimation of all the entries of the generalized signal. Then, vectorization of the derivative estimates such that it describes the generalized signal, yields the following matrix equation:

$$\tilde{\boldsymbol{\phi}}_p[k] \approx \boldsymbol{\mathcal{P}}_p \boldsymbol{\Delta} \tilde{\boldsymbol{\phi}}_d[k]$$

with $\boldsymbol{\mathcal{P}}_p$ and $\boldsymbol{\Delta}$ as defined in the previous appendix.

Note how the combination of the procedures as described in the current and the previous appendices can be used for direct mapping between (measurable) generalized signals with embedded history and generalized signals with embedded predictions:

$$\begin{aligned}
\tilde{\boldsymbol{\phi}}_p[k] &\approx \boldsymbol{\mathcal{P}}_p \boldsymbol{\Delta} \boldsymbol{\Delta}^{-1} \boldsymbol{\mathcal{P}}_p \mathbf{I}_{\pm} \tilde{\boldsymbol{\phi}}_h[k] \\
&= \boldsymbol{\mathcal{P}}_p \boldsymbol{\mathcal{P}}_p \mathbf{I}_{\pm} \tilde{\boldsymbol{\phi}}_h[k]
\end{aligned}$$

A-5 Log-likelihood

Consider a hidden- and measured-data joint-density maximum-likelihood function:

$$\mathrm{ML} = \prod_{k=0}^{N} \frac{1}{||\mathbf{Q}||\sqrt{2\pi}} e^{-\frac{1}{2}\boldsymbol{\varepsilon}_x^\top[k]\mathbf{Q}^{-1}\boldsymbol{\varepsilon}_x[k]} \frac{1}{||\mathbf{R}||\sqrt{2\pi}} e^{-\frac{1}{2}\boldsymbol{\varepsilon}_y^\top[k]\mathbf{R}^{-1}\boldsymbol{\varepsilon}_y[k]} \tag{A-6}$$

then, using the fact that any $\log(a \cdot b) = \log(a) + \log(b)$ brings:

$$\mathrm{LL} = \log(\mathrm{ML}) = \sum_{k=0}^{N} \Big(\log\left(\tfrac{1}{||\mathbf{Q}||}\right) + \log\left(\tfrac{1}{||\mathbf{R}||}\right) + 2\log\left(\tfrac{1}{\sqrt{2\pi}}\right) + \log(e^{-\frac{1}{2}\boldsymbol{\varepsilon}_x^\top[k]\mathbf{Q}^{-1}\boldsymbol{\varepsilon}_x[k]}) + \\ + \log(e^{-\frac{1}{2}\boldsymbol{\varepsilon}_y^\top[k]\mathbf{R}^{-1}\boldsymbol{\varepsilon}_y[k]})\Big) \tag{A-7}$$

then, using $\log(e^a) = a$ brings:

$$\mathrm{LL} = \sum_{k=0}^{N} \Big(\log\left(\tfrac{1}{||\mathbf{Q}||}\right) + \log\left(\tfrac{1}{||\mathbf{R}||}\right) + 2\log\left(\tfrac{1}{\sqrt{2\pi}}\right) - \frac{1}{2}\boldsymbol{\varepsilon}_x^\top[k]\mathbf{Q}^{-1}\boldsymbol{\varepsilon}_x[k] - \frac{1}{2}\boldsymbol{\varepsilon}_y^\top[k]\mathbf{R}^{-1}\boldsymbol{\varepsilon}_y[k]\Big) \tag{A-8}$$

and finally by simplifying $\log(\frac{1}{a}) = -\log(a)$ yields:

$$\mathrm{LL} = \sum_{k=0}^{N} \Big(-\log(||\mathbf{Q}||) - \log(||\mathbf{R}||) - \log(2\pi) - \frac{1}{2}\boldsymbol{\varepsilon}_x^\top[k]\mathbf{Q}^{-1}\boldsymbol{\varepsilon}_x[k] - \frac{1}{2}\boldsymbol{\varepsilon}_y^\top[k]\mathbf{R}^{-1}\boldsymbol{\varepsilon}_y[k]\Big) \tag{A-9}$$

Note that generally, including in the definition used in this thesis, the $-\log(2\pi)$ is omitted since it is static and thus does not influence the optima of the LL.

A-6 Shift matrices

For each of the three information embedding settings $i \in \{p,\ d,\ h\}$ as discussed in this thesis, the following holds:

$$\begin{aligned} \hat{\tilde{\boldsymbol{x}}}_i[k+1] &= \tilde{\mathbf{A}}_d \tilde{\boldsymbol{x}}_i[k] + \tilde{\mathbf{B}}_d \tilde{\boldsymbol{u}}_i[k] \\ \tilde{\boldsymbol{y}}[k] &= \tilde{\mathbf{C}}_d \tilde{\boldsymbol{x}}_i[k] + \tilde{\mathbf{D}}_d \tilde{\boldsymbol{u}}_i[k] \end{aligned} \tag{A-10}$$

and thus, in order for the internal consistency error:

$$\boldsymbol{\varepsilon}_{\tilde{\boldsymbol{x}}}[k] := \tilde{\boldsymbol{x}}'_i[k+1] - \tilde{\boldsymbol{x}}_i[k+1] \tag{A-11}$$

to be properly defined, the following must hold

$$\tilde{\boldsymbol{x}}'_i[k+1] = \boldsymbol{\mathcal{D}}_i \hat{\tilde{\boldsymbol{x}}}_i[k] \tag{A-12}$$

with $\boldsymbol{\mathcal{D}}_i$ a linear operator which brings the generalized signal as closely to its one step ahead prediction as possible.

Embedded predictions Firstly, for a generalized signal with embedded predictions it is easy to show that the original shift up operator native to DEM as described by Friston fulfils this property:

$$\underbrace{\begin{bmatrix} \boldsymbol{x}[k+1] \\ \boldsymbol{x}[k+2] \\ \vdots \\ \boldsymbol{x}[k+p] \\ \mathbf{0} \end{bmatrix}}_{\tilde{\boldsymbol{x}}'_p[k+1]} = \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{I} & & & \\ & \mathbf{0} & \mathbf{I} & & \\ & & \mathbf{0} & \ddots & \\ & & & \ddots & \mathbf{I} \\ & & & & \mathbf{0} \end{bmatrix}}_{\boldsymbol{\mathcal{D}}_p \equiv \boldsymbol{\mathcal{D}}} \underbrace{\begin{bmatrix} \boldsymbol{x}[k] \\ \boldsymbol{x}[k+1] \\ \boldsymbol{x}[k+2] \\ \vdots \\ \boldsymbol{x}[k+p] \end{bmatrix}}_{\tilde{\boldsymbol{x}}'_p[k]} \tag{A-13}$$

Embedded derivatives Secondly, for a generalized signal with embedded derivatives the original shift up operator native to DEM as described by Friston in fact yields a derivative step:

$$\underbrace{\begin{bmatrix} \dot{\boldsymbol{x}}[k] \\ \ddot{\boldsymbol{x}}[k] \\ \vdots \\ \boldsymbol{x}^{(p)}[k] \\ \mathbf{0} \end{bmatrix}}_{\dot{\tilde{\boldsymbol{x}}}'_d[k]} = \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{I} & & & \\ & \mathbf{0} & \mathbf{I} & & \\ & & \mathbf{0} & \ddots & \\ & & & \ddots & \mathbf{I} \\ & & & & \mathbf{0} \end{bmatrix}}_{\boldsymbol{\mathcal{D}}} \underbrace{\begin{bmatrix} \boldsymbol{x}[k] \\ \dot{\boldsymbol{x}}[k] \\ \ddot{\boldsymbol{x}}[k] \\ \vdots \\ \boldsymbol{x}^{(p)}[k] \end{bmatrix}}_{\tilde{\boldsymbol{x}}_d[k]} \tag{A-14}$$

and thus, following the discretization approach from the field of digital control, which states if $\dot{x}(t) = Ax(t)$ then $x[k+1] = e^{A\Delta \mathrm{t}}x[k]$ and thus:

$$\dot{\tilde{\boldsymbol{x}}}'_d[k] = \boldsymbol{\mathcal{D}} \tilde{\boldsymbol{x}}_d[k] \tag{A-15}$$

$$\tilde{\boldsymbol{x}}'_d[k+1] = \underbrace{e^{\boldsymbol{\mathcal{D}}\Delta \mathrm{t}}}_{\boldsymbol{\mathcal{D}}_d} \tilde{\boldsymbol{x}}_d[k] \tag{A-16}$$

Embedded history For a generalized signal with embedded history, the it appears that the best candidate for the shift operation is in fact a shift-down matrix:

$$\underbrace{\begin{bmatrix} \mathbf{0} \\ \boldsymbol{x}[k] \\ \boldsymbol{x}[k-1] \\ \vdots \\ \boldsymbol{x}[k-p] \\ \boldsymbol{x}[k-p+1] \end{bmatrix}}_{\tilde{\boldsymbol{x}}'_h[k+1]} = \underbrace{\begin{bmatrix} \mathbf{0} & & & & \\ \mathbf{I} & \mathbf{0} & & & \\ & \mathbf{I} & \mathbf{0} & & \\ & & \ddots & \ddots & \\ & & & \mathbf{I} & \mathbf{0} \end{bmatrix}}_{\boldsymbol{\mathcal{D}}_p \equiv \boldsymbol{\mathcal{D}}} \underbrace{\begin{bmatrix} \boldsymbol{x}[k] \\ \boldsymbol{x}[k-1] \\ \boldsymbol{x}[k-2] \\ \vdots \\ \boldsymbol{x}[k-p] \end{bmatrix}}_{\tilde{\boldsymbol{x}}'_h[k]} \tag{A-17}$$

Bibliography

[1] N.Wiener, *Collected Works vol.1.* MIT Press, 1976.

[2] A. Rosales-Lagarde, E. E. Rodriguez-Torres, B. A. Itzá-Ortiz, P. Miramontes, G. Vázquez-Tagle, J. C. Enciso-Alva, V. García-Muñoz, L. Cubero-Rego, J. E. Pineda-Sánchez, C. I. Martínez-Alcalá, and J. S. Lopez-Noguerola, "The color of noise and weak stationarity at the nrem to rem sleep transition in mild cognitive impaired subjects," *Frontiers in Psychology*, vol. 9, p. 1205, 2018.

[3] E. Santos, M. Khosravy, M. Lima, A. Cerqueira, C. Duque, and A. Yona, "High accuracy power quality evaluation under a colored noisy condition by filter bank esprit," *Electronics*, vol. 8, p. 1259, 11 2019.

[4] M. Dimian, O. Manu, and P. Andrei, "Influence of noise color on stochastic resonance in hysteretic systems," *Journal of Applied Physics*, vol. 111, no. 7, 2012.

[5] H. Trentelman, A. A. Stoorvogel, and M. Hautus, *Control Theory for Linear Systems.* Springer-Verlag London, 2001.

[6] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm (with discussion)," *J. Roy. Statist. Soc. Ser. B*, vol. 39.

[7] "An approach to time series smoothing and forecasting using the em algorithm," *Journal of Time Series Analysis*, vol. 3, no. 4, pp. 253–264, 1982.

[8] R. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Fluids Engineering*, vol. 82, pp. 35–45, 03 1960.

[9] K. Friston, J. Mattout, N. Trujillo-Barreto, J. Ashburner, and W. Penny, "Variational free energy and the laplace approximation," *Neuroimage*, vol. 43, no. 1, p. 220234, 2007.

[10] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2004.

[11] K. Friston, S. Klaas, L. Baojuan, and D. Jean, "Generalised filtering," *Mathematical Problems in Engineering*, vol. 2010, 06 2010.

[12] K. Friston, N. Trujillo-Barreto, and J. Daunizeau, “Dem: A variational treatment of dynamic systems,” pp. 849–85, 08 2008.

[13] K. Friston, J. Kilner, and L. Harrison, “A free energy principle for the brain,” *Journal of Physiology-Paris*, vol. 100, no. 1, pp. 70 – 87, 2006. Theoretical and Computational Neuroscience: Understanding Brain Functions.

[14] S. Grimbergen, C. van Hoof, P. M. Esfahani, and M. Wisse, “Active inference for state space models: A tutorial.”.

[15] A. A. Meera and M. Wisse, “Free energy principle based observer design for linear systems with coloured noise outperforms kalman filter.”.

[16] D. Cox and H. Miller, *The Theory of Stochastic Processes.* Chapman and Hall, 1977.

[17] K. Friston, “Statistical parametric mapping package.”

[18] P. Ioannou and J. Sun, *Robust Adaptive Control.* Dover Books on Electrical Engineering Series, Dover Publications, Incorporated, 2012.

[19] R. H. Middleton and G. C. Goodwin, *Digital control and estimation: a unified approach.* Prentice Hall Information and System Sciences Series, Prentice Hall, 1990.

[20] M. Green and J. B. Moore, “Persistence of excitation in linear systems,” *Systems & Control Letters*, vol. 7, no. 5, pp. 351 – 360, 1986.

[21] R. Bitmead, “Persistence of excitation conditions and the convergence of adaptive schemes,” *IEEE Transactions on Information Theory*, vol. 30, pp. 183–191, March 1984.