

ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE SCHOOL
OF LIFE SCIENCES



Master project in Life Sciences and Technology

**Fitting adaptive autoregressive models for
frequency analysis of non-stationary EEG data
via hybrid Kalman and Sequential Monte Carlo
filtering**

Carried out in the Neuroscience Statistics Research Lab
at Massachusetts Institute of Technology
Under the supervision of Professor Emery N. Brown

Done by

KRITON KONSTANTINIDIS

Under the direction of
Professor Jose del R. Millan
In the laboratory of NCBI

EPFL

Lausanne, June 20, 2019

To my grandparents, Stelios , Chrysoula, Myrofora and Kriton.

The dream of my grandfathers was to live long enough until i become a scientist. Pappou Stelios did not make it. Pappou this is especially for you.

Acknowledgments

First of all, I would like to express my gratitude to my supervisors, Professor Emery Brown and Professor Jose del R. Millan.

A special thanks to Mr. Andrew Song and Dr. Sourish Chakravarty for their help when things got stuck and hope was fading.

I am grateful to my amazing roommates Juan Martin Muñoz and Arjun Ramanathan who made sure i maintained my mental health after a long day at MIT.

Last but not least, I cannot express in words my love and appreciation for what my parents, Katerina and Giorgos, have offered to me all these years.

Abstract

The main target of this thesis is to develop a hybrid adaptive autoregressive model framework to fit non-stationary Electroencephalogram (EEG) data. In this setup, a state-space formulation is adopted. Hybridity arises from the fact that the state vector, which includes the autoregressive parameters, evolves in continuous time, while the observation equation is discrete, to account for the fact that the observations (EEG data) are discrete points in time. In other words, while the autoregressive model itself is discrete, its autoregressive parameters are continuous variables. Their adaptive estimation by means of Hybrid Kalman Filtering (HKF) and Expectation-Maximization (EM) algorithm can lead to robust, high-quality estimation of non-stationary EEG spectrograms. A variant of the model using non-Gaussian state noise and sequential Monte Carlo filtering on a self-organizing state-space model is also presented. Higher quality spectra are crucial in anesthesia related procedures where EEG frequency content can drastically help track different brain states. As shown in this thesis, hybrid autoregressive models yield smoother estimation of frequency spectra, outperforming other current purely discrete parametric methods as well as various non-parametric approaches. Examples of dynamic EEG data were taken from patients under gradually varying doses of the most commonly used anesthetics.

Keywords : Electroencephalogram (EEG) Spectral Analysis, Brain State Tracking, Anesthesia, Autoregressive models, Hybrid Kalman Filtering (HKF), Expectation Maximization (EM), Sequential Monte Carlo, Self-Organizing State-Space, Propofol, Ketamine

Contents

List of Figures	1
List of Tables	2
1 Introduction	3
1.1 Motivation	3
1.2 Non-stationary signal frequency analysis	3
1.3 Literature review on parametric adaptive methods for EEG analysis	4
1.4 Thesis Organization	5
2 The Model	6
2.1 State-Space Formulation	6
2.2 Hybrid Kalman Filtering (HKF)	7
2.3 Expectation Maximization (EM) Algorithm	8
2.4 Marginal Likelihood	9
2.5 Kalman Filter as a generative process	10
2.6 Model Selection	11
2.7 Goodness-of-fit	11
2.8 Spectral Estimation	12
2.9 Quantifying smoothness	14
2.10 Practical considerations	14
2.11 Rationale behind hybridity	15
2.12 Summary of the algorithm	16
3 Model Validation	17
3.1 Linear Frequency Transitions	17
3.2 Step Frequency Transitions	18
3.3 Expectation-Maximization	19
4 EEG Spectrograms under anesthetics	21
4.1 Propofol	21
4.2 Ketamine	24
4.3 Sevoflurane	29
5 Discussion on the hybrid Normal model	33
5.1 Comparative analysis	33
5.2 Over/Under Fitting	34
5.3 Smoothness differences in frequency tracking between discrete and hybrid models	39

6 Extending the hybrid model: Cauchy state noise	45
6.1 Self-Organizing State-Space Formulation	46
6.2 Bootstrap Hybrid Filtering with Sequential Importance Resampling	47
6.2.1 Discretization of the Ito state Integral using the Euler-Maruyama scheme	47
6.2.2 Sampling from a Cauchy distribution using the inverse transform method	47
6.2.3 Filtering	48
6.2.4 Likelihood of the model	49
6.3 Results on simulated Data	49
6.4 Propofol Spectrograms	50
6.5 Ketamine Spectrograms	51
6.6 Discussion	53
6.7 Improving estimates using a local implicit Markov Chain Monte Carlo algorithm	55
6.8 Improving estimates using a Metropolis-Hastings Sequential Markov Chain Monte Carlo algorithm	57
6.9 Particle Parallelization using virtual particle resampling	59
7 Conclusion & Future Research	62
A MATLAB Routines	63
A.1 Normal State Noise	63
A.2 Cauchy State Noise	65
B Combining Empirical Mode Decomposition with low order Hybrid Autoregressive Models for propofol spectral decomposition	66
B.1 MATLAB Routine	67
Bibliography	70

List of Figures

2.1	Summary of the algorithm	16
3.1	Simulated Noisy Sinusoidal: Linear frequency transitions	17
3.2	Simulated Noisy Sinusoidal: Linear frequency transitions Spectrogram	18
3.3	Simulated Noisy Sinusoidal: Step-wise frequency transitions	18
3.4	Simulated Noisy Sinusoidal: Step-wise frequency transitions Spectrogram	19
3.5	Simulated observations with ground truth \mathbf{Q} and R	19
4.1	Raw EEG under propofol	21
4.2	Non-parametric spectrograms of EEG under propofol	22
4.3	Discrete parametric spectrogram of EEG under propofol	23
4.4	Exponentially smoothed discrete spectrogram of EEG under propofol	23
4.5	Hybrid spectrogram of EEG under propofol	23
4.6	Raw EEG under ketamine 1	24
4.7	Non-parametric spectrograms of EEG under ketamine 1	24
4.8	Discrete parametric spectrogram of EEG under ketamine 1	25
4.9	Exponentially smoothed discrete spectrogram of EEG under ketamine 1	25
4.10	Hybrid spectrogram of EEG under ketamine 1	25
4.11	Raw EEG under ketamine 2	26
4.12	Non-parametric spectrograms of EEG under ketamine 2	26
4.13	Discrete parametric spectrogram of EEG under ketamine 2	27
4.14	Exponentially smoothed discrete spectrogram of EEG under ketamine 2	27
4.15	Hybrid spectrogram of EEG under ketamine 2	27
4.16	Autoregressive coefficients estimated using the hybrid filter	28
4.17	Raw EEG under sevoflurane with outlier rejection	29
4.18	Non-parametric spectrograms of EEG under sevoflurane with outlier rejection	29
4.19	Raw EEG under sevoflurane without outlier rejection	30
4.20	Non-parametric spectrograms of EEG under sevoflurane without outlier rejection	30
4.21	Discrete parametric spectrogram of EEG under sevoflurane	31
4.22	Exponentially smoothed discrete spectrogram of EEG under sevoflurane	31
4.23	Hybrid spectrogram of EEG under sevoflurane	31
5.1	Discrete parametric spectrogram of EEG under propofol, $\mathbf{Q} = 0.0001\mathbf{I}_p$	33
5.2	Spectrograms of order $p=10$	36
5.3	Spectrograms of order $p=11$	36
5.4	Spectrograms of order $p=16$	37
5.5	Akaike information criterion (AIC) for orders 2 until 20	38
5.6	Evolution of marginal likelihood in function of EM iterations for the chosen order $p=11$	38
5.7	Raw data for instantaneous frequency estimation	40
5.8	Instantaneous frequency estimation: Linear transitions	40
5.9	Instantaneous frequency estimation at 10 seconds	41

5.10	Instantaneous frequency estimation: Step transitions	42
5.11	Coefficients of discrete model for ketamine	43
5.12	Coefficients of hybrid model for ketamine	43
5.13	Coefficients of hybrid model for propofol	44
5.14	Coefficients of discrete model for propofol	44
6.1	Normal vs Cauchy distributions	45
6.2	Instantaneous frequency tracking of the Cauchy model	49
6.3	Hybrid Cauchy Spectrogram of EEG under propofol	50
6.4	Difference in power estimated by the Normal and Cauchy models for propofol .	50
6.5	AIC for Cauchy model	51
6.6	Hybrid Cauchy spectrogram of EEG under ketamine 1	52
6.7	Difference in power estimated by the Normal and Cauchy models for ketamine 1	52
6.8	Hybrid Cauchy spectrogram of EEG under ketamine 2	52
6.9	Difference in power estimated by the Normal and Cauchy models for ketamine 2	53
6.10	Temporal evolution of the scale of the Cauchy distribution of the autoregressive parameter	54
6.11	Hybrid Cauchy local MCMC Spectrogram of EEG under propofol	55
6.12	Difference in power estimated by the Cauchy SMC and Cauchy local MCMC models for propofol	55
6.13	Hybrid Cauchy MCMC Spectrogram of EEG under propofol	57
6.14	Difference in power estimated by SMC and SMCMC models for propofol	57
B.1	EMD Hybrid Spectrogram of EEG under propofol	67

List of Tables

5.1	Execution times	34
5.2	MSE for different orders and different \mathbf{Q}	35
5.3	NMSE for different orders and different \mathbf{Q}	35
5.4	Smoothness Metrics	42
6.1	Mean Square Errors of instantaneous frequency estimates for Gaussian and Cauchy models	50
6.2	Log-likelihood values for Gaussian and Cauchy hybrid models	53

Chapter 1

Introduction

1.1 Motivation

Despite the tremendous progress of modern medicine, a solid definition of anesthesia has not yet been provided. Current practices include various indices (e.g., Bispectral index) to determine anesthesia depth that are not very accurate and assume various generalities across populations, thus are not adapted to each individual and therefore cannot be reliable.

The utmost target of the Neuroscience Statistics Research Lab is to define anesthesia via a systems neuroscience approach. Such understanding can have profound effects in the way anesthesia is induced in patients undergoing surgery. Defining anesthesia in a more robust scientific manner will ensure the unconsciousness of patients during operations as well as their more efficient and less risky regain of consciousness. Deeper anesthesia levels can be achieved with smaller doses but more directed use of anesthetics, that in turn can drastically reduce side-effects of anesthesia procedures. These include post-operative memory deficits as well as sudden regain of consciousness during the operation (that sometimes surgeons do not realize, how terrifying!).

To achieve this goal, it is necessary to develop methods that are able to reliably track brain states in real-time in order to determine different anesthesia depths. One of the methods towards this direction is the frequency analysis of the electroencephalogram (EEG). This method calculates the frequency spectrum of the non-stationary EEG signal, which carries important information about the evolution of the brain state of the patient as time elapses.

1.2 Non-stationary signal frequency analysis

Non-stationary signal frequency analysis has been traditionally a thematic that attracted the attention of the scientific community. This is due to its enormous practical benefits that are a result of the fact that most signals with a temporal representation encountered in real-life applications are characterized by time-varying statistics.

The issue of non-stationarity was usually overcome by various assumptions about local stationarity properties over short time intervals and application of techniques suitable only for stationary signals. Even though such assumptions can have practical benefits and often provide satisfying performances, they are not always sufficient and can be a source of noise.

Current techniques used to estimate frequency spectra of non-stationary signals include various non-parametric methods, such as the multitaper approach developed by Thompson [1]. Non-parametric approaches suffer from a specific limitation: due to the principle of uncertainty, they cannot yield both high temporal and frequency resolution simultaneously. Indeed, they rely on the assumption of local stationarity. Improving time resolution by using a shorter, assumed stationary data segment, results in lower frequency resolution, and vice versa.

To overcome this issue, an alternative to non-parametric approaches is parametric methods. Such methods are usually based on time-varying linear predictive models, such as autoregressive (AR), moving average (MA) and autoregressive moving average (ARMA) models. The use of such models can lead to a higher resolution of time-varying frequency spectra, with the possibility to track relatively fast frequency changes by adaptive estimation of their parameters, that typical non-parametric methods may sometimes have trouble to follow.

Inspired by the work of Jones [2] and Harvey & Stock [3], where continuous, non-adaptive, AR models were fit to discrete samples of EEG recordings, this thesis adopts an intermediate between discrete and continuous approaches and develops algorithms for efficient spectral estimation of non-stationary signals. Finally, it explores their performance on different EEG datasets and compares it to that obtained by other widely used parametric and non-parametric methods.

1.3 Literature review on parametric adaptive methods for EEG analysis

Autoregressive models have been traditionally used to fit time-series for prediction and frequency analysis purposes. The first to use a state-space framework to fit adaptive autoregressive models for EEG analysis was [4]. A recursive least squares method was used to fit the model. [5] extends the approach to the multivariate case. A discrete random walk is assumed for the parameters which were then fit using a Kalman filter approach coupled with an exponential smoothing procedure to reduce the variance of the estimates. [6] proposes an alternative by adopting a Bayesian framework to fit autoregressive models to EEG data, coupled with a Kalman smoother, where future observations are used to reduce the variability of the estimates. [7] introduces an adaptive ARMA model applied to Event-Related Synchronization analysis. [8] proposes a Kalman smoother approach to fit many different low order autoregressive models to different frequency bands present in the same signal obtained by bandpass filtering. Fitting of adaptive autoregressive models to estimate time-varying spectra of EEG under the anesthetic propofol is done in [9]. In [5] and [8] the observation and process covariance matrices are constant and set before the filtering procedure. In [6], the components of the state-space model are adapted according to the Bayesian framework. In [9], observation variance and process noise covariance are estimated iteratively but the covariance matrix is constrained to be diagonal with all the diagonal terms being equal, thus leading to reduced adaptability. Finally, [10] uses an expectation-maximization approach to estimate the components of the state-space model in subsequent data windows and then runs a Kalman smoother to finely estimate the autoregressive coefficients.

All of the aforementioned approaches are purely discrete. Observations and states are treated as discrete variables. The work presented here attempts in a sense to combine all these approaches with the continuous work of [2] and [3] mentioned in the previous section.

1.4 Thesis Organization

The rest of the thesis is organized as follows.

In chapter 2 the state-space formulation of the proposed model, the equations of the HKF and an EM algorithm to adaptively estimate the continuous probability distribution of the autoregressive parameters, as more and more incoming EEG samples are received, are described in detail. Finally, an outline of the overall algorithm based on the model is provided.

Chapter 3, explores the performance of the algorithm on ground truth data as a model validation procedure.

Chapter 4 investigates the performance of the hybrid model in generating spectrograms from EEG data of patients under anesthetics and compares them to those obtained by other non-parametric and purely discrete parametric methods.

Chapter 5 discusses over/underfitting of the hybrid model and provides some insights to the effect of hybridity.

Chapter 6 suggests an alternative approach by extending the hybrid model using non-Gaussian state noise and compares it to the hybrid Gaussian model on simulated and real EEG data.

Finally, Chapter 7 briefly discusses some future steps that could be useful in the effort of even more accurate frequency tracking of non-stationary brain signals.

Chapter 2

The Model

2.1 State-Space Formulation

The model evolves according to the following equations:

$$\text{State Equation } \frac{da(t)}{dt} = w(t) \quad (1)$$

$$\text{Observation Equation } z_k = H_k a_k + v_k \quad (2)$$

$w(t) \sim \mathcal{N}(\mathbf{0}, Q(t))$ is a continuous multivariate Gaussian process with $\mathbf{0}$ mean and covariance $Q(t)$ (symmetric, positive-definite)

$v_k \sim \mathcal{N}(0, R(t))$ is a discrete scalar white Gaussian observation noise with 0 mean and variance $R(t)$

$a(t) = [a_1(t), a_2(t), \dots, a_p(t)]^T$ is the state vector with the continuous time-varying autoregressive coefficients and covariance matrix $P(t)$

$H_k = [z_{k-1}, \dots, z_{k-p}]$ is the observation matrix at time t_k comprising the p past discrete observations

$a_k = a(t_k)$ is the sampled state vector at time t_k

p : autoregressive model order

z_k : observation at time t_k

Let $Z = \{z_k\}_{k=1}^N$ denote the set of the observations.

Let $A = \{a_k\}_{k=1}^N$ denote the state vector at all times t_k of the observations z_k .

Let N denote the total number of observations in the data Z .

2.2 Hybrid Kalman Filtering (HKF)

With the above model, the HKF equations become:

$$\text{Initialization: } \mathbf{a}_{0|0} = \text{aryule}(Z, p) \quad (3), \quad \mathbf{P}_{0|0} = \mathbf{I}_p \quad (4)$$

$$\text{Prediction: } \frac{d\mathbf{a}(t)}{dt} = \mathbf{0} \Big|_{\mathbf{a}_{k-1|k-1}} \implies \int_{t_{k-1}}^{t_k} \frac{d\mathbf{a}(\tau)}{d\tau} d\tau = \mathbf{a}_{k|k-1} \quad (5)$$

$$\frac{d\mathbf{P}(t)}{dt} = \mathbf{Q} \Big|_{\mathbf{P}_{k-1|k-1}} \implies \int_{t_{k-1}}^{t_k} \frac{d\mathbf{P}(\tau)}{d\tau} d\tau = \mathbf{P}_{k|k-1} \quad (6)$$

In this design of the hybrid Kalman filter, the prediction equations do not involve update from measurements. The update from the measurements is used in the discrete part of the filter. Therefore, the Riccati variance equation that is usually required to be solved in the purely continuous version of the Kalman filter is avoided here. Indeed, in equation (6) the unknown P does not appear in any quadratic expression. Also, since the derivative of the state follows a random walk, there is not a linear term either, as it would be in the more general case where a transition matrix would not be zero.

$$\text{Filtering: } \mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + R)^{-1} \quad (7)$$

$$\mathbf{a}_{k|k} = \mathbf{a}_{k|k-1} + \mathbf{K}_k (z_k - \mathbf{H}_k \mathbf{a}_{k|k-1}) \quad (8)$$

$$\mathbf{P}_{k|k} = (\mathbf{I}_p - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1} \quad (9)$$

$$\mathbf{S}_k = \mathbf{P}_{k|k} (\mathbf{P}_{k|k} + \mathbf{Q})^{-1} \quad (10)$$

Note that the filter is time-varying since \mathbf{H}_k changes with time. As a result, \mathbf{P}_k and \mathbf{K}_k depend on time too. This means that this version of the filter will be slightly more computationally expensive than the time-invariant filter, but the cost will be low as the observations are scalar quantities. Also, it is important to note that $\mathbf{P}_{k|k}$ does not necessarily converge to a steady-state value as it is the case in the classic Kalman filter.

$$\text{Smoothing: } \mathbf{a}_{k|N} = \mathbf{a}_{k|k} + \mathbf{S}_k (\mathbf{a}_{k+1|N} - \mathbf{a}_{k+1|k}) \quad (11)$$

$$\mathbf{P}_{k|N} = \mathbf{P}_{k|k} + \mathbf{S}_k (\mathbf{P}_{k+1|N} - \mathbf{P}_{k+1|k}) \mathbf{S}_k^T \quad (12)$$

2.3 Expectation Maximization (EM) Algorithm

In order to estimate the process noise covariance \mathbf{Q} and the observation noise variance R in a maximum likelihood approach, the following EM algorithm can be applied.

Let $\Theta = \{\mathbf{Q}, R\}$ be the set of parameters whose maximum likelihood values are searched. The goal is to maximize the log-likelihood $\log[P(Z|\Theta)]$. Since access to the states is not provided, what is feasible is to calculate the expected complete log-likelihood $\mathcal{Q} = \mathcal{E}_{A|Z}[\log P([Z, A|\Theta])]$ in the E-Step of the algorithm and maximize it with respect to Θ in the M-Step in an iterative manner of alternating the 2 aforementioned steps. As proved in [11], improvements in \mathcal{Q} result in improvements in the log-likelihood $\log[P(Z|\Theta)]$.

Taking into account the Markov property of the model and excluding the initial state since the state vector is initialized using the Yule-Walker equations and for simplicity the state covariance is initialized as the identity matrix of order p , one obtains:

$$\mathcal{Q} = \mathcal{E}_{A|Z}[\log[P(Z, A|\Theta)] = \mathcal{E}_{A|Z}[\log\left[\prod_{k=2}^N P(\mathbf{a}_k|\mathbf{a}_{k-1}) \prod_{k=1}^N P(z_k|\mathbf{a}_k, \mathbf{H}_k)\right]] \quad (13)$$

Developing (13), using (1) and (2):

$$\begin{aligned} \mathcal{Q} &= \mathcal{E}_{A|Z}\left[\sum_{k=2}^N \log[P(\mathbf{a}_k|\mathbf{a}_{k-1})] + \sum_{k=1}^N \log[P(z_k|\mathbf{a}_k, \mathbf{H}_k)]\right] = \\ &= \mathcal{E}_{A|Z}\left[\sum_{k=2}^N \log[(2\pi)^{-\frac{p}{2}} |\mathbf{Q}|^{-\frac{1}{2}} \exp\{-\frac{1}{2}(\mathbf{a}_k - \mathbf{a}_{k-1})^T \mathbf{Q}^{-1} (\mathbf{a}_k - \mathbf{a}_{k-1})\}] + \right. \\ &\quad \left. + \sum_{k=1}^N \log[(\frac{1}{2\pi R})^{\frac{1}{2}} \exp\{-\frac{1}{2R}(z_k - \mathbf{H}_k \mathbf{a}_k)^2\}]\right] \quad (14) \end{aligned}$$

Note that the state transition matrix between the observations has been approximated by \mathbf{I}_p since the derivatives of the coefficients follow a random walk between the observations and thus the equivalent model in discrete time is characterized by an identity state transition matrix.

Developing (14) using the properties of the \log function and inserting the expectation inside the sums:

$$\begin{aligned} \mathcal{Q} &= -\frac{(N-1)p}{2} \log(2\pi) - \frac{N-1}{2} \log(|\mathbf{Q}|) - \frac{N}{2} \log(2\pi R) - \\ &\quad - \frac{1}{2} \sum_{k=2}^N \mathcal{E}_{A|Z}[(\mathbf{a}_k - \mathbf{a}_{k-1})^T \mathbf{Q}^{-1} (\mathbf{a}_k - \mathbf{a}_{k-1})] - \frac{1}{2R} \sum_{k=1}^N [z_k^2 - 2z_k \mathbf{H}_k \mathcal{E}_{A|Z}[\mathbf{a}_k] + \\ &\quad + \mathbf{H}_k \mathcal{E}_{A|Z}[\mathbf{a}_k \mathbf{a}_k^T] \mathbf{H}_k^T] \quad (15) \end{aligned}$$

To finalize the expression for \mathcal{Q} , the following relations will be used:

$$\mathcal{E}_{A|Z}[\mathbf{a}_k] = \mathcal{E}[\mathbf{a}_k|Z] = \mathbf{a}_{k|N} \quad (16)$$

$$\mathcal{E}_{A|Z}[\mathbf{a}_k \mathbf{a}_k^T] = \mathcal{E}[\mathbf{a}_k \mathbf{a}_k^T|Z] = \mathbf{P}_{k|N} + \mathbf{a}_{k|N} \mathbf{a}_{k|N}^T \quad (17)$$

(16) and (17) are obtained using (11) and (12).

Last required expression is the following:

$$\mathcal{E}_{A|Z}[\mathbf{a}_k \mathbf{a}_{k-1}^T] = \mathcal{E}[\mathbf{a}_k \mathbf{a}_{k-1}^T | Z] = \mathbf{S}_{k-1} \mathbf{P}_{k|N} + \mathbf{a}_{k|N} \mathbf{a}_{k-1|N}^T \quad (18)$$

(18) can be obtained using (10),(11) and (12).

Now the final expression of \mathcal{Q} is obtained from (15), using (16),(17),(18) and expanding the expectation terms:

$$\begin{aligned} \mathcal{Q} = & -\frac{(N-1)p}{2} \log(2\pi) - \frac{N-1}{2} \log(|\mathbf{Q}|) - \frac{N}{2} \log(2\pi R) - \frac{1}{2R} \sum_{k=1}^N [z_k^2 - 2z_k \mathbf{H}_k \mathbf{a}_{k|N} + \\ & + \mathbf{H}_k [\mathbf{P}_{k|N} + \mathbf{a}_{k|N} \mathbf{a}_{k|N}^T] \mathbf{H}_k^T] - \frac{1}{2} \sum_{k=2}^N \text{tr}[\mathbf{Q}^{-1} [\mathbf{P}_{k|N} + \mathbf{a}_{k|N} \mathbf{a}_{k|N}^T - 2(\mathbf{S}_{k-1} \mathbf{P}_{k|N} + \mathbf{a}_{k|N} \mathbf{a}_{k-1|N}^T) + \\ & + \mathbf{P}_{k-1|N} + \mathbf{a}_{k-1|N} \mathbf{a}_{k-1|N}^T]] \end{aligned} \quad (19)$$

The M-step can now be calculated. To do so, the partial derivatives of \mathcal{Q} with respect to \mathbf{Q}^{-1} and R will be calculated and set to zero. This yields:

$$\frac{\partial \mathcal{Q}}{\partial \mathbf{Q}^{-1}} = 0 \implies \mathbf{Q}_{ML} = \frac{1}{N-1} \left[\sum_{k=2}^N [\mathbf{P}_{k|N} + \mathbf{a}_{k|N} \mathbf{a}_{k|N}^T - \mathbf{S}_{k-1} \mathbf{P}_{k|N} - \mathbf{a}_{k|N} \mathbf{a}_{k-1|N}^T] \right] \quad (20)$$

$$\frac{\partial \mathcal{Q}}{\partial R} = 0 \implies R_{ML} = \frac{1}{N} \sum_{k=1}^N [z_k^2 - 2z_k \mathbf{H}_k \mathbf{a}_{k|N} + \mathbf{H}_k [\mathbf{P}_{k|N} + \mathbf{a}_{k|N} \mathbf{a}_{k|N}^T] \mathbf{H}_k^T] \quad (21)$$

Overall, the EM algorithm for the proposed model is given below:

Initialization: Initialize $\mathbf{Q}_{start} = \mathbf{I}_p$, $R_{start} = R(0)$

E-Step: Calculate \mathcal{Q} using (19).

M-Step: Find maximum likelihood \mathbf{Q}_{ML} and R_{ML} using (20) and (21) respectively.

The above 2 steps are repeated until convergence.

2.4 Marginal Likelihood

The next step is to calculate the marginal likelihood \mathcal{L} that the observed data were generated by the model. To do so, integration over the values of the hidden state has to take place. \mathcal{L} will be firstly calculated here and then used in the model selection procedure of the next section. It will also be used to monitor convergence of the EM algorithm described in the previous section.

Using the chain rule from probability theory:

$$\mathcal{L} = p(Z) = \prod_{k=1}^N p(z_k | z_{k-1}, \dots, z_0)$$

and using once again the markovianess of the model while integrating over the states,

$$\begin{aligned} \mathcal{L} &= \prod_{k=1}^N \int p(z_k | \mathbf{a}_k) p(\mathbf{a}_k | z_{k-1}, \dots, z_0) d\mathbf{a}_k = \\ &= \prod_{k=1}^N \int \mathcal{N}(z_k; \mathbf{H}_k \mathbf{a}_k, R) \mathcal{N}(\mathbf{a}_k; \mathbf{a}_{k|k-1}, \mathbf{P}_{k|k-1}) d\mathbf{a}_k = \\ &= \prod_{k=1}^N \mathcal{N}\left(z_k; \mathbf{H}_k \mathbf{a}_{k|k-1}, \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + R\right) \quad (22) \end{aligned}$$

For convenience, the log of \mathcal{L} will be used. It is calculated as follows:

$$\begin{aligned} \log \mathcal{L} &= \log \prod_{k=1}^N \mathcal{N}\left(z_k; \mathbf{H}_k \mathbf{a}_{k|k-1}, \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + R\right) = \\ &= \sum_{k=1}^N \log \mathcal{N}\left(z_k; \mathbf{H}_k \mathbf{a}_{k|k-1}, \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + R\right) = \\ &= \sum_{k=1}^N \log \left[\left(\frac{1}{2\pi(R+\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T)} \right)^{\frac{1}{2}} \exp \left\{ -\frac{1}{2(R+\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T)} (z_k - \mathbf{H}_k \mathbf{a}_{k|k-1})^2 \right\} \right] = \\ &= -\frac{N}{2} \log(2\pi) - \frac{1}{2} \sum_{k=1}^N \log(R + \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T) - \frac{1}{2} \sum_{k=1}^N \frac{(z_k - \mathbf{H}_k \mathbf{a}_{k|k-1})^2}{R + \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T} \quad (23) \end{aligned}$$

Convergence can be monitored by tracking the evolution of $\log \mathcal{L}$ in function of the number of EM iterations using (23). When the change in \mathcal{L} between two consecutive iterations is smaller than a predefined tolerance, the algorithm is considered to have converged. Note that convergence of the algorithm is guaranteed, as shown in [12]. Also, note that the second derivative test was omitted since the derived equations are special cases of the general equations of EM for similar state space models as described in [13] and negativity of the second derivatives evaluated at the maximum likelihood estimates has been well studied. A further indication that the algorithm is working properly is the non-decreasing evolution of $\log \mathcal{L}$ that was observed in every case of this thesis that EM algorithm was used.

2.5 Kalman Filter as a generative process

The filter described in section 2.2 can be used as a generative process to generate new observations. This will be very useful for validation of the EM algorithm, where observations generated by the model as ground truth data with predetermined, known parameters $\Theta = \{\mathbf{Q}, R\}$ will be fed back to the model to test if the EM algorithm can correctly estimate Θ .

In order to generate new observations, the following procedure will be followed :

1. Initialize p initial states as $\mathbf{a}_i = \mathcal{N}(\mathbf{a}_{i|i}, \mathbf{P}_{i|i})$, where $\mathbf{a}_{i|i} = \mathbf{0}$ and $\mathbf{P}_{i|i} = \mathbf{I}_p$.
2. Generate p initial random observations $[z_0, z_1, \dots, z_p]$. It is impossible to generate observations $z_i, 0 \leq i \leq p$ from the model, as the observation matrix depends on the past p observations. The first p observations were randomly generated from the standard distribution.
3. For $k = p + 1, p + 2, p + 3, \dots, K$

Sample the next hidden state \mathbf{a}_k from the transition model $p(\mathbf{a}_k | \mathbf{a}_{k-1}) = \mathcal{N}(\mathbf{a}_{k-1}, \mathbf{Q})$.

Sample an observation z_k from the observation model $p(z_k | \mathbf{a}_k) = \mathcal{N}(\mathbf{H}_k \mathbf{a}_k, R)$.

At the end of this procedure, K observations from the model have been generated.

2.6 Model Selection

Having calculated the marginal log-likelihood, model selection can now proceed to determine the optimal autoregressive order of the model. The most widely used model selection criterion is the Akaike Information Criterion (*AIC*), defined as follows.

$AIC(p) = 2p - 2\mathcal{L}_{ML}$, where p is the order of the autoregressive model and \mathcal{L}_{ML} is the maximized marginal log-likelihood for the model, obtained from the EM algorithm and the marginalization procedure described in the previous section.

Alternatively, there is the option to use the Bayesian Information Criterion (*BIC*), defined as $BIC(p) = \ln(N)p - 2\mathcal{L}_{ML}$, where N is the total number of observations. BIC places a higher penalty on the model order than AIC and is valid when the number of samples is much larger than the number of the parameters. Since model selection will take place in short EEG data intervals due to the computational complexity of the EM algorithm (and thus the number of samples will not be very high), AIC seems to be a better option.

In what follows, the model that yields the lowest AIC value will be selected.

2.7 Goodness-of-fit

The marginal likelihood function was used as a metric for model selection and as a convergence criterion for the EM algorithm. Once the autoregressive order is selected and the model is fit to the data, metrics can be used to assess the goodness-of-fit between the model and the data.

Let z_k denote the observation at time t_k , \hat{z}_k denote the predicted observation at time t_k , $\|\cdot\|$ denote the 2-norm of a vector and \bar{Z} denote the mean of the observations Z . The three different metrics that will be used are defined as follows.

1. *Mean Square Error (MSE)* : The *MSE* between the raw data and the model is defined as

$$MSE = \frac{\sum_{k=1}^N \|\hat{z}_k - z_k\|^2}{N}$$

2. *Normalized Mean Square Error (NMSE)*: The *NMSE* between the raw data and the model is defined as

$$NMSE = 1 - \frac{\sum_{k=1}^N \|\hat{z}_k - z_k\|^2}{\sum_{k=1}^N \|z_k - \bar{Z}\|^2}$$

NMSE values vary between $-\infty$, indicating bad fit and 1, indicating perfect fit.

3. *Normalized Root Mean Square Error (NRMSE)*: The *NRMSE* between the raw data and the model is defined as

$$NRMSE = 1 - \frac{\sum_{k=1}^N \|\hat{z}_k - z_k\|}{\sum_{k=1}^N \|z_k - \bar{Z}\|}$$

NMRSE values vary between $-\infty$, indicating bad fit and 1, indicating perfect fit.

Those metrics can be used in conjunction with visual inspection of the plot of raw data overlapped with the predicted one to assess goodness-of-fit. Also, they can be used to assess over/under fitting of the model as it will be discussed in the next chapters of this thesis.

2.8 Spectral Estimation

Having set up the building blocks, now it is time to put them all together and build an efficient algorithm for spectral analysis of non-stationary EEG signals.

The algorithm starts by running the EM routine for different autoregressive model orders on an initial minibatch consisting of data of 10 seconds. This procedure yields maximum likelihood estimates for $\Theta = \{\mathbf{Q}, \mathbf{R}\}$. Then, by evaluating AIC using the marginal likelihood obtained from the EM routine, the best model order p is selected. The hybrid filter is run forward for the next 10 minutes. When 10 minutes of data have elapsed, the EM routine is rerun on a 10 second minibatch using as initial conditions the results of the previous EM run, in order to update $\Theta = \{\mathbf{Q}, \mathbf{R}\}$ by making use of previous information. Essentially, a prior is used here to account for the fact that there has to be some relationship between the distribution of the parameters at time t_k and at time $t_k - 10\text{min}$. The hybrid filter is then run again forward for 10 minutes. This alternating procedure of updating the parameters of the model using 10 second minibatches and running the hybrid filter forward for the next 10 minutes of data is repeated until the end of the EEG data. Ten minutes is an indicative number here and it can be changed depending on the computational resources one wishes to consume. It is considered a reasonable time in which the distribution of the parameters is expected to have changed. Finally, at the end of the data, a Kalman smoother is run backwards to smooth previous estimates once all the information from the observations is available.

At the end of the steps described in the above paragraph, the coefficients obtained by hybrid Kalman filtering and smoothing whose parameters have been adaptively estimated using online EM can be used to calculate the spectrogram.

Note that in the case where a faster spectrogram calculation is desired, EM and model selection can be only run once in an initial data minibatch. In this case, even though Θ is not adaptively estimated, it was empirically found that the obtained spectrograms are still characterized by high resolution.

An exponential smoothing procedure can be applied at this stage, based on [5]. Exponential smoothing is suggested for the purely discrete case to reduce the variance of the estimates. It is not necessary in the hybrid case but it will briefly be described here since maybe the user wishes to have even smoother parameter estimates.

Let \mathbf{a}_k denote the vector of the coefficients obtained at the end of the algorithm and $\check{\mathbf{a}}$ denote the exponentially smoothed vector at time t_k . $\check{\mathbf{a}}$ can be obtained as follows:

$$\check{\mathbf{a}}_k = (1 - c_k)\check{\mathbf{a}}_{k-1} + c_k \mathbf{a}_k, \quad \text{where } c_k = C \frac{(\mathbf{a}_k - \check{\mathbf{a}}_{k-1})^2}{1 + C(\mathbf{a}_k - \check{\mathbf{a}}_{k-1})^2} \quad (24)$$

In the rest of this thesis, exponential smoothing will not be used in any case where the hybrid filter is applied. It will only be used in cases where purely discrete Kalman filter is applied to enhance smoothness for comparison purposes.

Finally, the spectrogram is calculated. An instantaneous estimate of the spectral density can be given as:

$$S(f, t) = \frac{R}{|1 - \sum_{k=1}^p a_k(t) e^{-i2\pi k \frac{f}{f_s}}|^2}, \quad 0 \leq f \leq \frac{f_s}{2} \quad (25)$$

Overall, the spectral density estimate is calculated at all times when there is an observation, to produce the spectrogram corresponding to the temporal evolution of the frequency content of the input data. Note that (25) is a continuous function of the frequency f and time t and can be calculated at any frequency ranging from 0 up to $\frac{f_s}{2}$, where f_s is the sampling frequency of the raw data. As it can be evaluated at infinitely many time-frequency points, it can yield very high temporal and frequency resolution of the power spectral density.

Finally, note that in the case of online use of the model, smoothing can be ignored, as it was found empirically that no significant effects take place in the frequency tracking when only filtering is applied.

2.9 Quantifying smoothness

As described before, the aim of this dissertation is to yield high-resolution smooth spectrograms. Therefore, it is necessary to define a measure to quantify the relative smoothness of the autoregressive coefficients. This way, comparisons between hybrid and discrete models will be more accurate. A typical convention to measure the smoothness of a curve between two points α and β is the integral of the square of the second derivative of the curve. More precisely, let $f(t)$ denote a 2-dimensional curve and $S(f(t))$ denote the smoothness metric. Then,

$$S(f(t)) = \int_{\alpha}^{\beta} \left[\frac{d^2 f(t)}{dt^2} \right]^2 dt. \quad (26)$$

The above definition will be approximated by calculating the second differences between the estimates of the coefficients at each time point and then numerically integrating their square using the trapezoid rule to calculate an approximation of the integral. The final value that will be reported is the average of the integrals for each coefficient $a(t)$. The lower the value of the defined metric is, the smoother the time-series $f(t)$ is across time.

2.10 Practical considerations

During the practical implementation of the algorithm described above, some practical issues occurred that led to slight modifications. They are described below:

1. Parameter Identifiability: The EM routine attempts to simultaneously estimate \mathbf{Q} and R . The issue here is that the variance in the temporal evolution of the coefficients can arise both from a large covariance matrix, that is high variability in the distribution that the parameters follow, or from a high variance in the observations. In other words, the EM routine may lead to an absorption of the variability by one of the two parameters to be estimated. As a result, it may yield a very high value for one of the two while minimizing the other to a lower value than expected. To circumvent this issue, EM routine is run while one of the parameters is fixed to a reasonable value. R was chosen to be kept constant, as \mathbf{Q} is a more complex entity and thus more difficult to be estimated without EM. R values that work were empirically found to vary between 0.05 and 1 for different datasets. Therefore, depending on the type of anesthetic used, the value of R is manually adjusted accordingly. For ketamine a value 0.15 was used while for propofol 0.05 was found to be optimal.

2. Computational Complexity: The routine for maximum likelihood estimation of \mathbf{Q} is quite computationally expensive due to both the large number of operations that need to take place at every iteration, as well as the number of total iterations until convergence. That number was empirically found to be between 15 and 50 iterations. Even 15 iterations can take quite some time for a batch of data of 5 to 10 seconds. As a result, the routine will not be run in consecutive windows, as it would be ideal, but instead, it will be run every 10 minutes of data. Even though this approach may be slightly suboptimal than updating \mathbf{Q} in consecutive windows, it is much simpler computationally and accelerates the overall algorithm substantially without significant performance deterioration.

3. Outlier Rejection and Data Normalization: It was empirically found that normalizing the data to the range $[-1, 1]$ greatly enhances filter stability. This can be achieved by dividing each observation z_i by $\max(|Z|)$.

Even though this normalization greatly stabilizes the filter, it can cause some problems. Indeed, when there are aberrant values in the raw data as a result of an artifact, the other normalized observations will be very close to 0. To correct this issue, the following rejection rule will be followed:

If $|z_i| > \text{mean}(Z) + 5\text{std}(Z)$, then z_i is removed. This rule may sometimes lead to rejection of some actual observations, in case of intense neuronal spiking, but their removal is not expected to affect the frequency content significantly while at the same time it greatly enhances the performance of the filter. Indeed, avoiding the accumulation of observations around 0 is an improvement, as such a case is not physiologically plausible. Outlier rejection can be enabled or not, depending on the existence of aberrant values. Their detection can be done with visual inspection of the raw EEG data. Results that follow in Chapter 4 indicate that outlier rejection is a good strategy, stabilizing the filter while not affecting the produced spectrogram.

2.11 Rationale behind hybridity

Having described the mathematical and algorithmic aspects of the hybrid model, this section is devoted to a more conceptual description and explanation of the motivation and the intuition that led to the idea of hybridity.

An inherent issue of the state-space formulation of an autoregressive process is that the observation matrix comprises past observations. Thus, if those observations are highly variant, the Kalman Gain will also be variant as a result of its dependence on the observation matrix, leading to instabilities and noise. The key idea is to allow the quick adaptation of the autoregressive parameters when a new observation is available, while simultaneously reduce the variability that arises from that flexibility in the temporal evolution of the parameters and the aforementioned fact in the previous sentence. Thus, while in the purely discrete case the parameters "jump" from one value to another, being updated after each new incoming observation, in the hybrid case a different framework is adopted which favours smoothness.

Indeed, in this setup, the parameters are continuous variables. As there is no new information in the time intervals between the observations, the parameters are modeled so that they follow a continuous random walk. Using this mathematical trick, the model allows for the definition of the parameters not only at the time points of the observations, but also in the intermediate intervals, essentially along the whole real line. They are allowed to evolve randomly while there is no new observation available and they are updated every time a new one arrives. As a result, the temporal evolution of the parameters is more smooth, while at the same time, the smoothness constraint that is imposed does not prevent the model from identifying abrupt changes in the frequency content of the analyzed signals.

2.12 Summary of the algorithm

The overall algorithm is summarized in the following schematic.

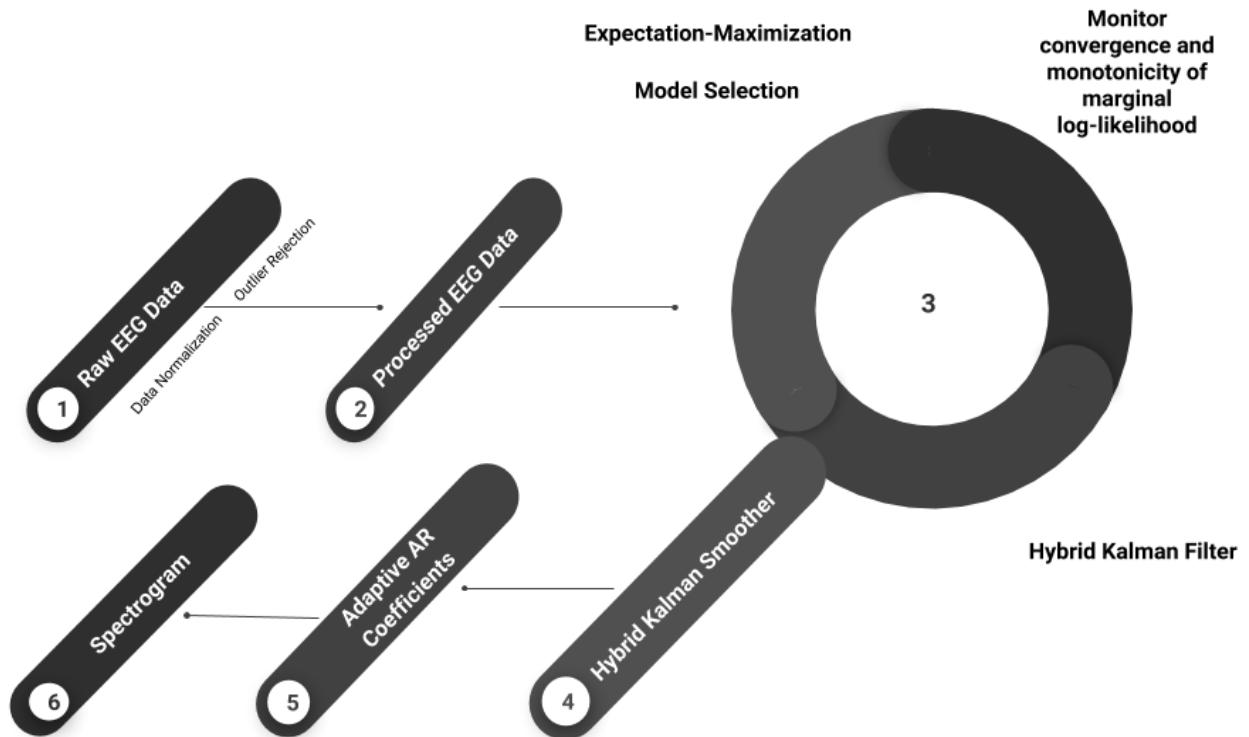


Figure 2.1: Summary of the algorithm

Chapter 3

Model Validation

The first thing that has to be done after a model has been developed is to be tested on ground truth simulated data. To do so, a simulated noisy sinusoidal with amplitude and frequency modulation was created.

The following signal was generated: $z_k = A_k \sin(\omega_k \frac{k}{f_s}) + v_k$, where A_k is the modulated amplitude, ω_k is the time-varying frequency and v_k is added white Gaussian noise with zero mean and unit variance. The duration of the signal is $N=60$ seconds and the sampling frequency is $f_s=250$ Hz (typical EEG sampling frequency).

2 different scenarios were tested. In the first, the transitions between the frequencies were linear and in the second the transitions were step functions. In both cases the model, is expected to be able to follow the frequency changes, irrespective of the amplitude modulation and the added noise.

3.1 Linear Frequency Transitions

Here the frequency component evolves as described by the following equation:

$$\omega_k = \begin{cases} 30+4\frac{k}{f_s}, & \text{if } 0 \leq \frac{k}{f_s} < 10 \\ 90-2\frac{k}{f_s}, & \text{if } 10 \leq \frac{k}{f_s} < 20 \\ -10+3\frac{k}{f_s}, & \text{if } 20 \leq \frac{k}{f_s} < 30 \\ 140-2\frac{k}{f_s}, & \text{if } 30 \leq \frac{k}{f_s} < 40 \\ -20+2\frac{k}{f_s}, & \text{if } 40 \leq \frac{k}{f_s} < N \end{cases} \quad (A)$$

while the amplitude varies as: $A_k = 1 + k/N$, $0 \leq k \leq N$. Those settings produced the following raw simulated data:

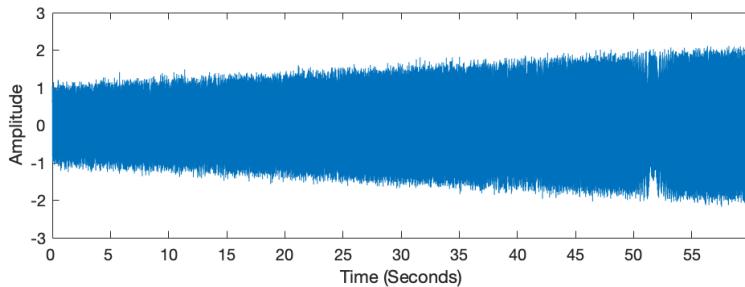


Figure 3.1: Simulated Noisy Sinusoidal: Linear frequency transitions

Applying the hybrid Kalman Filter using an autoregressive model of order $p = 4$, a covariance matrix $\mathbf{Q} = 0.01\mathbf{I}_p$ and the ground truth observation noise variance $R = 1$, produced the

following spectrogram.

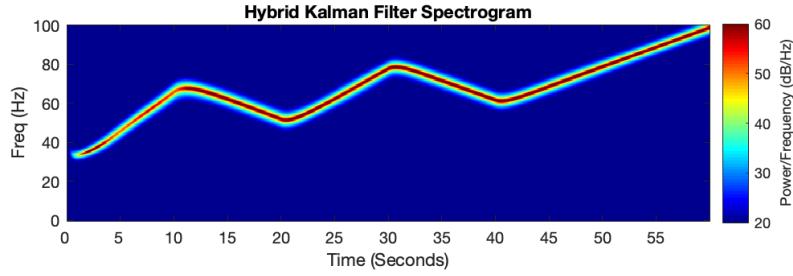


Figure 3.2: Simulated Noisy Sinusoidal: Linear frequency transitions Spectrogram

It becomes apparent that the obtained spectrogram shows clearly the frequency evolution that is described by the piece-wise function (A). The conclusion can be made that the model was able to follow the frequency changes, while its performance was not hindered by the time-varying amplitude or the added white noise.

3.2 Step Frequency Transitions

In this case the frequency component involves "jumps", as described by the following equation.

$$\omega_k = \begin{cases} 30, & \text{if } 0 \leq \frac{k}{f_s} < 10 \\ 70, & \text{if } 10 \leq \frac{k}{f_s} < 20 \\ 50, & \text{if } 20 \leq \frac{k}{f_s} < 30 \\ 80, & \text{if } 30 \leq \frac{k}{f_s} < 40 \\ 60, & \text{if } 40 \leq \frac{k}{f_s} < N \end{cases} \quad (B)$$

The amplitude follows the same equation as in the previous case. Those settings produced the following raw simulated data:

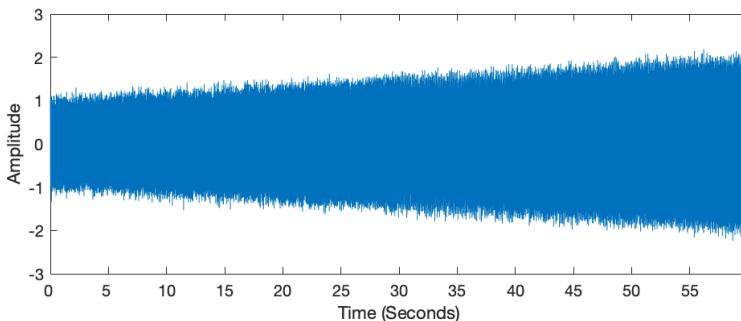


Figure 3.3: Simulated Noisy Sinusoidal: Step-wise frequency transitions

Applying the hybrid Kalman Filter using an autoregressive model of order $p = 4$, a covariance matrix $\mathbf{Q} = 0.01\mathbf{I}_p$ and the ground truth observation noise variance $R = 1$, produced the following spectrogram.

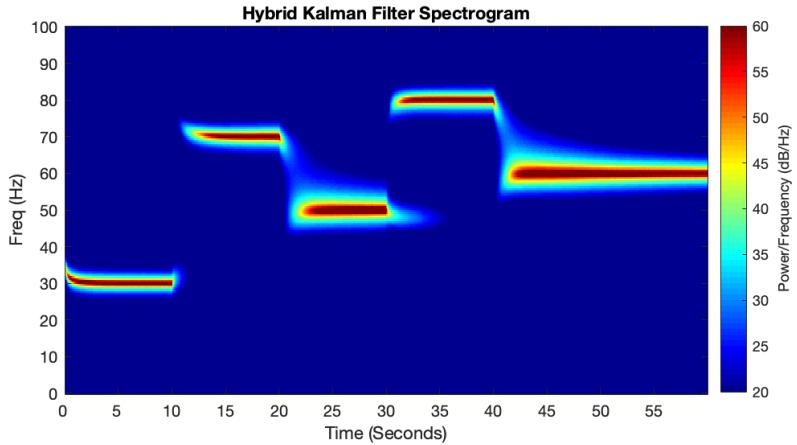


Figure 3.4: Simulated Noisy Sinusoidal: Step-wise frequency transitions Spectrogram

Once again, the model is able to quickly track frequency changes as described by the equation (B). Note that a higher covariance matrix coupled with a higher order autoregressive model could provide an even better result, but for the sake of low complexity and given the fact that the obtained spectrogram looks quite accurate, the same values for \mathbf{Q} and p as in the previous case of linear frequency transitions were used. Also, the parameters \mathbf{Q} and p were chosen manually as there is not ground truth for \mathbf{Q} and it is known that for a single frequency component an AR model of order 2 is enough. Here a model of order $p=4$ was used as it has been empirically shown that for noisy signals with a single frequency component, order 4 performs better.

3.3 Expectation-Maximization

In order to validate the EM algorithm, synthetic observations of 4 seconds were generated by applying the procedure described in section 2.5 for a given order $p=10$, $\mathbf{Q} = 0.001\mathbf{I}_p$ and $R = 0.5$ (Fig 3.5). The observations were then fed back to the EM algorithm.

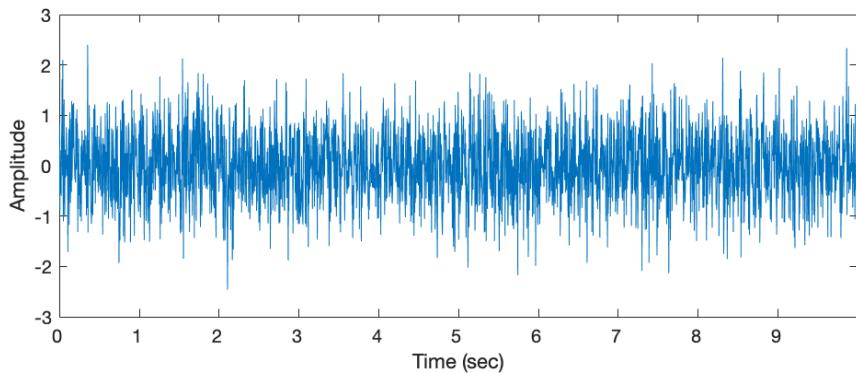


Figure 3.5: Simulated observations with ground truth \mathbf{Q} and R

The EM routine, using a tolerance of 0.001, converged to the following estimation of \mathbf{Q} after 21 iterations. Initial covariance matrix was set to $\mathbf{Q}_{start} = \mathbf{I}_p$.

Covariance matrix estimate of EM algorithm

$$\mathbf{Q}_{EM} = \begin{bmatrix} 0.0011 & -0.0006 & -0.0003 & 0.0006 & -0.0002 & -0.0005 & 0.0000 & -0.0001 & -0.0006 & 0.0004 \\ 0.0003 & 0.0009 & -0.0003 & 0.0003 & -0.000 & -0.0006 & -0.0001 & -0.0001 & -0.0001 & -0.0007 \\ -0.0002 & 0.0005 & 0.0013 & -0.0010 & 0.0004 & 0.0004 & -0.0001 & 0.0007 & 0.0006 & -0.0011 \\ -0.0002 & 0.0005 & -0.0001 & 0.0007 & 0.0013 & 0.0004 & 0.0010 & 0.0009 & 0.0003 & 0.0006 \\ 0.0003 & -0.0007 & -0.0008 & 0.0011 & 0.0020 & -0.0006 & 0.0013 & -0.0005 & -0.0015 & 0.0023 \\ 0.0006 & 0.0009 & -0.0001 & -0.0015 & 0.0007 & 0.0017 & 0.0003 & 0.0010 & 0.0007 & -0.0000 \\ 0.0000 & -0.0009 & -0.0005 & 0.0017 & 0.0001 & -0.0007 & 0.0016 & -0.0010 & -0.0012 & 0.0015 \\ 0.0002 & -0.0004 & -0.0001 & 0.0003 & -0.0005 & -0.0001 & 0.0002 & 0.0014 & -0.0002 & 0.0002 \\ -0.0000 & 0.0012 & 0.0005 & -0.0005 & -0.0011 & 0.0005 & -0.0003 & -0.0002 & 0.0018 & -0.0021 \\ 0.0001 & -0.0004 & -0.0009 & 0.0019 & 0.0008 & -0.0005 & 0.0013 & -0.0006 & -0.0002 & 0.0011 \end{bmatrix}$$

Overall, the EM algorithm is able to converge towards the true covariance matrix. Diagonal terms differ from their true values in the range of ± 0.001 , while some small off-diagonal terms is a result of numerical error and can be considered negligible. Slight deviance from the ground truth estimate of the covariance matrix also arises from the fact of randomly sampling the Gaussian distributions for a duration of only 10 seconds. A larger dataset would allow the algorithm to converge even closer to the true estimate, but 10 seconds were chosen in order to provide an accurate insight on how the algorithm is expected to perform when employed in 10 seconds of real EEG datasets, as explained in 2.8.

Chapter 4

EEG Spectrograms under anesthetics

Having validated the model in simulated ground truth data where the time-varying frequency content is known, it is now time to assess its performance on real EEG datasets. Datasets were taken from experiments where the anesthetics propofol, ketamine and sevoflurane were administered to patients. Those anesthetics (especially propofol) are currently the most widely used drugs for sedation during operational procedures, so they were chosen for model performance evaluation. As there are not ground truth data for the above cases, spectrograms obtained by the hybrid algorithm will be compared to purely discrete ones and to spectrograms obtained by non-parametric methods. Those include periodogram, state-space periodogram, multitaper spectrogram and state-space multitaper spectrogram.

4.1 Propofol

Propofol is the most widely used anesthetic drug worldwide. Administration of propofol triggers the apparition of an α frequency band, in addition to the slow oscillations which are prominent throughout the whole EEG.

The raw data that were used for spectrogram calculation are shown below. (Fig 4.1)

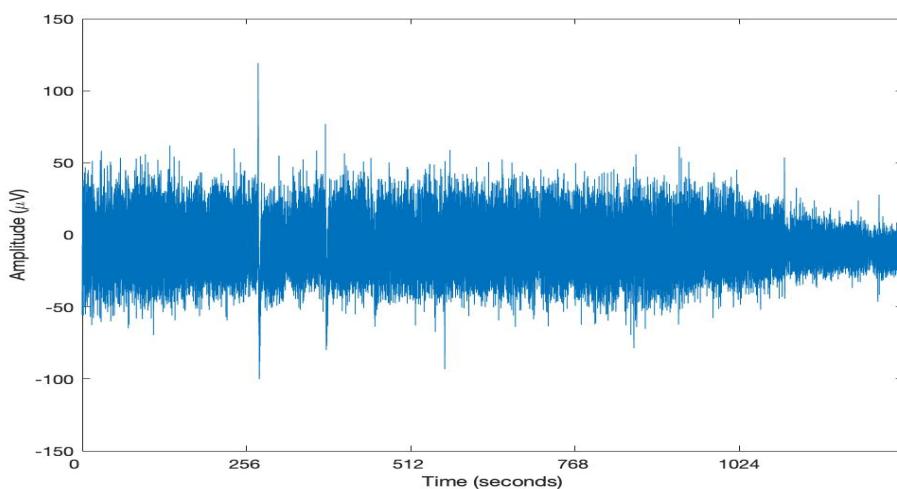


Figure 4.1: Raw EEG under propofol

The most widely used non-parametric spectral estimation techniques yielded the following spectrograms. (Fig 4.2) It is clear that each one of these 4 methods is able to identify the apparition of the α band in addition to the slow oscillations. Frequency power is in dB/Hz.

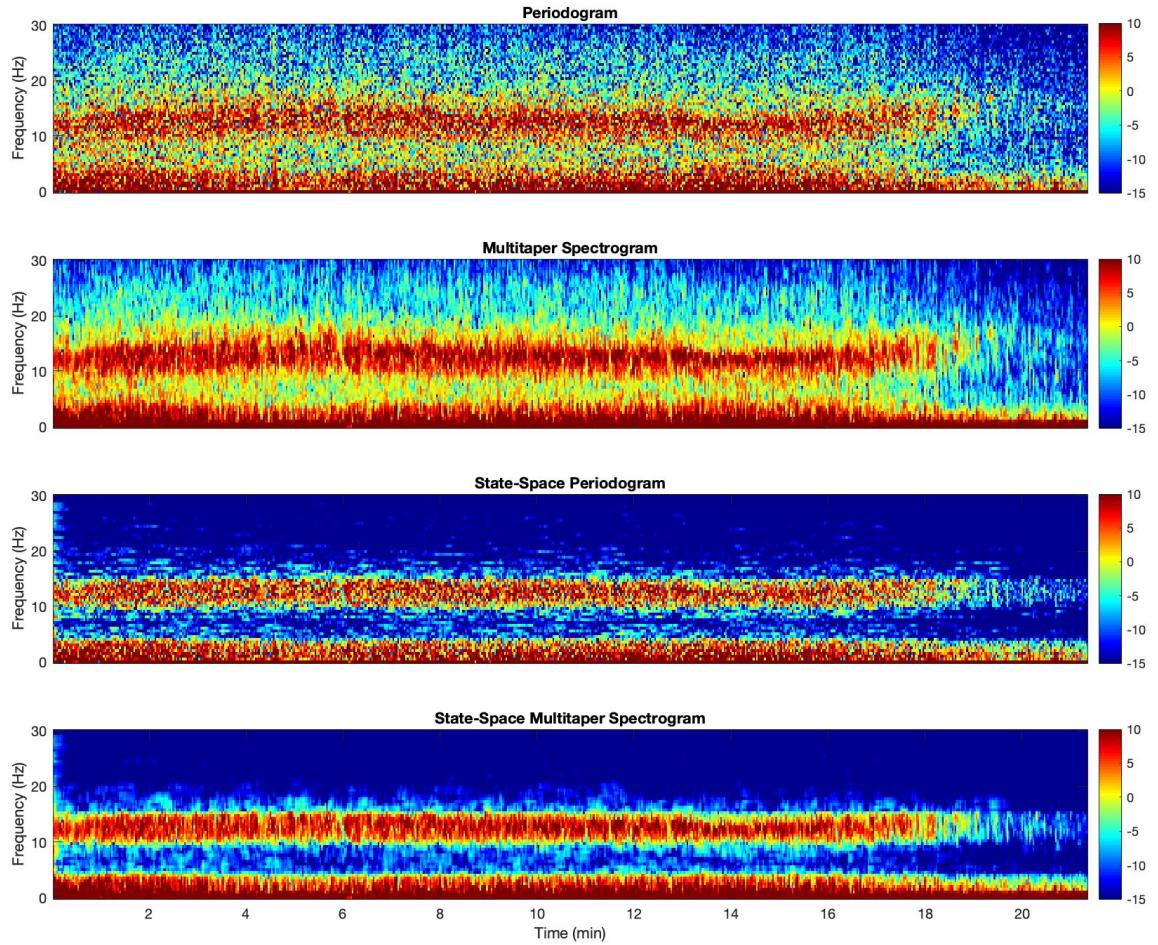


Figure 4.2: Non-parametric spectrograms of EEG under propofol

For comparison purposes, the same observation variance $R=0.05$ was used for the discrete and the hybrid case. Covariance matrices \mathbf{Q} were estimated using the adaptive EM routine over 5 seconds of minibatches as described in chapter 2. Order 14 was used for the autoregression. Parametric spectrograms follow. (Fig 4.3, 4.4, 4.5)

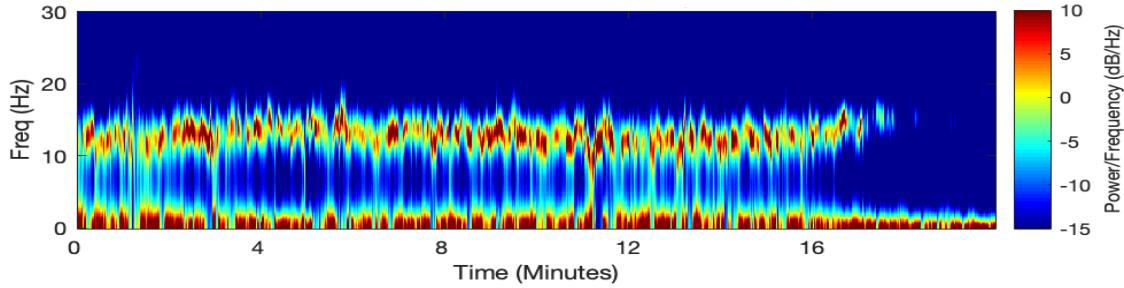


Figure 4.3: Discrete parametric spectrogram of EEG under propofol

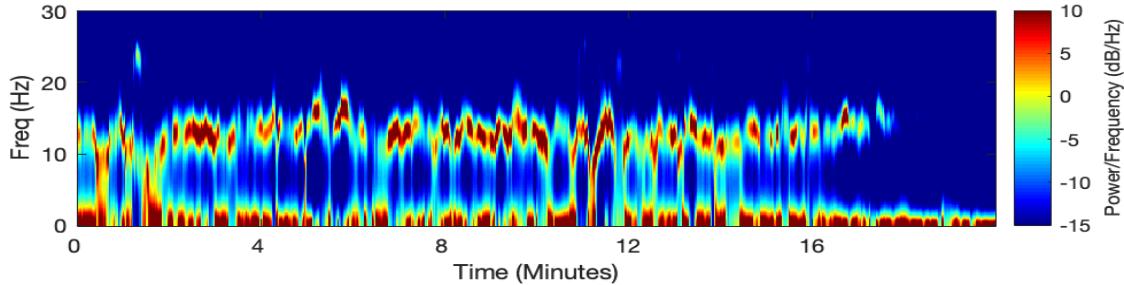


Figure 4.4: Exponentially smoothed discrete spectrogram of EEG under propofol

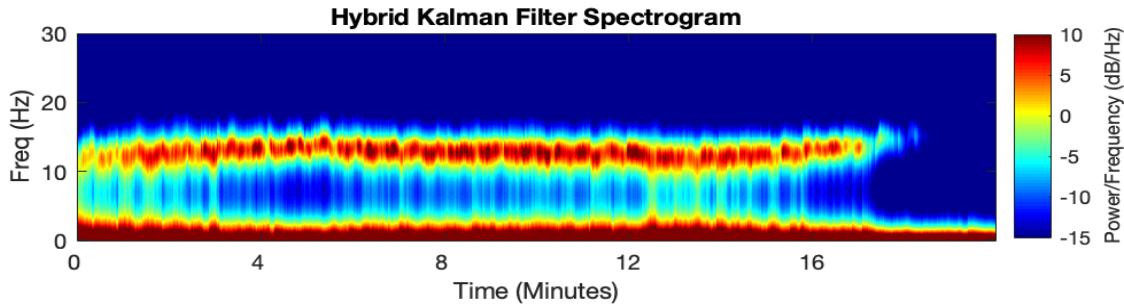


Figure 4.5: Hybrid spectrogram of EEG under propofol

The obtained spectrograms are quite similar. Extra noise is observed in the periodogram and the multitaper spectrogram, an indication of overfitting. On the other hand, the state-space periodogram and multitaper periodogram seem to cut more noise than actually exists. Indeed, this is a case of underfitting as it is very implausible that the α band is evolving as a completely straight line in an actual EEG experiment.

The discrete parametric spectrogram seems able to grasp the subtle changes in frequency but is not able to reduce the underlying noise to the same extent as the hybrid spectrogram. Exponential smoothing can potentially smooth the discrete spectrogram. In this case, it seems that it makes things worse by enhancing the noise, making artifacts more prominent across the spectrogram. Overall, the hybrid model is performing better in tracking the slight frequency changes along the α band without introducing extra noise.

Even though the hybrid spectrogram does yield the best spectrogram in terms of frequency resolution and reduction of noise, the difference between the methods is not that evident in the case of propofol, as this specific anesthetic triggers a clear α frequency band, easy to be tracked.

4.2 Ketamine

Ketamine is an anesthetic drug whose administration triggers bursts of activity in the EEG which in turn lead to the apparition of a β - γ frequency band. Being able to identify it and follow the temporal evolution of this specific band could be a potential challenge for the model. Two different EEG datasets were tested. One with more smooth bursts and one with sudden large spikes that are considered to be one of the most difficult challenges that the hybrid filter could have to face, as the change in the frequency content is big and abrupt. The raw data that were used for spectrogram calculation in the first case are shown below. (Fig 4.6)

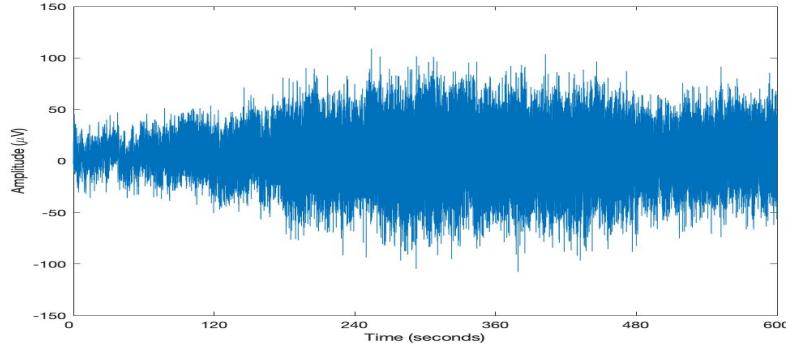


Figure 4.6: Raw EEG under ketamine 1

Non-parametric spectrograms follow. (Fig 4.7)

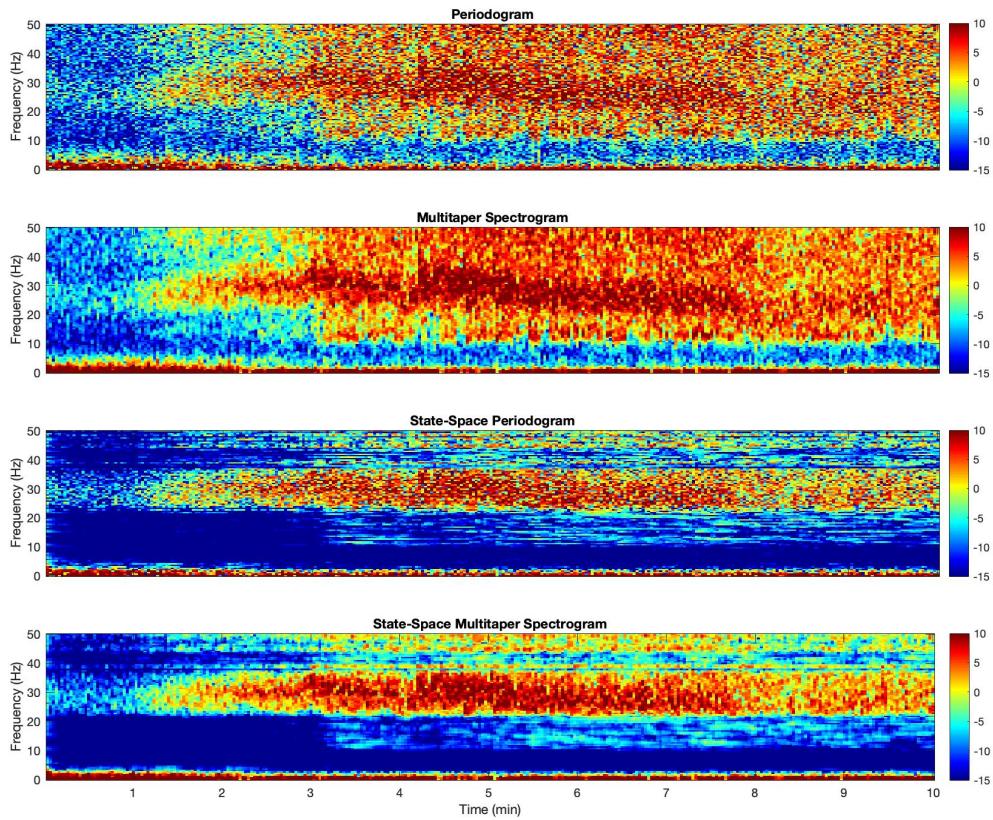


Figure 4.7: Non-parametric spectrograms of EEG under ketamine 1

The apparition of β - γ frequency band is evident in all non-parametric spectrograms with peaks at around 30Hz and 50Hz. Once again, the periodogram and multitaper spectrogram track changes along the bands but they suffer from high noise. On the other hand, as it was the case with propofol, state-space periodogram and multitaper spectrogram oversimplify the frequency evolution and yield a constantly evolving β - γ band.

For comparison purposes, the same observation variance $R=0.15$ was used for the discrete and the hybrid case. Covariance matrices \mathbf{Q} were estimated using the adaptive EM routine over 5 seconds of minibatches as described in chapter 2. Order 11 was used for the autoregression. Parametric spectrograms follow. (Fig 4.8, 4.9, 4.10)

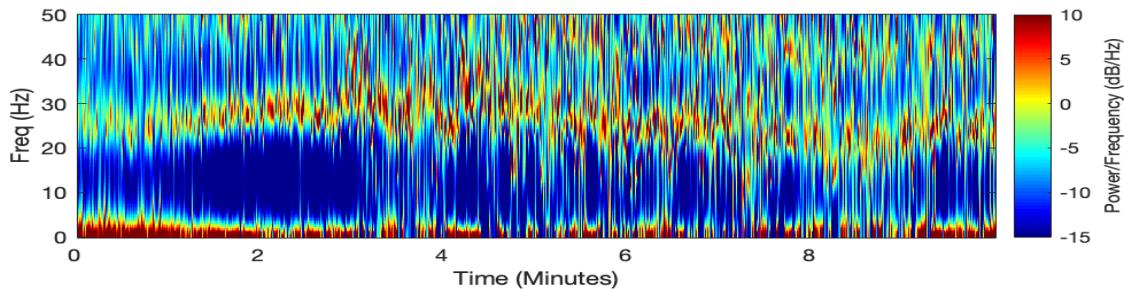


Figure 4.8: Discrete parametric spectrogram of EEG under ketamine 1

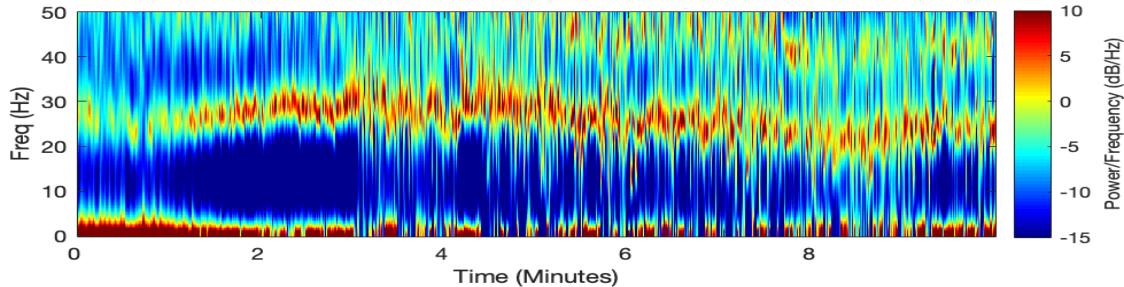


Figure 4.9: Exponentially smoothed discrete spectrogram of EEG under ketamine 1

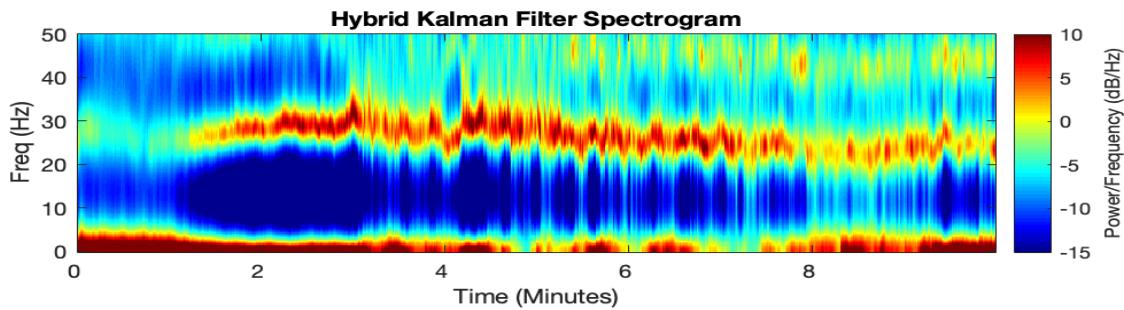


Figure 4.10: Hybrid spectrogram of EEG under ketamine 1

In this case, there are quite large differences between the spectrograms. Indeed, the discrete spectrograms are quite noisy and while in this case exponential smoothing improves the final result, the final spectrogram is still rather noisy. The hybrid spectrogram seems once again to provide the best result, combining the frequency tracking of the multitaper spectrogram with the noise rejection of the state-space multitaper spectrogram.

The second ketamine dataset is shown below. (Fig 4.11)

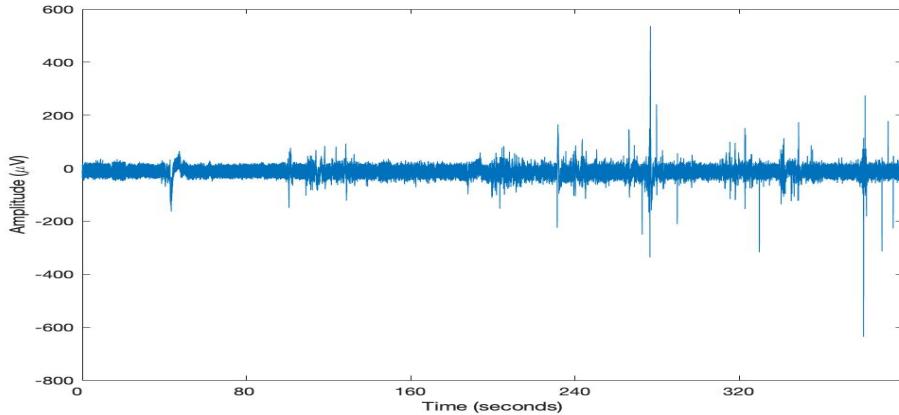


Figure 4.11: Raw EEG under ketamine 2

Non-parametric spectrograms are shown below. (Fig. 4.12)

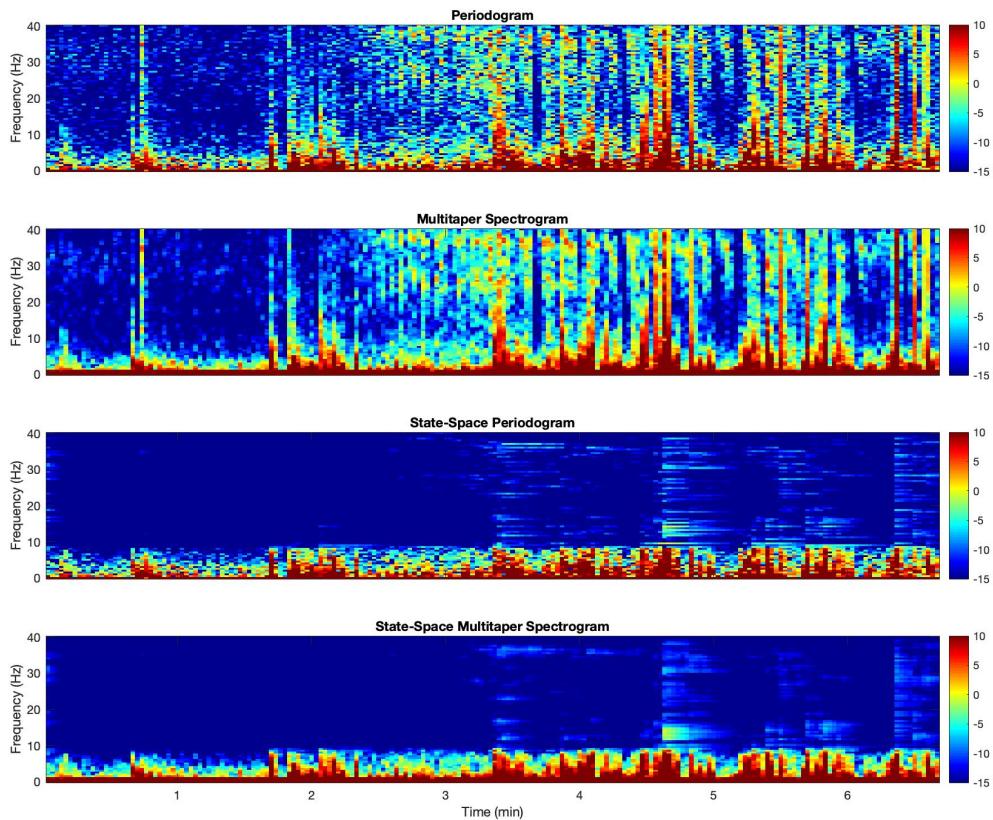


Figure 4.12: Non-parametric spectrograms of EEG under ketamine 2

The state-space approach is not able to identify the very abrupt changes in the frequency content due to the large neuronal spikes. The periodogram and the multitaper spectrogram show some activity in β - γ band at the times where the spikes occur, but the spectrograms are very noisy.

For comparison purposes, the same observation variance $R=0.07$ was used for the discrete and the hybrid case. Covariance matrices \mathbf{Q} were estimated using the adaptive EM routine over 5 seconds of minibatches as described in chapter 2. Order 18 was used for the autoregression. Parametric spectrograms follow. (Fig 4.13, 4.14, 4.15).

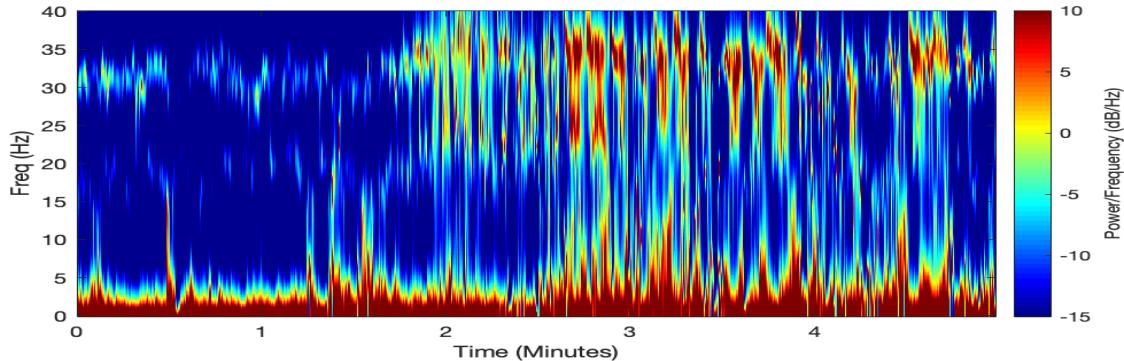


Figure 4.13: Discrete parametric spectrogram of EEG under ketamine 2

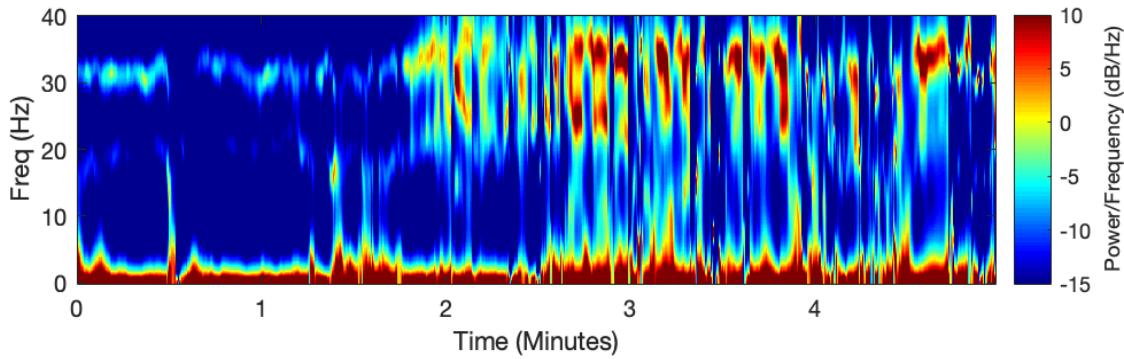


Figure 4.14: Exponentially smoothed discrete spectrogram of EEG under ketamine 2

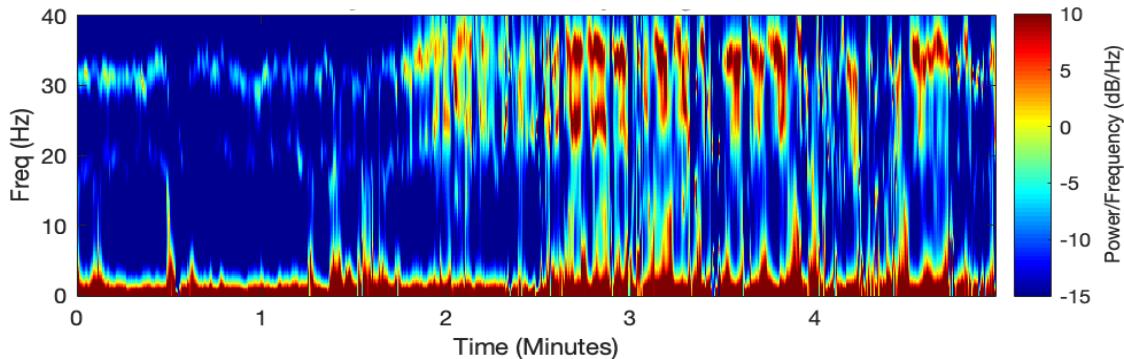


Figure 4.15: Hybrid spectrogram of EEG under ketamine 2

In this case, both discrete and hybrid filters perform similarly. This is expected since the EEG data was much less smooth than the previous datasets. Therefore, if the hybrid filter attempted to smooth more, there would be noise as a result of over-smoothness, and the results would be similar to those of state-space periodogram and state-space multitaper spectrogram, which fail to track the frequency correctly.

It is important that the hybrid filter is able to track the frequency even in this extreme case of fast and large neural spikes. Indeed, when bursts are observed in the raw data, the corresponding frequency power in the spectrogram "jumps" to higher values for higher frequencies.

To illustrate the ability of the hybrid filter to follow the sudden increase in power of higher frequency bands due to the large spikes, the autoregressive coefficients' evolution is shown (Fig 4.16).

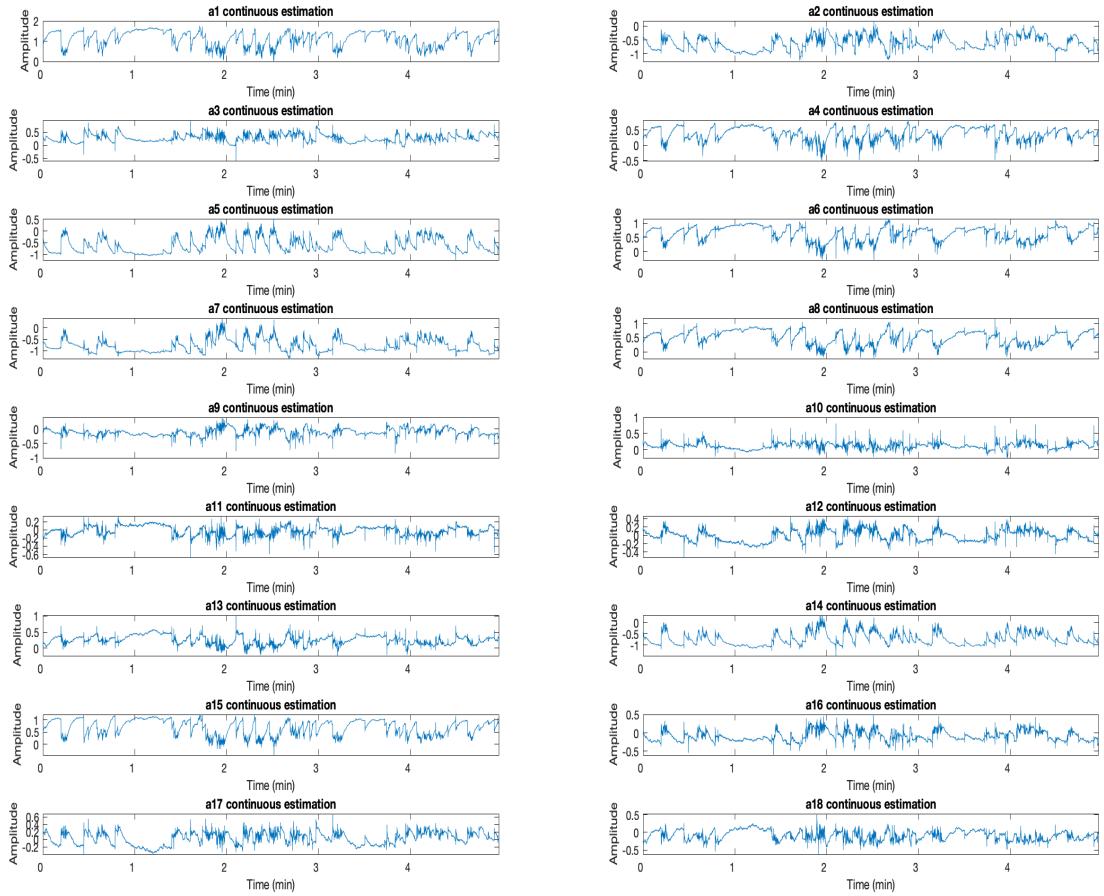


Figure 4.16: Autoregressive coefficients estimated using the hybrid filter

As it can be observed, the filter has the ability to cause sudden drops in the coefficients' values closer to zero to account for the increase in frequency power.

Note that due to outlier rejection, some samples have been eliminated, but as it becomes apparent they do not affect the frequency estimation; the filter is able to grasp all the abrupt frequency changes. Trying to run the filter in the normalized data without outlier rejection led to numerical instabilities. It can thus be concluded that outlier rejection greatly enhances filter stability without further implications in the frequency analysis of the signals.

4.3 Sevoflurane

Sevoflurane is an anesthetic whose use triggers a merge between α and slow oscillations. It will be interesting to see if the filter will be able to follow this merge. The raw data that were used for spectrogram calculation are shown below. (Fig 4.17)

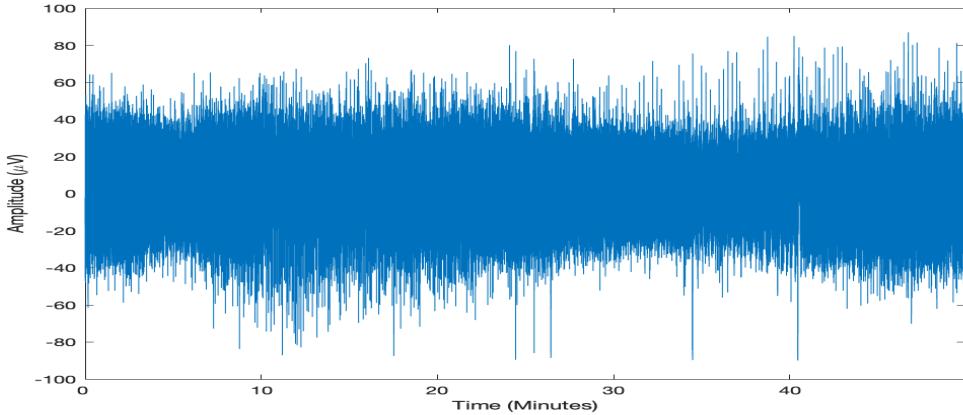


Figure 4.17: Raw EEG under sevoflurane with outlier rejection

Non-parametric methods after outlier rejection yielded the following spectrograms, where one can observe the merge of the two frequency bands described above. (Fig 4.18)

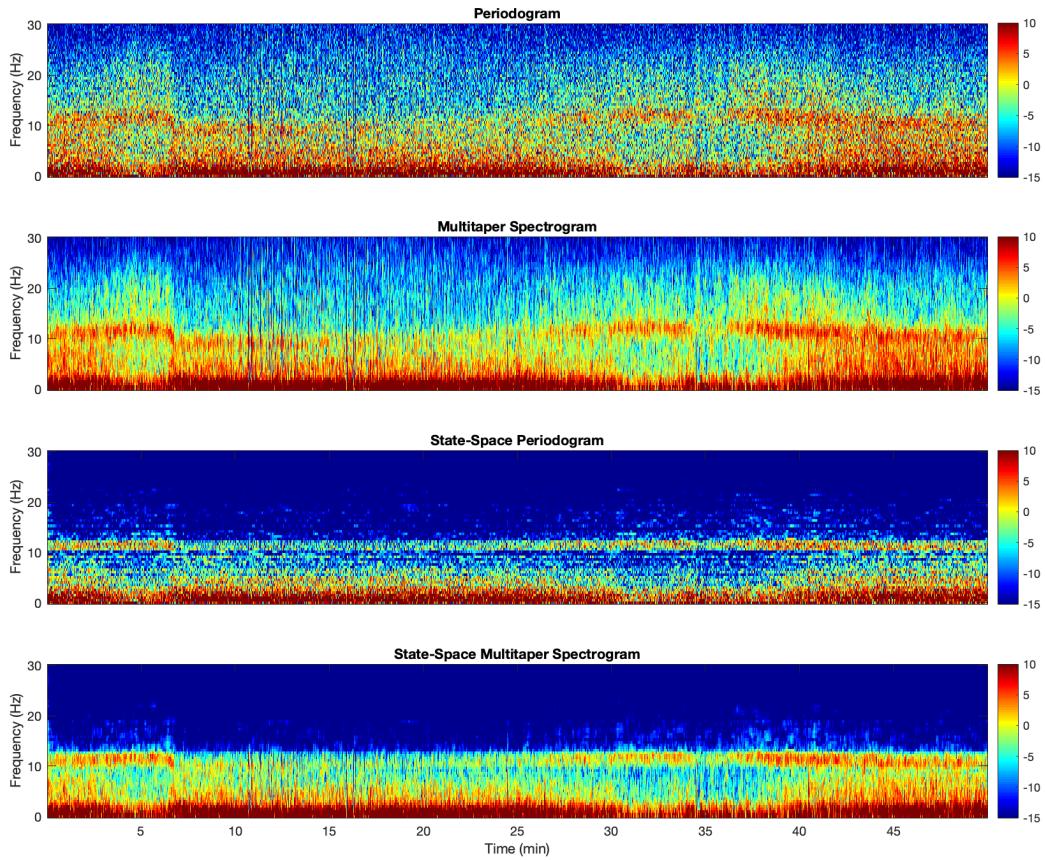


Figure 4.18: Non-parametric spectrograms of EEG under sevoflurane with outlier rejection

It is interesting to see the effect of outlier rejection. Without applying the rejection, running the non-parametric algorithms to the raw data below (Fig 4.19), the following spectrograms are obtained. (Fig 4.20).

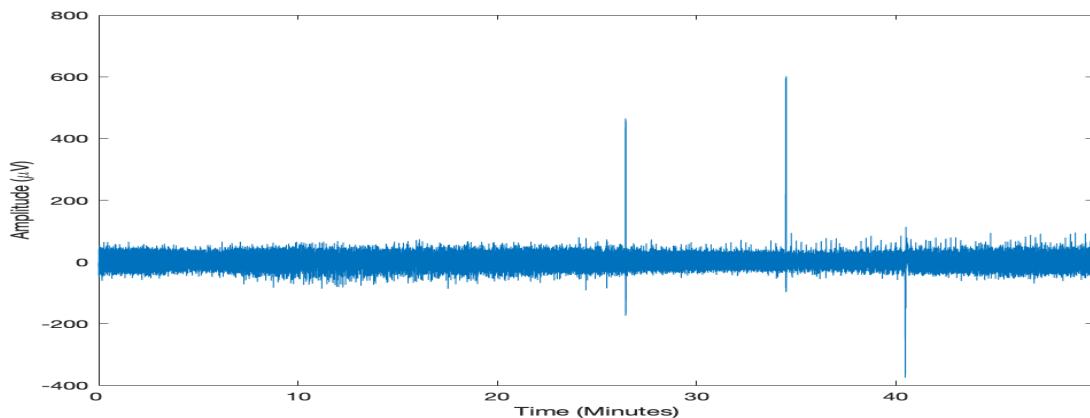


Figure 4.19: Raw EEG under sevoflurane without outlier rejection

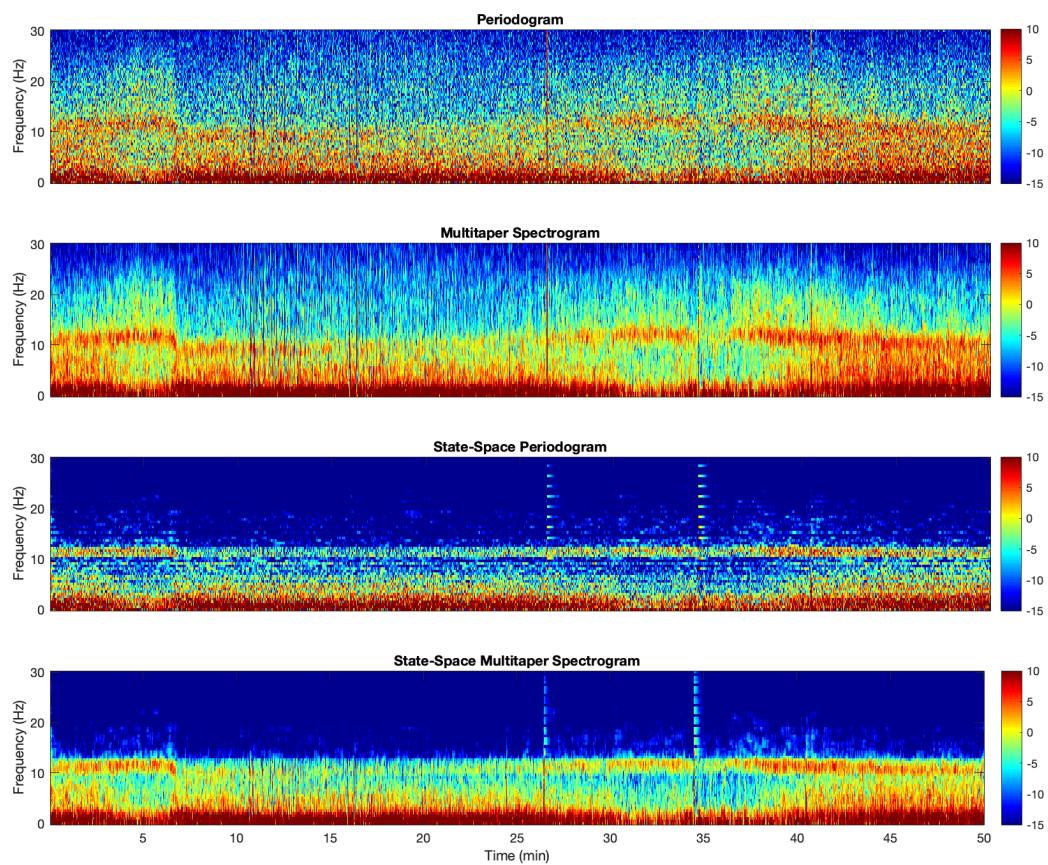


Figure 4.20: Non-parametric spectrograms of EEG under sevoflurane without outlier rejection

The presence of outliers causes the apparition of the red vertical lines at $t = 26.5\text{min}$, $t = 35\text{min}$ and $t = 41\text{min}$ in the spectrograms. Although there is a chance that those spikes are not artifacts, their removal yields smoother spectrograms while not affecting the frequency content, as proved by the identical spectrograms of Fig 4.18 and Fig. 4.20. Therefore, for the generation of the parametric spectrograms, outlier rejection will be employed.

For comparison purposes, the same observation variance $R=0.07$ was used for the discrete and the hybrid case. Covariance matrices \mathbf{Q} were estimated using the adaptive EM routine over 5 seconds of minibatches as described in chapter 2. Order 20 was used for the autoregression. Parametric spectrograms follow. (Fig 4.21, 4.22, 4.23).

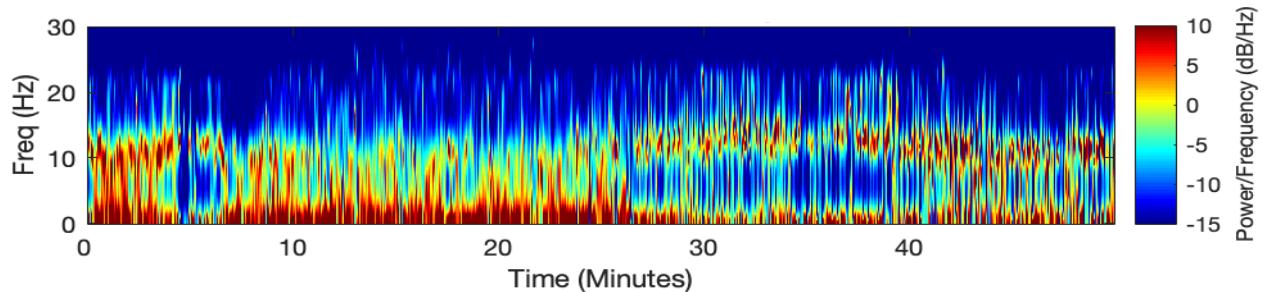


Figure 4.21: Discrete parametric spectrogram of EEG under sevoflurane

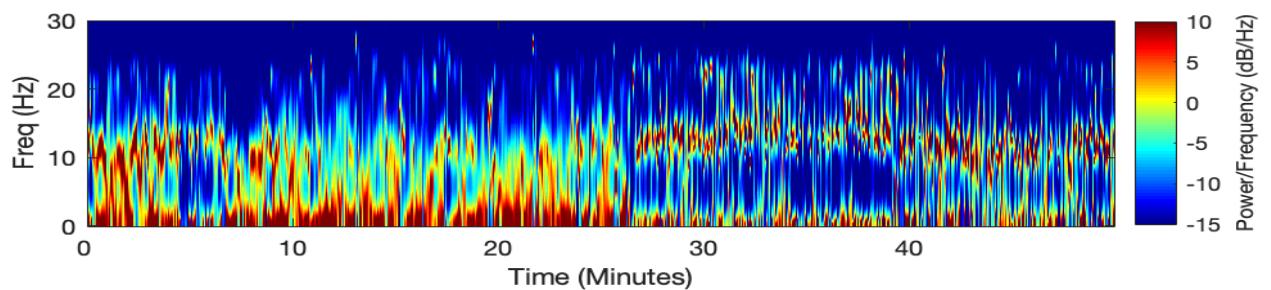


Figure 4.22: Exponentially smoothed discrete spectrogram of EEG under sevoflurane

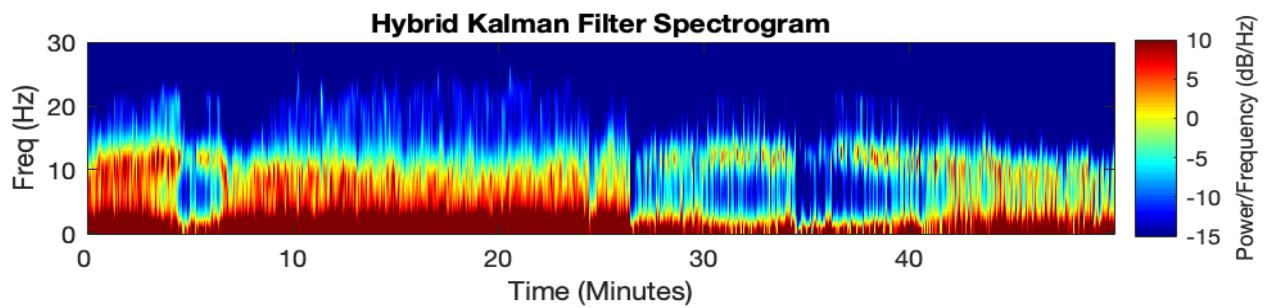


Figure 4.23: Hybrid spectrogram of EEG under sevoflurane

The advantage of the hybrid filter, due to its ability to capture the smoothness in the data is apparent again here. Indeed, the discrete and exponentially discrete spectrograms are noisier than the hybrid one. In relation to the non-parametric spectrograms, the pattern is once again observed: The hybrid spectrogram is characterized by the separability in the frequency bands of the state-space multitaper spectrogram, while also being able to track the frequency changes shown in the multitaper spectrograms, avoiding oversimplification. Their difference in smoothness is quite similar to the case of propofol, where the hybrid spectrogram also seemed to have captured the intrinsic smoothness in the EEG data better than its purely discrete counterparts.

Chapter 5

Discussion on the hybrid Normal model

In this chapter, some useful insights about the hybrid Normal model are given.

5.1 Comparative analysis

The hybrid filter is able to compromise efficiently the underfitting regime of the state-space multitaper periodogram with the overfitting regime of the multitaper spectrogram. Results of frequency analysis of three of the most widely used anesthetics consistently confirm that.

In comparison to the purely discrete versions, the hybrid filter is better able to capture the smoothness in the data, while at the same time this ability does not prevent it from following sharp frequency changes, as was illustrated by the spectrogram of Fig 4.15. The performance of the purely discrete filter with smaller covariance matrices as a measure to reduce the variability was tested, but a low covariance matrix led the model to miss frequency changes that the hybrid filter was able to track, while still capture the data smoothness. To better illustrate this, the propofol data will be used, as the frequency content of EEG under propofol is expected to be easier to be tracked in comparison to ketamine. Fig 4.3 shows the spectrogram obtained by the discrete model, using the covariance matrix yielded by the EM algorithm. The diagonal terms were in the order of 0.001. In an effort to reduce variability, an even smaller, diagonal, covariance matrix is tested, namely $\mathbf{Q} = 0.0001\mathbf{I}_p$. The result is the following (Fig 5.1).

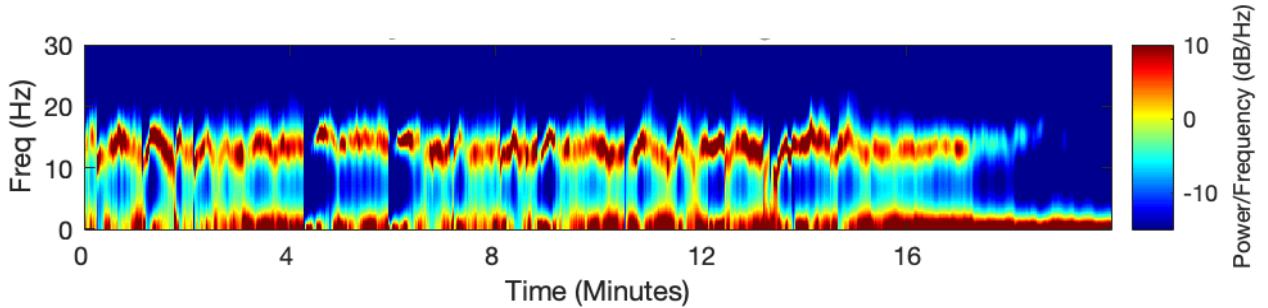


Figure 5.1: Discrete parametric spectrogram of EEG under propofol, $\mathbf{Q} = 0.0001\mathbf{I}_p$

Indeed, it can be concluded that lowering the covariance matrix to reduce variability of estimates of the discrete model is not a solution since it prevents the model from correctly estimate the changes in signal frequency and introduces artifacts in the spectrogram. Lower model orders and covariance matrices led to even worse results.

The price that has to be paid for the advantages of the hybrid model is higher computational complexity with respect to the non-parametric methods that leads to longer execution times. Longer execution times were also consistently observed with comparison to the purely discrete versions. This is expected since the integration of the differential equations for the state vector and the state covariance matrix, that is the main idea of this approach takes more time than just "jumping" from a value at time t_k to another at t_{k+1} .

To illustrate that point, the EEG dataset used to generate spectrogram 4.10 will be used.

Run on a 2.9 GHz Intel Core i5 with an 8GB RAM the following execution times were obtained for the hybrid and discrete models. (Table 5.1)

The tolerance of convergence for the EM algorithm was 0.001, the EM routine was run on an initial minibatch of 5 seconds. The algorithm was run five times for each case to establish an indicative average and standard deviation. The frequency resolution was 4 index/Hz. All values in the table are in seconds.

AR order	p=10	p=11	p=18	p=20
EM (5sec)	54.3±2.4	57.9±3.1	72.7±3.9	77.9±4.2
Hybrid Kalman Smoothing	223.6±6.7	256.7±8	282.6±8.5	302.4±10.4
Discrete Kalman Smoothing	127.3±3	149.1±3.9	178.2±5.2	204±5.6

Table 5.1: Execution times

For reference, when the same dataset was used for the non-parametric methods, 50 seconds elapsed before the 4 spectrograms were generated.

For this specific dataset the order that was used was 11. The spectrograms generated by the parametric methods are around 5 times slower for the hybrid and 4 times slower for the discrete. The difference in execution times becomes even larger when the algorithms are run on longer EEG datasets. Note that a way to accelerate the algorithm is to reduce the frequency resolution but it was empirically found that less than 2 index/Hz results in significant deterioration of the spectrogram quality.

Overall, the hybrid algorithm is able to combine the advantages of the multitaper and state-space multitaper spectrograms better than its purely discrete counterpart. This benefit comes at a higher computational cost.

5.2 Over/Under Fitting

It is interesting to note the trade-off that takes place between values of \mathbf{Q} , autoregressive model order, goodness-of-fit and noise in the spectrograms. When \mathbf{Q} has very large entries, typically diagonal terms larger than 0.5, the coefficients are "allowed" to fluctuate a lot and thus the model is able to almost perfectly fit the observations. In this overfitting regime, with very high values for the goodness-of-fit metrics, the produced spectrograms are noisy. This is a result of the fact that high \mathbf{Q} values allow the model to not only track frequency changes, but also capture the noise in the observations. On the other hand, when \mathbf{Q} has very low entries, typically diagonal entries lower than 0.001, the model is not able to identify the frequency changes, as the coefficients are not allowed to vary enough. As a result, underfitting takes place and the resulting spectrograms are over-smoothed. Consequently, there is some optimal \mathbf{Q} that

is able to correctly model frequency changes while not also fit the noise in the data. This \mathbf{Q} is approximated by the EM algorithm.

To illustrate this point, the data that were used to generated the first ketamine spectrogram (Fig 4.10) will be used.

Different order models were run (orders 10, 11, 16) with different diagonal covariance matrices (ranging from $0.0001\mathbf{I}$ to $5*\mathbf{I}$). Covariance matrices larger than $5*\mathbf{I}$ led to numerical instabilities, as the fluctuation in the parameters was too high, rendering the autoregressive model unstable. Observation noise variance was kept constant at 0.15. Mean Square Errors (MSE) and Normalized Mean Square Errors (NMSE) are shown in the following tables. (Tables 5.2, 5.3)

\mathbf{Q} /AR order	p=10	p=11	p=16
$0.0001\mathbf{I}$	0.0141	0.0136	0.0130
$0.001\mathbf{I}$	0.0138	0.0133	0.0127
$0.01\mathbf{I}$	0.0132	0.0126	0.0120
$0.1\mathbf{I}$	0.0119	0.011	0.0102
\mathbf{I}	0.0089	0.0079	0.065
$5\mathbf{I}$	0.0055	0.0045	0.0031

Table 5.2: MSE for different orders and different \mathbf{Q}

\mathbf{Q} /AR order	p=10	p=11	p=16
$0.0001\mathbf{I}$	0.3775	0.4076	0.4380
$0.001\mathbf{I}$	0.3997	0.4299	0.4614
$0.01\mathbf{I}$	0.4355	0.4680	0.5047
$0.1\mathbf{I}$	0.5119	0.5501	0.5996
\mathbf{I}	0.6614	0.7058	0.7675
$5\mathbf{I}$	0.8	0.8453	0.8970

Table 5.3: NMSE for different orders and different \mathbf{Q}

Overfitting can occur in two different ways: Very high model order p or very high covariance process noise matrix \mathbf{Q} . This is consistently confirmed by the trends in the two above tables. Indeed, MSE tends to be lower for higher order and larger \mathbf{Q} , while NMSE approaches 1 under the same conditions, indicating better goodness-of-fit to the raw data.

The spectrograms of orders 10,11 and 16 for $\mathbf{Q}=0.0001\mathbf{I}, 0.01\mathbf{I}$ and $5\mathbf{I}$ follow (Fig 5.2,5.3,5.4).

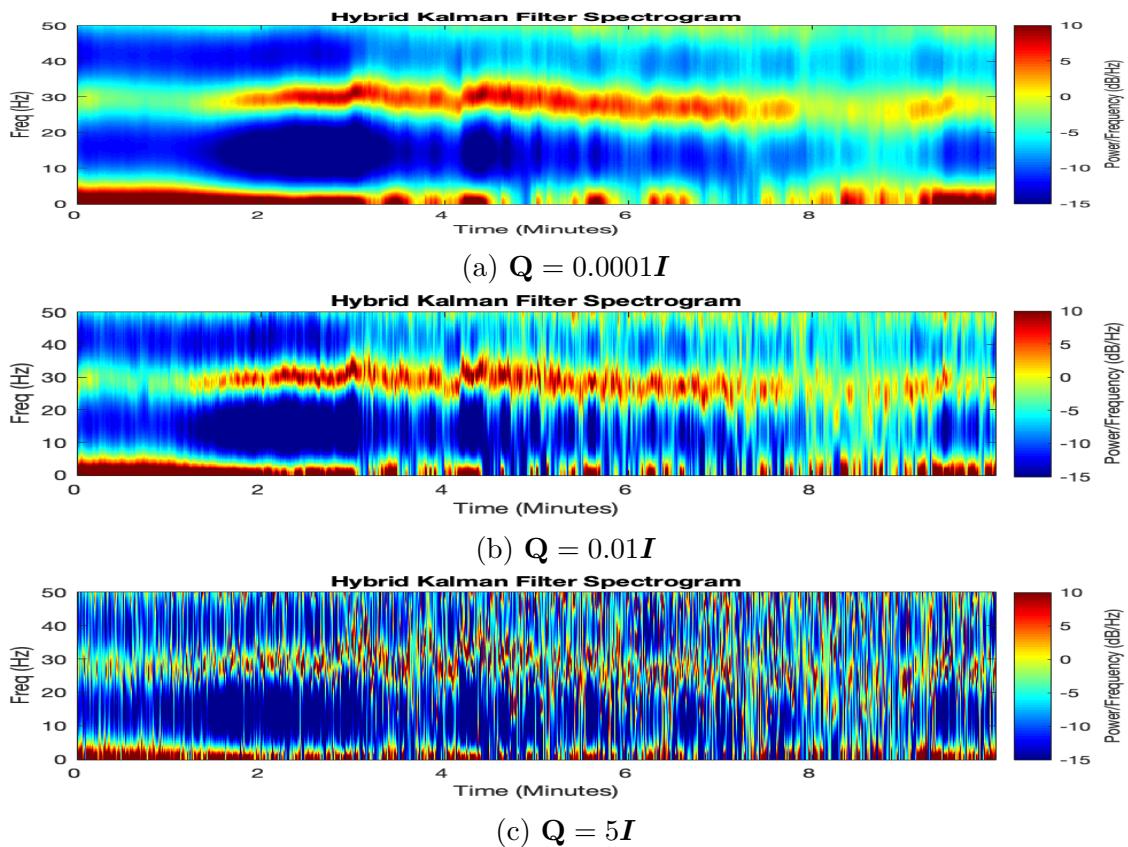


Figure 5.2: Spectrograms of order $p=10$

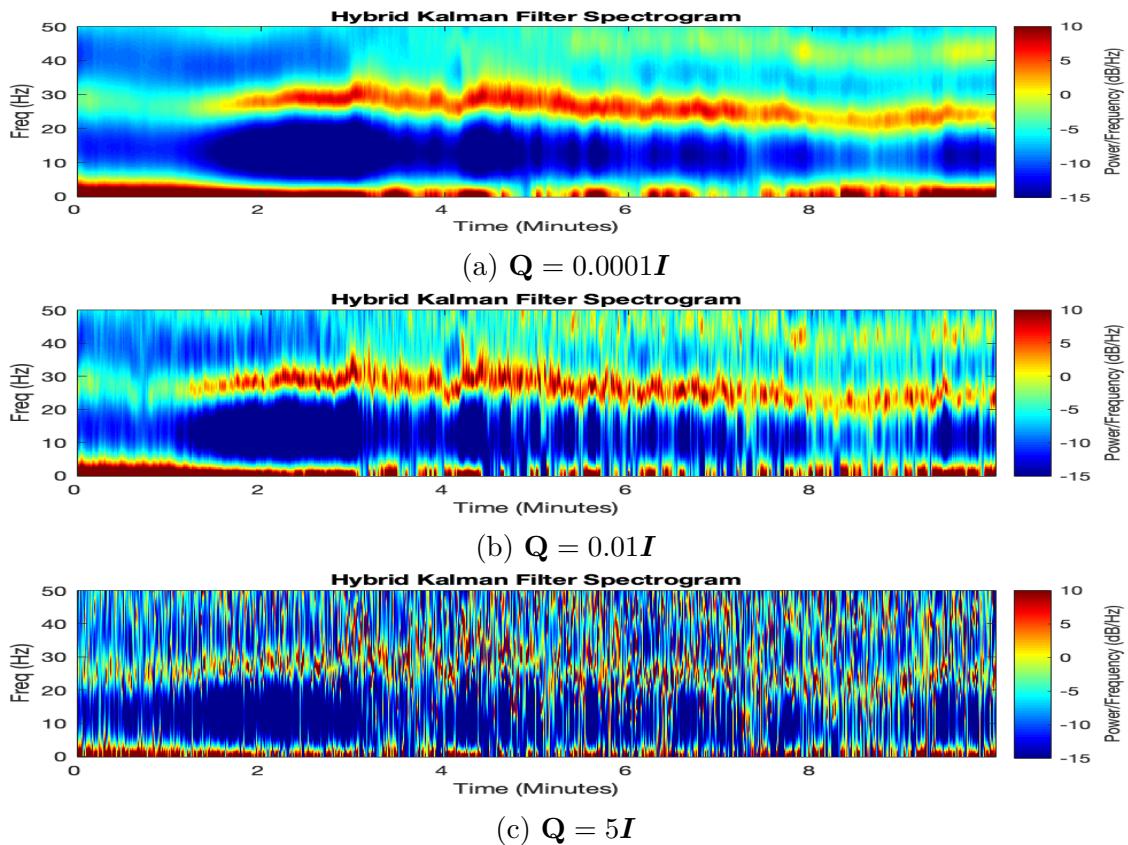


Figure 5.3: Spectrograms of order $p=11$

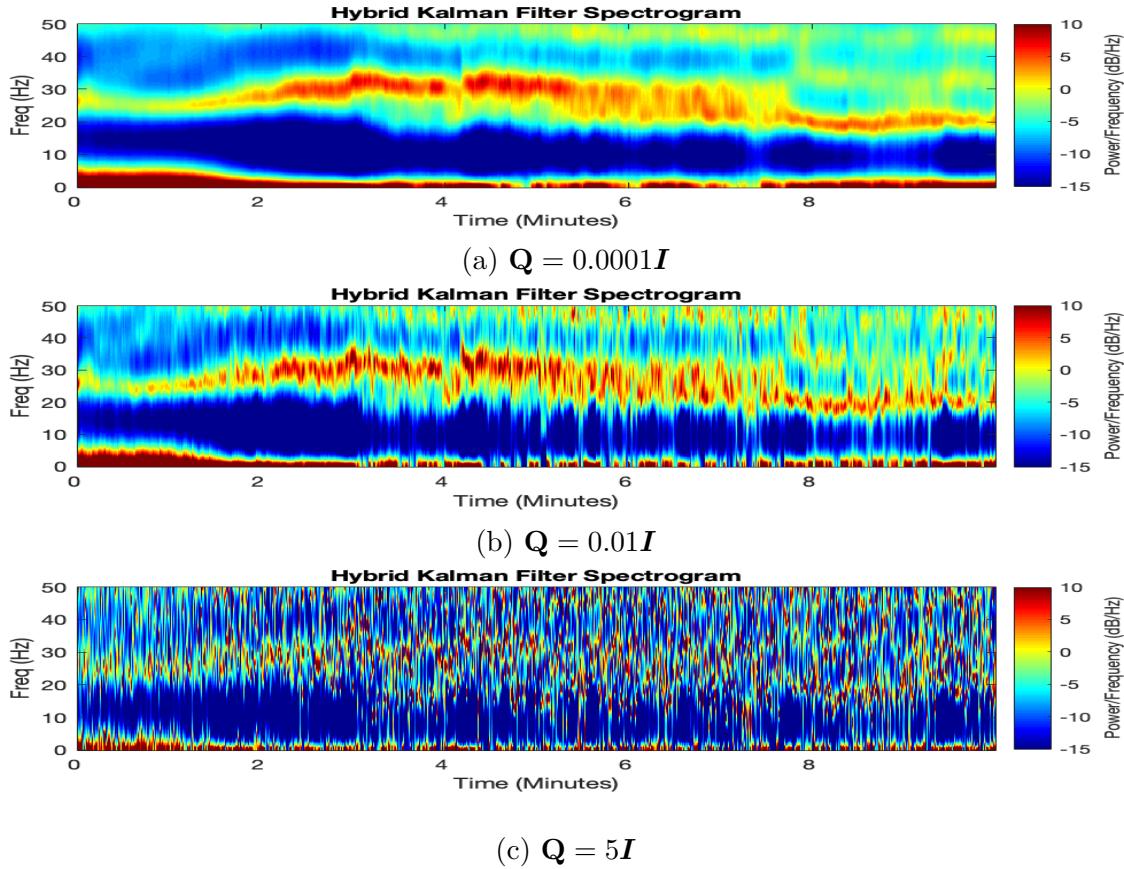


Figure 5.4: Spectrograms of order $p=16$

Observing the above spectrograms, it can be concluded that for autoregressive order $p=10$, the model seems to underfit as it fails to capture the frequency bands at around 45 Hz that is apparent in the non-parametric spectrograms. On the other hand, for $p=16$, spurious effects start to happen as the frequency band at 20-40 Hz becomes thicker after $t = 4min$, indicating some extra noise in the spectrogram. Order 11 seems to be optimal here, as it was also indicated by the AIC criterion (Fig 5.5). The evolution of the marginal likelihood in function of EM iterations is also shown (Fig 5.6). Note the improvement in the spectrograms between Fig. 4.10 and Fig 5.3b. In the former, covariance matrix \mathbf{Q} was calculated using the EM algorithm while in the latter it was hard-coded to be $\mathbf{Q} = 0.01\mathbf{I}$. The artifacts in the slow oscillations, where for some intervals the power in that band decreases to around -15dB, that occur by using a diagonal covariance matrix, are significantly eliminated as shown in Fig 4.10. Finally, for reference, the NMSE for the model with the covariance matrix estimated using the EM algorithm was 0.4368, very close to the case of $\mathbf{Q} = 0.01\mathbf{I}$.

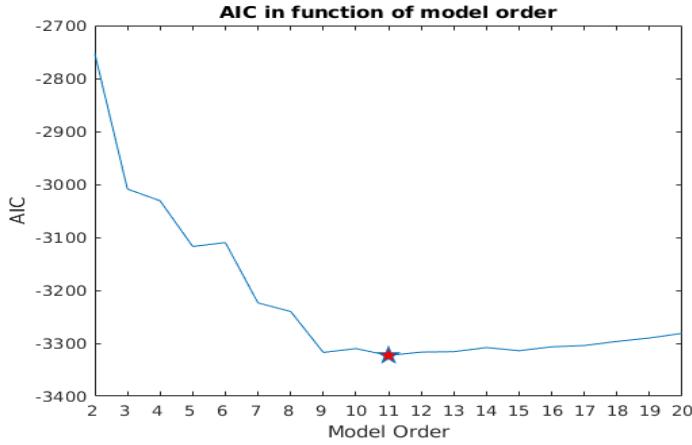


Figure 5.5: Akaike information criterion (AIC) for orders 2 until 20

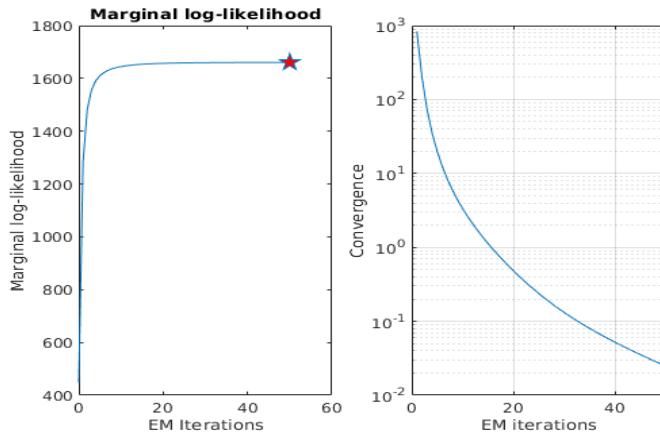


Figure 5.6: Evolution of marginal likelihood in function of EM iterations for the chosen order $p=11$

Concerning \mathbf{Q} , it is clear that for very low diagonal values, e.g., 0.0001, the model is too slow to be able to track the frequency evolution of the signal efficiently. As a result, the algorithm fails to accurately track the band at 45 Hz, irrespective of the model order. On the other hand, when the values of \mathbf{Q} become high, as is the case in the last spectrogram of each of the previous images, the spectrogram becomes incredibly noisy and uninformative. Finally, the non-decreasing property of the EM algorithm observed in Fig. 5.6 provides another validation metric for the model.

Overall, it is interesting to see that the optimal spectrogram is not obtained by the model parameters that yield the highest goodness-of-fit with the raw data, but by the model parameters that are able to grasp the frequency content while discarding most of its noise components.

5.3 Smoothness differences in frequency tracking between discrete and hybrid models

To further illustrate the additional smoothness constraint that the hybrid approach imposes, due to the integration of the states in the hybrid model, the following setup is proposed.

An artificial sinusoidal signal $z_k = A_k \sin(\omega_k k) + v_k$ is generated at a sampling rate $f_s = 250\text{Hz}$, similar to the procedure followed in chapter 3. v_k is added white Gaussian noise with zero mean and unit variance. Amplitude evolves as $A_k = 1 + k/N, 0 \leq k \leq N$.

Let ω_k evolve following the equation below.

$$\omega_k = \begin{cases} 50 + 2\frac{k}{f_s}, & \text{if } 0 \leq \frac{k}{f_s} < 10 \\ 80 - \frac{k}{f_s}, & \text{if } 10 \leq \frac{k}{f_s} < 30 \end{cases}$$

Using this signal, discrete and hybrid models can be used to estimate the coefficients of the autoregression. In this case, in order to estimate the dominant instantaneous frequency, the phase of the poles is all that is necessary, since it determines the peak in the power spectrum of the signal.

Having calculated the coefficients, an estimation of the instantaneous frequency can be given according to the following procedure.

Let $P(k) = 1 - \sum_{i=1}^p a_{k|N}^i z^{-i}$ be the characteristic polynomial of the autoregression at time t_k .

P can be written as $P(k) = \prod_{i=1}^p (1 - c_{k|N}^i z^{-1})$, where $c_{k|N}$ are the roots of the polynomial P .

$c_{k|N}$ can be written as $c_{k|N} = r_{k|N} e^{-i\omega_{k|N}}$, where $r_{k|N}$ is the modulus and $\omega_{k|N}$ is the phase of each root $c_{k|N}$.

Therefore, by computing the roots of the characteristic polynomial at each time t_k when the autoregressive coefficients are updated and then computing their phase ω , an estimation of the dominant instantaneous cyclical frequency is obtained. For an even model order, the complex roots are complex conjugate numbers and therefore, the absolute value of their phase coincides. The phase of either pole can be used.

The dominant frequency is proportional to the phase of the complex roots of P and can be obtained as: $f_{k|N} = \frac{f_s |\omega_{k|N}|}{2\pi}$.

The index notation $k|N$ is used to indicate that the variable above that index is instantiated at time t_k and has been smoothed using all the future observations up to time t_N .

Applying this procedure for a discrete and a hybrid model on the simulated data (Fig.5.7) using an autoregressive model of order 2, $R = 1$, and $\mathbf{Q} = 0.001\mathbf{I}$ the following estimations are obtained (Fig.5.8). A low order was used for increased tractability of the poles.

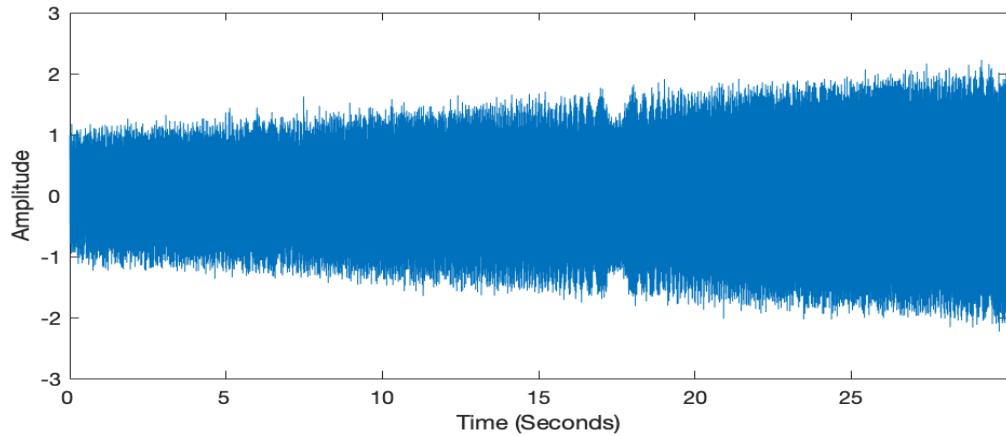


Figure 5.7: Raw data for instantaneous frequency estimation

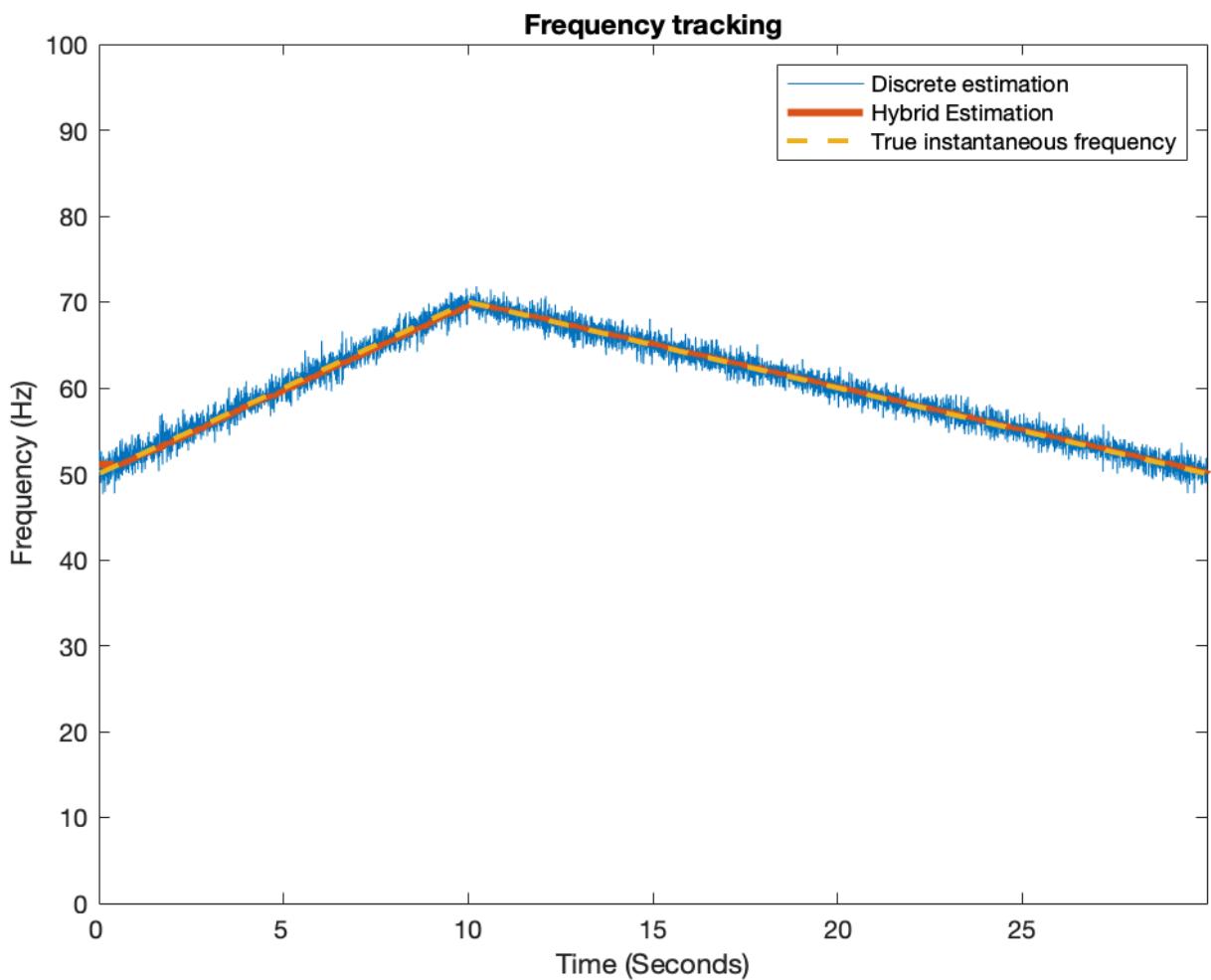


Figure 5.8: Instantaneous frequency estimation: Linear transitions

To facilitate the comparison of the two methods, a zoomed in figure at the point where the frequency change is most abrupt ($t = 10$ seconds) is given below (Fig. 5.9).

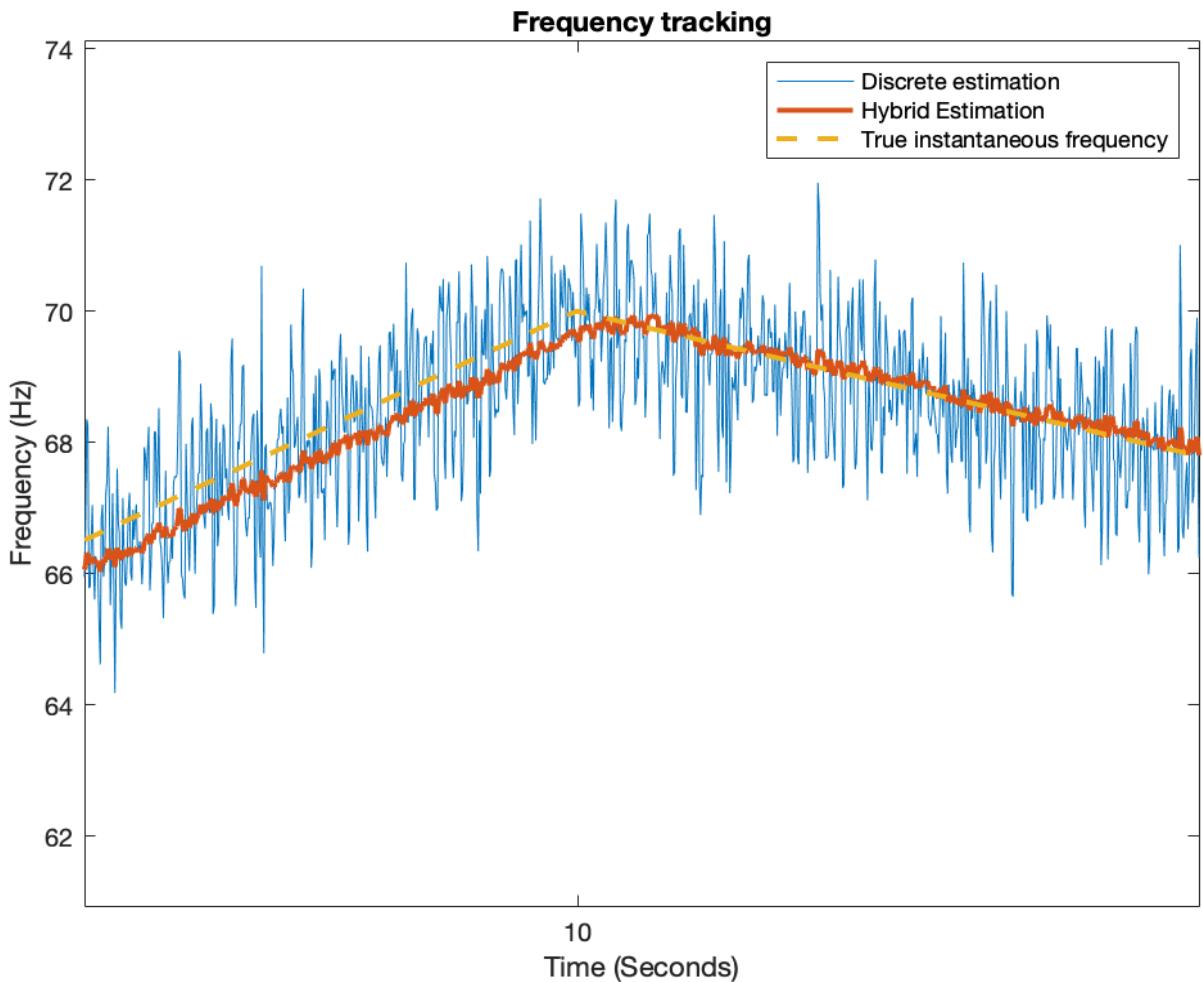


Figure 5.9: Instantaneous frequency estimation at 10 seconds

It is obvious that the hybrid estimates of the instantaneous frequency fluctuate much less around the true value when compared to the purely discrete estimates. Note that a lower covariance matrix could potentially be used for the discrete model, but as it was shown in 5.1, such a strategy does not help in efficiently smoothing EEG spectrograms.

Another useful example to give insights into the smoothness aspect of the hybrid model is to run the two methods in the simulated data of Chapter 3 with the step changes in frequency. The following figure shows the results in frequency tracking (Fig 5.10).

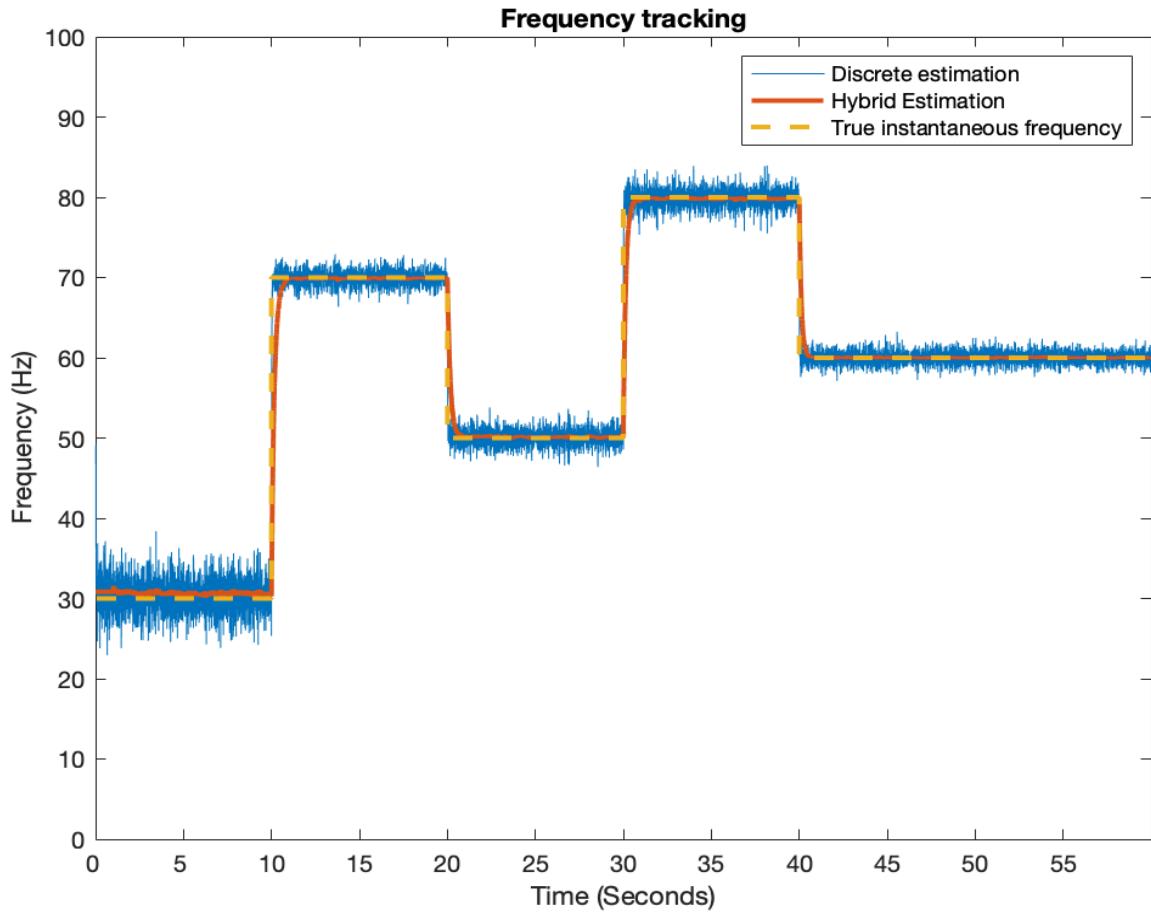


Figure 5.10: Instantaneous frequency estimation: Step transitions

Here, it is interesting to note that the hybrid estimations do not jump directly from one value to another during the step transitions, as is the case for the discrete. Instead, they exponentially approach the new set value for the frequency. This is another indication that the hybrid model assumes some intrinsic smoothness which it attempts to approximate when calculating the spectrograms. Of course, for this specific case, there is no smoothness in the data at the times of the step transitions and that is why the model needs a short time until it identifies the new frequency band.

Finally, to illustrate that point in real EEG data, the figures 5.11 until 5.14 of the next pages show the coefficients' evolution in the case of propofol and ketamine data that were used to generate the spectrograms of page 23 and 25 respectively, for the discrete and the hybrid models.

The smoothness metrics, as defined in section 2.9, are shown in the following table.

	Discrete	Hybrid
Propofol	$3.3380 \cdot 10^{-5}$	$6.5674 \cdot 10^{-7}$
Ketamine	$3 \cdot 10^{-3}$	$9.4747 \cdot 10^{-5}$

Table 5.4: Smoothness Metrics

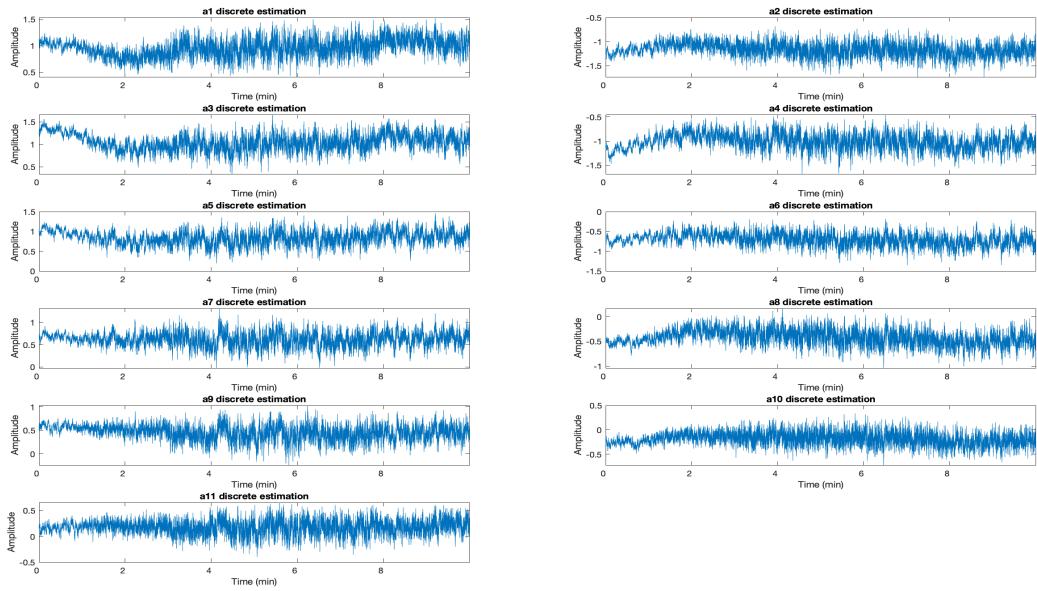


Figure 5.11: Coefficients of discrete model for ketamine

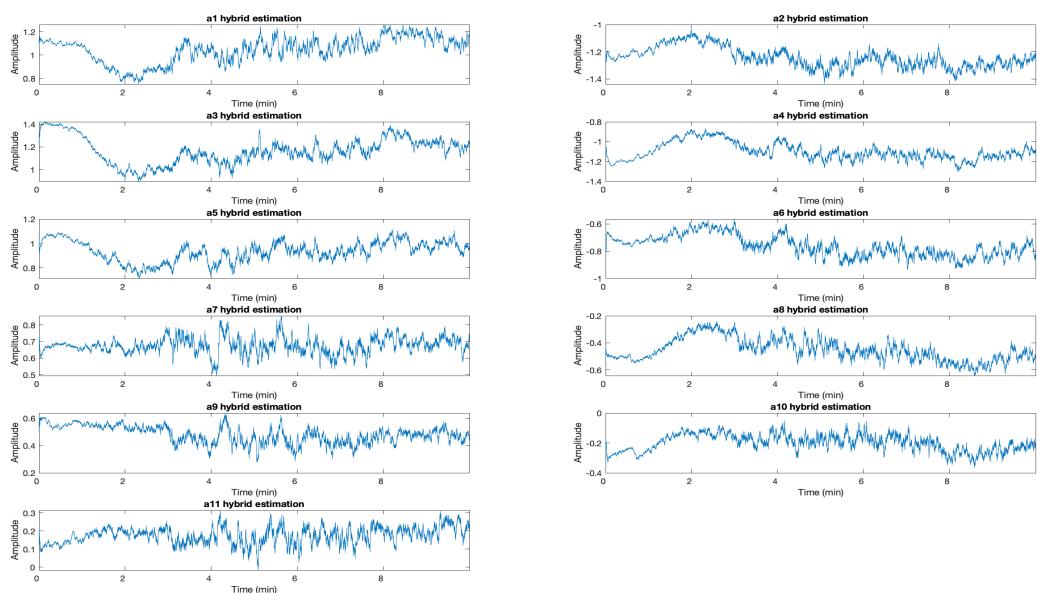


Figure 5.12: Coefficients of hybrid model for ketamine

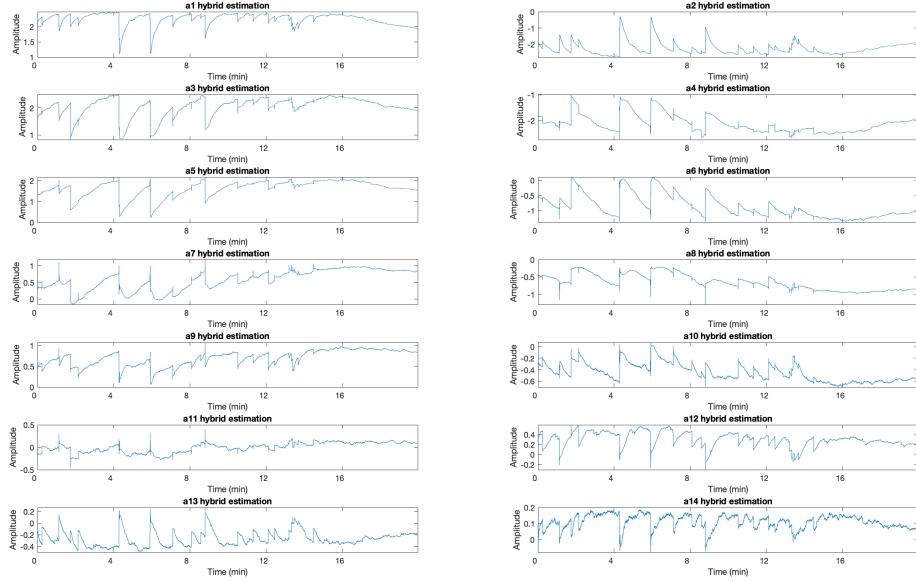


Figure 5.13: Coefficients of hybrid model for propofol

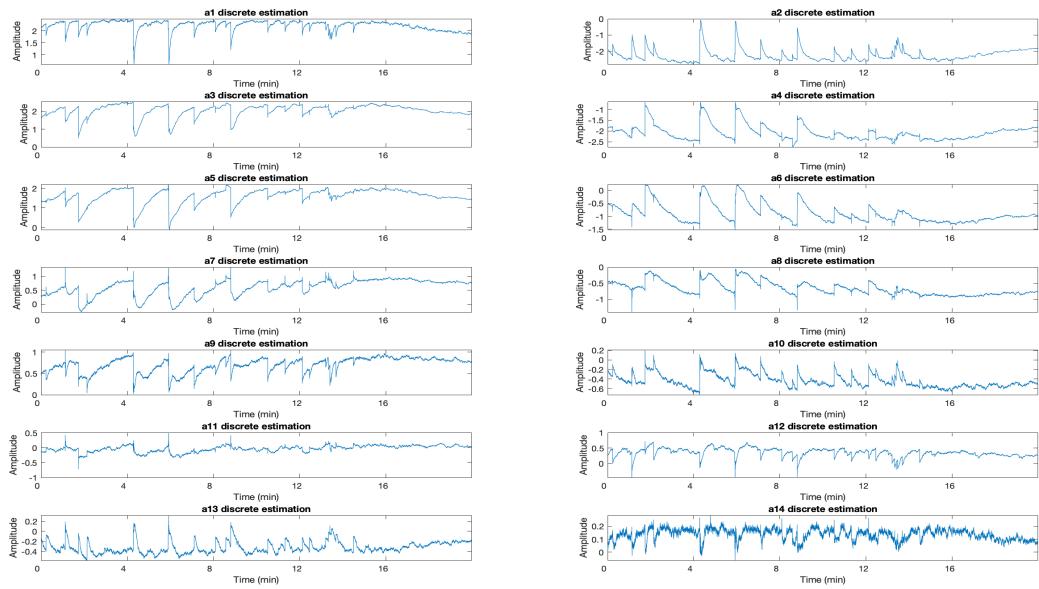


Figure 5.14: Coefficients of discrete model for propofol

As someone would conclude by visually inspecting the coefficients, the metrics of Table 5.4 verify their smoother time evolution.

The above figures were used to illustrate the advantage of the hybrid method: Yield smoother frequency spectra while tracking abrupt changes in the frequency structure of non-stationary signals.

Chapter 6

Extending the hybrid model: Cauchy state noise

As suggested in [32], the use of non-Gaussian state noise can be advantageous in decomposing EEG spectra via state-space modeling. More specifically, replacement of Gaussian by Cauchy noise can provide significant improvements in the tracking of abrupt frequency changes while at the same time provide a smooth spectrogram. In [32], the performance of the filter is better than the Gaussian one in identifying frequency changes in abrupt regimes but performs worse in stationary parts of simulated data. Motivated by the approach in [32], a bootstrap hybrid algorithm is described here that attempts to improve the tracking performance of the Gaussian filter while mitigating outliers in more stationary parts, by taking advantage of hybridity.

The model is formulated as a self-organizing state-space model and bootstrap hybrid sequential importance resampling is used to fit it to artificial and EEG data. Motivation for the use of the Cauchy distribution as state noise is briefly given below.

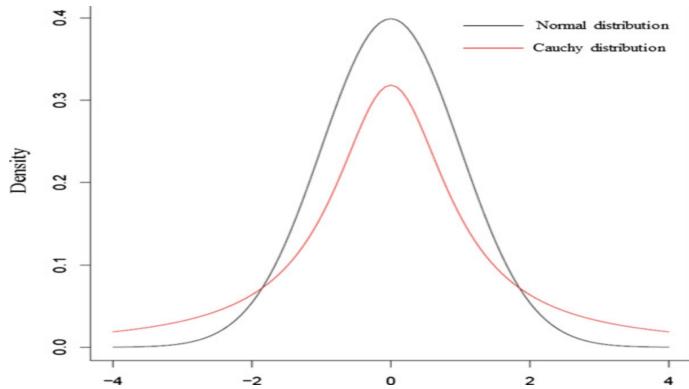


Figure 6.1: Normal vs Cauchy distributions

The heavier tails of the Cauchy distribution (Fig. 6.1) will lead to the proposal of aberrant values for the change of the autoregressive parameters more often than its Gaussian counterpart. Those values will then be weighted according to the observations. As a result, when there are no significant changes in the EEG the parameters will not change much as low proposed values will have larger weights, while if an abrupt change is detected in the EEG, the model will be able to better capture it by placing higher importance on the larger values generated by the Cauchy distribution. Details of the model and algorithm are given in the rest of this chapter.

6.1 Self-Organizing State-Space Formulation

The model evolves according to the following equations:

$$\text{State Equation } d\mathbf{x}(t) = \mathbf{w}(t)dt \quad (1)$$

$$\text{Observation Equation } z_k = \mathbf{H}_k \mathbf{x}_k + v_k \quad (2)$$

$\mathbf{w}(t) = [w_1(t), w_2(t), \dots, w_p(t), \epsilon_1(t), \epsilon_2(t), \dots, \epsilon_p(t)]^T$, where $w_j(t) \sim \mathcal{C}(0, q_j(t))$ is a continuous Cauchy process with 0 mean and scale parameter $q_j(t)$ and $\epsilon_j(t) \sim \mathcal{N}(0, R_\epsilon)$ is a continuous Gaussian process with 0 mean and variance R_ϵ

$v_k \sim \mathcal{N}(0, R(t))$ is a discrete scalar white Gaussian observation noise with 0 mean and variance $R(t)$

$\mathbf{x}(t) = [a_1(t), a_2(t), \dots, a_p(t), \ln(q_1(t)), \ln(q_2(t)), \dots, \ln(q_p(t))]^T = [\mathbf{a}(t), \mathbf{q}(t)]^T$ is the augmented state vector with the continuous time-varying autoregressive coefficients and their respective scale parameter. To ensure positivity of the scale parameters, their logarithm is estimated.

$\mathbf{H}_k = [z_{k-1}, \dots, z_{k-p}, 0, \dots, 0]$ is the observation matrix at time t_k comprising the p past discrete observations and p zeros

$\mathbf{x}_k = \mathbf{x}(t_k)$ is the sampled state vector at time t_k

p : autoregressive model order

z_k : observation at time t_k

Let T denote the total number of observations in the data, indexed by k .

Let N denote the total number of particles used for the sequential importance resampling, indexed by i .

Let f_s denote the sampling frequency of the observations.

Let j index the autoregressive parameters $[a_1(t), a_2(t), \dots, a_p(t)]$.

Based on this formulation, the algorithm described in the next section is used to fit the model to EEG data.

6.2 Bootstrap Hybrid Filtering with Sequential Importance Resampling

6.2.1 Discretization of the Ito state Integral using the Euler-Maruyama scheme

To start the algorithm, it is necessary to discretize (1). To do so, the Euler-Maruyama scheme is deployed, briefly described here.

For a discretization step Δ , (1) can be written as $\mathbf{x}(t + \Delta) = \mathbf{x}(t) + \int_t^{t+\Delta} d\mathbf{w}(t)$ (2). According to Euler-Maruyama, the Ito integral in equation (2) can be discretized as $\Delta\mathbf{x}(t) = \Delta\mathbf{w}(t)$, where $\Delta\mathbf{x}(t) \approx \mathbf{x}(t + \Delta) - \mathbf{x}(t)$, $\Delta\mathbf{w}(t) \sim \mathcal{C}(\mathbf{0}, \mathbf{Q}\Delta)$, for the first p dimensions where \mathbf{Q} is the diagonal matrix with entries $q_j(t)$ and $\Delta\mathbf{w}(t) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_q\Delta)$, where \mathbf{Q}_q is the diagonal matrix with entries $\epsilon_j(t)$ for the next p dimensions.

The Self-Organizing state-space equations can now be written as:

$$\text{State Equation: } \Delta\mathbf{x}(t) = \Delta\mathbf{w}(t) \quad (3)$$

$$\text{Observation Equation } z_k = \mathbf{H}_k \mathbf{x}_k + v_k \quad (4)$$

The rest of the algorithm will be based on the above discretized formulation.

6.2.2 Sampling from a Cauchy distribution using the inverse transform method

The probability density function (PDF) of a Cauchy distribution with location μ and scale σ is

$$f(x; \mu, \sigma) = \frac{1}{\pi\sigma \left[1 + \left(\frac{x-\mu}{\sigma} \right)^2 \right]} = \frac{1}{\pi\sigma} \left[\frac{\sigma^2}{(x-\mu)^2 + \sigma^2} \right] \quad (5)$$

Using this PDF, the cumulative distribution function (CDF) of a Cauchy distribution with location μ and scale σ is calculated:

$$F(x; \mu, \sigma) = \frac{1}{\pi} \arctan \left(\frac{x-\mu}{\sigma} \right) + \frac{1}{2} \quad (6)$$

To sample from a Cauchy distribution, inverse transform sampling will be used as following:

[A]: Sample a point u_i from the uniform distribution defined in the interval $[0, 1]$ denoted as $u_i \sim \mathcal{U}(0, 1)$.

[B]: Calculate $c_i = F^{-1}(u_i; \mu, \sigma) = \sigma \tan(\pi(u_i - \frac{1}{2})) + \mu$. Then, c_i is a sample from the Cauchy distribution with location μ and scale σ .

6.2.3 Filtering

Let B denote the number of bootstrap samples of the filtering procedure and $w_k^{(i),b}$ denote the unnormalized weight of the particle i at time k for the bootstrap sample b and $W_k^{(i),b}$ the normalized one respectively. Let $a_{k,j}^{(i),b}$ denote the value of the particle i for the coefficient j at time k for the bootstrap sample b . Let ESS denote efficient sample size and $[\alpha, \beta]$ denote the interval for uniform initialization of the Cauchy scale parameters. Let \mathbf{Q}_q denote the covariance matrix of the normal distribution of the scale parameters of the Cauchy distributions of the coefficients, as defined in 6.2.1. The posterior density for a given bootstrap sample is approximated by $\hat{p}(\mathbf{a}_k | z_{1:k}) = \sum_{i=1}^N W_k^{(i)} \delta_{\mathbf{a}_k^{(i)}}(\mathbf{a}_k)$ and the autoregressive coefficients are approximated by the weighted mean of the filtered density. δ denotes Dirac delta mass. The algorithm is described below.

```

for  $b=1:B$  do
    Initialization:  $\mathbf{a}_0^{(i),b} = aryule(Z, p)$ ,  $W_0^{(i),b} = \frac{1}{N}$ ,  $\mathbf{q}_0^{(i),b} = diag(\mathcal{U}(\alpha, \beta))$ 
    Filtering: for  $k=1:T$  do
        Sample  $\mathbf{a}_{k-1+\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
         $\mathbf{q}_{k-1+\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1}^{(i),b}, \mathbf{Q}_q\Delta)$ 
         $\mathbf{a}_{k-1+2\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
         $\mathbf{q}_{k-1+2\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$ 
         $\vdots$ 
         $\mathbf{a}_k^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
         $\mathbf{q}_k^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$ 
         $w_k^{(i),b} = W_{k-1}^{(i),b} p(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R) = W_{k-1}^{(i),b} \mathcal{N}(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R)$ 
         $W_k^{(i),b} = \frac{w_k^{(i),b}}{\sum_{i=1}^N w_k^{(i),b}}$ 
         $ESS_k^b = \frac{1}{\sum_{i=1}^N (W_k^{(i),b})^2}$ 
        if  $ESS_k^b < \frac{N}{3}$  then
            Resample with replacement  $N$  particles from  $\{\mathbf{x}_k^{(i),b}\}_{k=1}^N$  according to
            probabilities  $W_k^{(i),b}$ .
            Reset  $W_k^{(i),b} = \frac{1}{N}$ 
        end
         $\hat{a}_{k,j}^b = \sum_{i=1}^N a_{k,j}^{(i),b} W_k^{(i),b}$  (Weighted Mean of filtered density)
    end
end
 $\hat{a}_{k,j} = \frac{1}{B} \sum_{b=1}^B \hat{a}_{k,j}^b$ 

```

Algorithm 1: Hybrid Sequential Importance Resampling

6.2.4 Likelihood of the model

Let $\boldsymbol{\theta}$ denote the scale parameters q_j . The log-likelihood that the observed data are generated by the model is defined as $\mathcal{L}(\boldsymbol{\theta}) = \log \prod_{k=1}^N p(z_k | z_{k-1}, \dots, z_0, \boldsymbol{\theta})$. (7)

(7) can be approximated using $p(z_k | z_{k-1}, \dots, z_0, \boldsymbol{\theta}) = \int p(z_k | \mathbf{x}_k) p(x_k | (z_{k-1}, \dots, z_0, \boldsymbol{\theta})) d\mathbf{x}_k \approx \frac{1}{N} \sum_{i=1}^N p(z_k | t_k^{(i)}) = \frac{1}{N} \sum_{i=1}^N w_k^{(i)}$ (8), where $t_k^{(i)}$ denotes the particle i at time k .

Using (8), (7) can be approximated as $\mathcal{L} = \sum_{k=1}^T \log(\sum_{i=1}^N w_k^{(i)})$ (9), where $w_k^{(i)}$ is the unnormalized weight of particle i at time k .

The approximated likelihood (9) will be used for model selection with the AIC and for model comparison between Normal and Cauchy models.

6.3 Results on simulated Data

The algorithm was run on the sinusoidal noisy signal with stepwise frequency modulation that was used in section 3.2, to simulate both abrupt and stationary regimes. $B=10$ realizations were done, the scale parameters were uniformly initialized in $[10^{-4}, 10^{-5}]$ and $N=500$ particles were used. Covariance matrix of the random walk of the scale parameters was set to $\mathbf{Q}_q = 10^{-5} \mathbf{I}$. Results are shown in Figure 6.2.

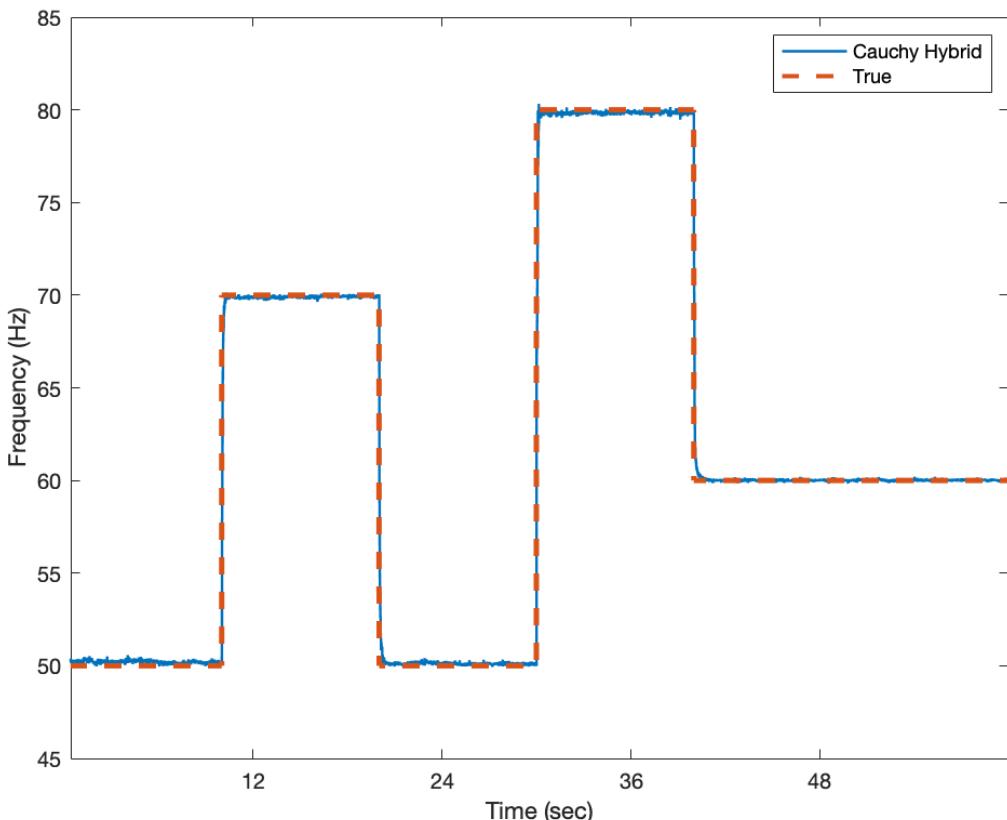


Figure 6.2: Instantaneous frequency tracking of the Cauchy model

The instantaneous frequency estimation for the hybrid Cauchy model is much closer to the ground truth in comparison to the discrete and hybrid Gaussian estimates as shown in Figure 5.10.

Mean square errors (MSE) shown in the next table corroborate this fact. Average value and standard deviation are shown in Table 6.1 for 100 runs. Results indicate that the self-organizing hybrid Cauchy model is closer to the truth instantaneous frequency. Results are further verified when comparing the log-likelihood values: $(1.6630 \pm 0.085) \cdot 10^4$ and $(1.3169 \pm 0.0567) \cdot 10^4$ for the Cauchy and Normal models respectively.

	Discrete	Hybrid
Normal	$(3.7836 \pm 0.0117) \cdot 10^4$	$(2.6229 \pm 0.0075) \cdot 10^4$
Self-Organizing Cauchy	$(1.8685 \pm 0.0245) \cdot 10^4$	$(1.3162 \pm 0.008) \cdot 10^4$

Table 6.1: Mean Square Errors of instantaneous frequency estimates for Gaussian and Cauchy models

6.4 Propofol Spectrograms

The hybrid Cauchy model was fit to the same propofol data as the normal ones. $B=10$ realizations were done, the scale parameters were uniformly initialized in $[10^{-4}, 10^{-6}]$ and $N=500$ particles were used. Covariance matrix of the random walk of the scale parameters was set to $\mathbf{Q}_q = 10^{-5} \mathbf{I}$. Results are shown below (Figure 6.3).

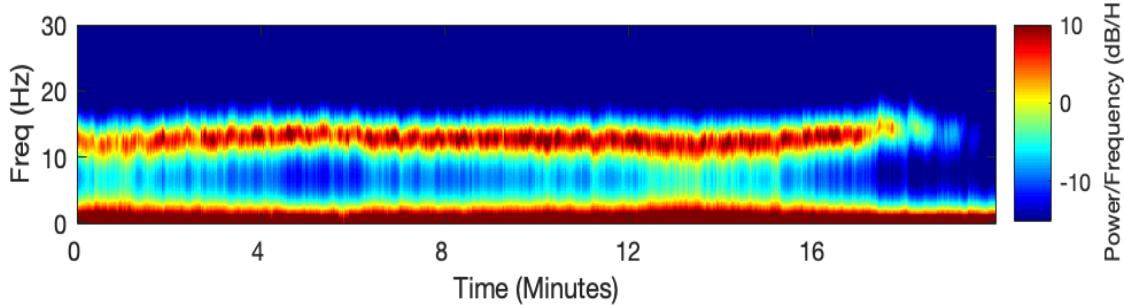


Figure 6.3: Hybrid Cauchy Spectrogram of EEG under propofol

Comparing Figures 4.5 and 6.3 (hybrid Normal and hybrid Cauchy spectrograms respectively), no noticeable difference is visible with the naked eye. To quantify the difference between the 2 spectrograms, a spectrogram showing the difference in frequency power between the two methods is shown in Figure 6.4.

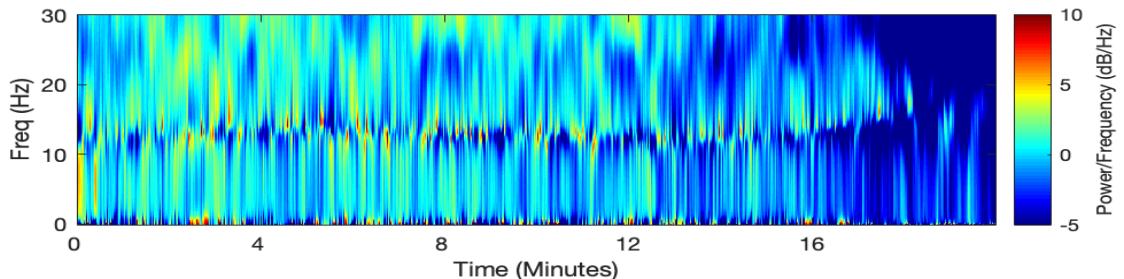


Figure 6.4: Difference in power estimated by the Normal and Cauchy models for propofol

Differences are mainly observed in the α frequency band. This indicates that the Cauchy model is able to identify slight changes across the band that the Normal model misses. This also reflects in the smoothness metric: $2.3785 \cdot 10^{-6}$ for the Cauchy model, between the values for hybrid and discrete Normal models. In other words, the hybrid Cauchy model is an intermediate between discrete and hybrid Gaussian models, avoiding the noise of the discrete while not oversmoothing like in the hybrid Normal case.

Finally, it is interesting to compare the model selection results for the Gaussian and Cauchy cases. For Gaussian, model $p=14$ was chosen, same as in the Cauchy case. AIC values for orders 4-50 are shown in Figure 6.5.

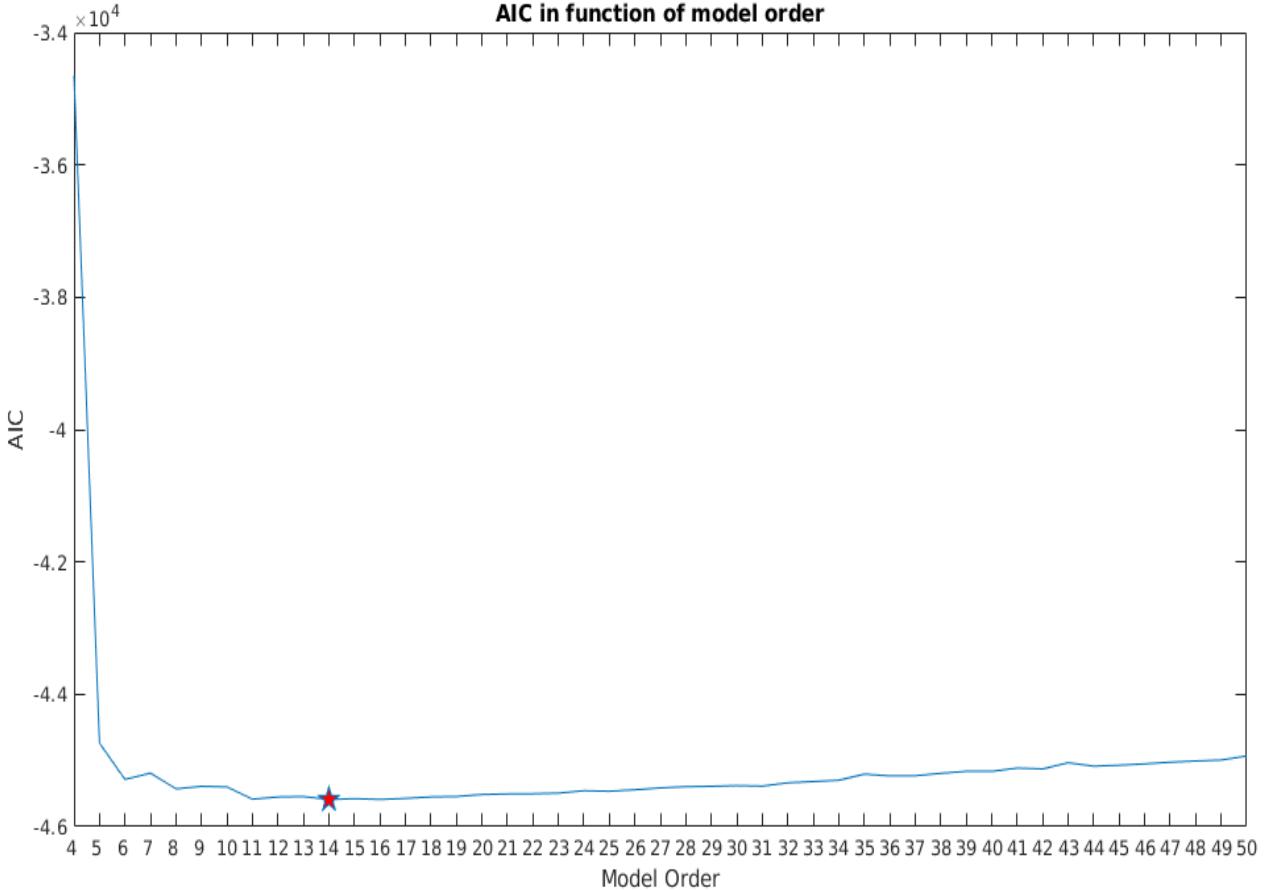


Figure 6.5: AIC for Cauchy model

6.5 Ketamine Spectrograms

Ketamine spectrogram generated by the hybrid Cauchy model is shown in Figure 6.6. $B=10$ realizations were done, the scale parameters were uniformly initialized in $[10^{-4}, 10^{-6}]$, the autoregressive model order was $p = 11$ and $N=500$ particles were used. Covariance matrix of the random walk of the scale parameters was set to $\mathbf{Q}_q = 10^{-4} \mathbf{I}$.

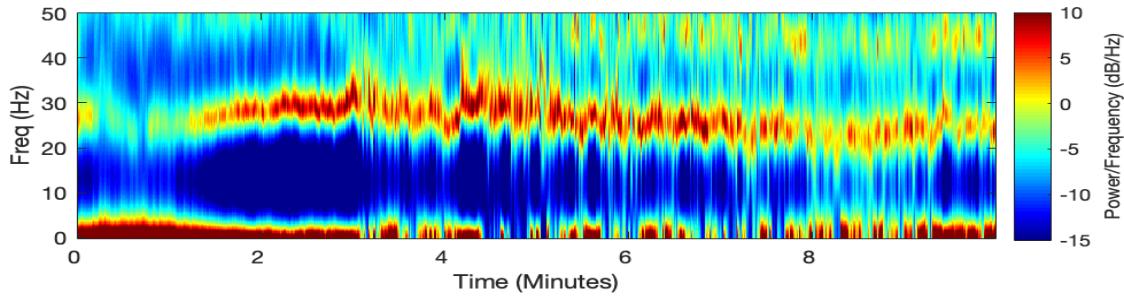


Figure 6.6: Hybrid Cauchy spectrogram of EEG under ketamine 1

To quantify the difference between the spectrograms of Figures 4.10 and 6.6 (hybrid Normal and hybrid Cauchy spectrograms respectively), a spectrogram showing the difference in frequency power between the two methods is shown in Figure 6.7.

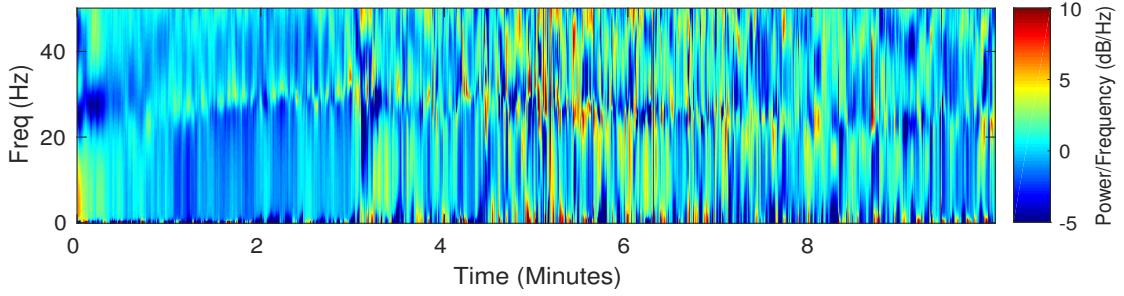


Figure 6.7: Difference in power estimated by the Normal and Cauchy models for ketamine 1

Differences are observed in the higher frequency bands. This is an indication that the increased adaptability of the Cauchy model allows it to identify more subtle frequency changes in those bands.

Comparison of the two models was also performed on the spike data of Figure 4.11. All model parameters were the same as in the previous case, with only difference the model order $p = 18$ and the uniform initialization of the Cauchy scales in $[10^{-3}, 10^{-4}]$ to allow for faster tracking of the spikes. $B = 10$ bootstrap samples were used. Results and differences in power between Normal and Cauchy model are shown in Figures 6.8 and 6.9 respectively.

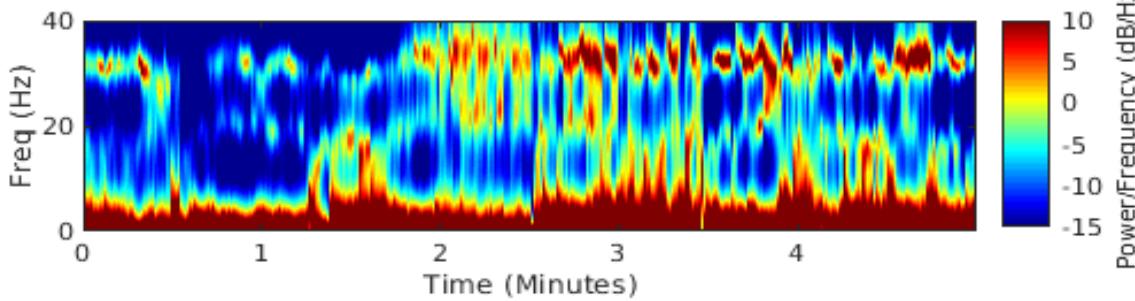


Figure 6.8: Hybrid Cauchy spectrogram of EEG under ketamine 2

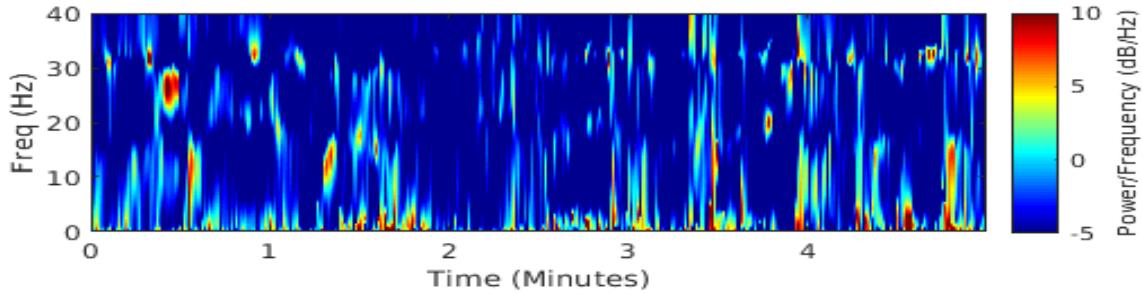


Figure 6.9: Difference in power estimated by the Normal and Cauchy models for ketamine 2

The effect of the Cauchy model on the ketamine data with abrupt spikes is also quite profound, as expected since the use of Cauchy noise allows for better tracking of the spikes. Differences are observed across the higher frequency bands, especially at the times of intense spiking activity after minute 3 until the end of the recording.

6.6 Discussion

The hybrid Cauchy model is able to better identify frequency changes, abrupt or smooth, in the simulated data in comparison to its Gaussian counterpart. When applied to propofol and ketamine EEG data, differences become apparent when the spectrogram of the difference in frequency power is calculated.

Apart from the insight from the simulated data that the Cauchy model is closer to the true instantaneous frequency of the EEG data, log-likelihood values were calculated for the two hybrid models. Results are shown in the following Table. They are the average values for the log-likelihood of $B=100$ bootstrap realizations. For ketamine, the first column refers to the data of Figure 4.6 and the second one to the data of Figure 4.11.

	Propofol	Ketamine
Normal	$(4.0854 \pm 0.03485) \cdot 10^5$	$(8.3613 \pm 0.98) \cdot 10^4 - (3.7458 \pm 0.1183) \cdot 10^3$
Self-Organizing Cauchy	$(4.3261 \pm 0.07536) \cdot 10^5$	$(1.0306 \pm 0.0083) \cdot 10^5 - (6.4073 \pm 0.2542) \cdot 10^3$

Table 6.2: Log-likelihood values for Gaussian and Cauchy hybrid models

Indeed, the log-likelihood for the Cauchy model is higher, which is a further indication that the Cauchy model is better in identifying the temporal evolution of the frequency bands of the EEG. The difference between the likelihoods is bigger for ketamine, indicating that using the Cauchy model in EEG under ketamine makes a bigger difference than using it for propofol EEG. This is expected, as tracking the frequency evolution of EEG under ketamine is more demanding, with more abrupt frequency changes due to the presence of higher frequency bands.

Concerning the effect of the self-organizing state-space model, when an abrupt change is detected in the EEG, the variance of each coefficient is adaptively updated to accommodate for this change. This way the model becomes even more potent in frequency tracking. This becomes very clear in the case of the simulated data where at the times of the abrupt changes of the frequency of the sinusoid, the scale parameters of the Cauchy distributions of the autoregressive parameters transition immediately to higher values, while in the stationary regimes their values tend to become lower (Figure 6.10). This is a result of the fact that particles whose values are more compatible with the observations tend to have sampled also variances of the Cauchy scales that are more compatible. In the case of abrupt changes, a particle with high variances of the Cauchy scales is able to change its values fast also for the autoregressive coefficients. So, the higher sampled variance provided the particle with the advantage of being able to quickly change its proposed values for the autoregressive coefficients and thus obtain a higher importance weight.

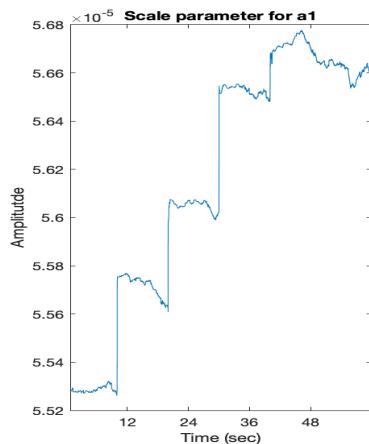


Figure 6.10: Temporal evolution of the scale of the Cauchy distribution of the autoregressive parameter

Overall, the advantage of the Cauchy model is that it allows higher adaptability than its Gaussian counterpart without introducing extra noise. Even when some outliers occur because of not properly weighing the corresponding particle, the use of bootstrap overcomes this issue, as it is very improbable that the outliers will occur at the same time points across realizations and thus they are averaged out.

6.7 Improving estimates using a local implicit Markov Chain Monte Carlo algorithm

Using the self-organized state-space formulation, a variant of the algorithm of section 6.2.3 is described here. A local Gaussian random walk is employed as a Markov Chain Monte Carlo (MCMC) method using the Metropolis-Hastings algorithm to improve local estimates. More specifically, between 2 subsequent observations, after the Cauchy random walk step has been done and the weights of the particles have been accordingly adjusted, the particles make L steps in the $2p$ -dimensional space for that given time point in an attempt to locally optimize the parameters' estimates. This local search further increases the log-likelihood and thus helps obtain more accurate estimates of the autoregression.

Running the algorithm on the simulated data with the stepwise frequency modulation produced a mean square error of $(1.2607 \pm 0.0094) \cdot 10^4$, while when run on propofol it produced a log-likelihood value of $(4.3997 \pm 0.05538 \cdot 10^5)$. In both cases, the metrics are better for the model with the local search. This is the result of locally searching for more compatible estimates for the parameters. Propofol spectrogram is shown below (Figure 6.11). The covariance matrix of the local random walks was manually set to $10^{-4} \mathbf{I}$. $L = 100$ steps of the MCMC were made at each time step in search for a more optimal local estimation. The scale parameters were uniformly initialized in $[10^{-4}, 10^{-6}]$ and $N=500$ particles were used. Covariance matrix of the random walk of the scale parameters was set to $\mathbf{Q}_q = 10^{-5} \mathbf{I}$.

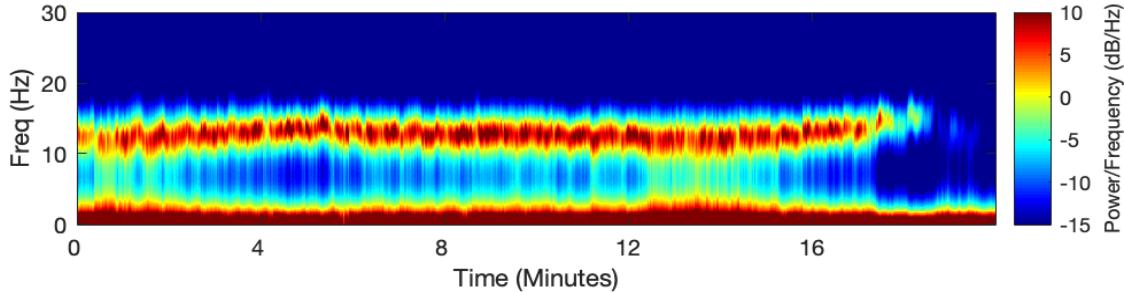


Figure 6.11: Hybrid Cauchy local MCMC Spectrogram of EEG under propofol

Differences in power are found between the non-locally optimized version of the algorithm as described in section 6.2.3. (Figure 6.12). Differences across the α frequency band are less prominent than the differences when the spectrograms of the hybrid Normal with the hybrid Cauchy were compared. This is expected since the difference in the log-likelihoods is also less ($4.0854 \cdot 10^5$, $4.3261 \cdot 10^5$ for Normal and Cauchy respectively). In other words, while local optimization helps the model identify even more subtle changes, the advantage of using a Cauchy random walk to using a Normal random walk for the parameters is even bigger. This comes at an increased computational cost, which is a result of the local search procedure. The algorithm is described in the next page in detail.

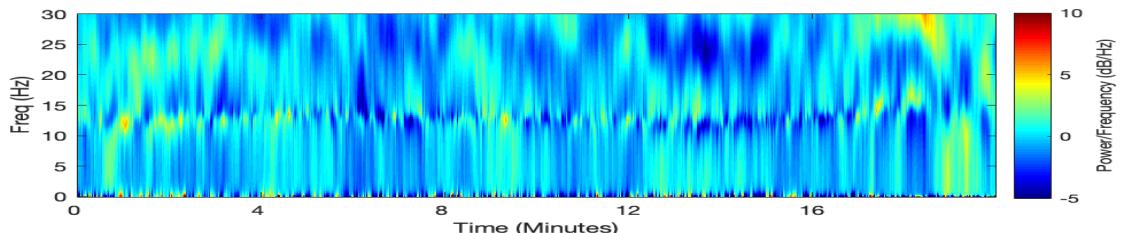


Figure 6.12: Difference in power estimated by the Cauchy SMC and Cauchy local MCMC models for propofol

Let L denote the number of steps of the MCMC that the algorithm makes at each time point of a new observation, \mathbf{Q}_{MCMC} the diagonal covariance of the random walk at the times of the observations and \mathcal{L} the log-likelihood.

```

for  $b=1:B$  do
  Initialization:  $\mathbf{a}_0^{(i),b} = aryule(Z, p)$ ,  $W_0^{(i),b} = \frac{1}{N}$ ,  $\mathbf{q}_0^{(i),b} = diag(\mathcal{U}(\alpha, \beta))$ 
  Filtering: for  $k=1:T$  do
    Sample  $\mathbf{a}_{k-1+\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
     $\mathbf{q}_{k-1+\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1}^{(i),b}, \mathbf{Q}_q\Delta)$ 
     $\mathbf{a}_{k-1+2\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
     $\mathbf{q}_{k-1+2\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$ 
     $\vdots$ 
     $\mathbf{a}_k^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
     $\mathbf{q}_k^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$ 
     $w_k^{(i),b} = W_{k-1}^{(i),b} p(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R) = W_{k-1}^{(i),b} \mathcal{N}(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R)$ 
     $W_k^{(i),b} = \frac{w_k^{(i),b}}{\sum_{i=1}^N w_k^{(i),b}}$ 
     $ESS_k^b = \frac{1}{\sum_{i=1}^N (W_k^{(i),b})^2}$ 
    if  $ESS_k^b < \frac{N}{3}$  then
      Resample with replacement  $N$  particles from  $\{\mathbf{x}_k^{(i),b}\}_{k=1}^N$  according to
      probabilities  $W_k^{(i),b}$ . Reset  $W_k^{(i),b} = \frac{1}{N}$ 
    end
    Set  $\tilde{\mathbf{a}}_k^{(i),b}(0) = \mathbf{a}_k^{(i),b}$ ,  $\hat{a}_{k,j}^b = \sum_{i=1}^N a_{k,j}^{(i),b} W_k^{(i),b}$ ,  $\tilde{\mathbf{q}}_k^{(i),b}(0) = \mathbf{q}_k^{(i),b}$ ,  $\hat{q}_{k,j}^b = \sum_{i=1}^N q_{k,j}^{(i),b} W_k^{(i),b}$ 
     $\tilde{\mathcal{L}}(0) = \sum_{t=1}^k \log(\sum_{i=1}^N w_t^{(i),b})$ ,  $\tilde{w}_k^{(i),b}(0) = w_k^{(i),b}$ 
    for  $l=1:L$  do
       $\tilde{\mathbf{q}}_k^{(i),b}(l) \sim \mathcal{N}(\tilde{\mathbf{q}}_k^{(i),b}(l-1), \mathbf{Q}_{MCMC})$   $\tilde{\mathbf{a}}_k^{(i),b}(l) \sim \mathcal{N}(\tilde{\mathbf{a}}_k^{(i),b}(l-1), \mathbf{Q}_{MCMC})$ 
       $\tilde{w}_{temp}^{(i),b}(l) = \tilde{w}_k^{(i),b}(l-1) \times$ 
       $\times p(z_k | \mathbf{H}_k, \tilde{\mathbf{a}}_k^{(i),b}(l), \tilde{\mathbf{q}}_k^{(i),b}(l), R) / p(z_k | \mathbf{H}_k, \tilde{\mathbf{a}}_k^{(i),b}(l-1), \tilde{\mathbf{q}}_k^{(i),b}(l-1), R)$ 
       $\tilde{\mathcal{L}}_{temp}(l) = \tilde{\mathcal{L}}(l-1) - \log(\sum_{i=1}^N \tilde{w}_k^{(i),b}(l-1)) + \log(\sum_{i=1}^N \tilde{w}_{temp}^{(i),b}(l))$ 
       $W_{temp}^{(i),b} = \frac{\tilde{w}_{temp}^{(i),b}}{\sum_{i=1}^N \tilde{w}_{temp}^{(i),b}}$ 
      With probability  $\min(1, \tilde{\mathcal{L}}_{temp}(l) / \tilde{\mathcal{L}}(l-1))$  set  $\hat{a}_{k,j}^b = \sum_{i=1}^N a_{k,j}^{(i),b} W_{temp}^{(i),b}(l)$ ,
       $\tilde{\mathcal{L}}(l) = \tilde{\mathcal{L}}_{temp}(l)$ ,  $W_k^{(i),b} = W_{temp}^{(i),b}$ 
      Otherwise set  $\tilde{\mathbf{a}}_k^{(i),b}(l) = \tilde{\mathbf{a}}_k^{(i),b}(l-1)$ ,  $\tilde{\mathcal{L}}(l) = \tilde{\mathcal{L}}(l-1)$ 
    end
  end
  end
   $\hat{a}_{k,j} = \frac{1}{B} \sum_{b=1}^B \hat{a}_{k,j}^b$ 

```

Algorithm 2: Hybrid Sequential Importance Resampling with local MCMC optimization

6.8 Improving estimates using a Metropolis-Hastings Sequential Markov Chain Monte Carlo algorithm

In the SMC algorithms of the previous sections, the estimates of the coefficients were approximated by the weighted mean of the filtered posterior density. This is a simple, reasonable choice but does not always lead to the highest likelihood values. An alternative approach is to construct a global Markov Chain and perform updates using the approximated posterior distribution as a proposal density for the Metropolis-Hastings update. Allowing the algorithm to explore the space by using a high value for the MCMC steps is computationally expensive but it can lead the parameters to areas of higher likelihood than the strategy of using the weighted mean, as the chance to sample a state of high likelihood becomes higher with a bigger number of steps. A major issue with this algorithm is that due to the quite high dimensional space many steps are required for the MCMC, a fact that renders the algorithm impractical for online use.

Propofol spectrograms and differences in power between SMC and SMCMC models are shown in Figures 6.13 and 6.14. L=100 steps of MCMC were used for each transition.

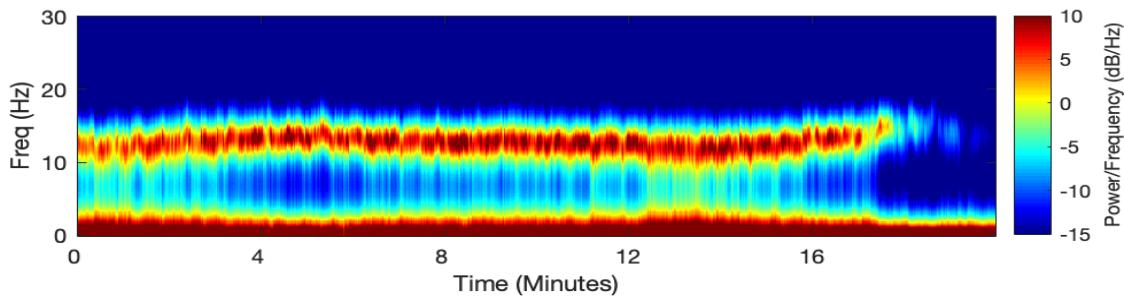


Figure 6.13: Hybrid Cauchy MCMC Spectrogram of EEG under propofol

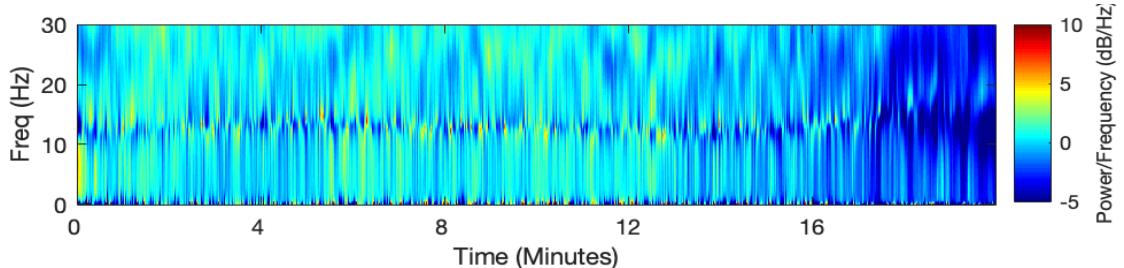


Figure 6.14: Difference in power estimated by SMC and SMCMC models for propofol

The log-likelihood value was calculated to be $(4.7468 \pm 0.1472) \cdot 10^5$. Bigger differences are observed than in the case of SMC vs. local SMCMC, as expected due to the bigger difference in the likelihood values. Using more MCMC steps could potentially lead to even higher log-likelihood values and thus more accurate estimates.

Overall, this algorithm can provide very accurate estimates of the autoregressive coefficients but it is computationally very intensive. It can be useful for offline analysis of EEG data.

It is described in detail on the next page.

for $b=1:B$ **do**

Initialization: $\mathbf{a}_0^{(i),b} = aryule(Z, p)$, $W_0^{(i),b} = \frac{1}{N}$, $\mathbf{q}_0^{(i),b} = diag(\mathcal{U}(\alpha, \beta))$

Filtering: **for** $k=1:T$ **do**

 Sample $\mathbf{a}_{k-1+\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$

$\mathbf{q}_{k-1+\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1}^{(i),b}, \mathbf{Q}_q\Delta)$

$\mathbf{a}_{k-1+2\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$

$\mathbf{q}_{k-1+2\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$

\vdots

$\mathbf{a}_k^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$

$\mathbf{q}_k^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$

$w_k^{(i),b} = W_{k-1}^{(i),b} p(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R) = W_{k-1}^{(i),b} \mathcal{N}(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R)$

$W_k^{(i),b} = \frac{w_k^{(i),b}}{\sum_{i=1}^N w_k^{(i),b}}$

$ESS_k^b = \frac{1}{\sum_{i=1}^N (W_k^{(i),b})^2}$

if $ESS_k^b < \frac{N}{3}$ **then**

 Resample with replacement N particles from $\{\mathbf{x}_k^{(i),b}\}_{k=1}^N$ according to

 probabilities $W_k^{(i),b}$

 Reset $W_k^{(i),b} = \frac{1}{N}$

end

 Approximate $p(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k})$ by $\hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k}) = \sum_{i=1}^N W_k^{(i)} \delta_{\mathbf{a}_k^{(i)}}(\mathbf{a}_k)$

 Sample $\tilde{\mathbf{q}}_k^{(i),b}(0) \sim \hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k}), \tilde{\mathbf{a}}_k^{(i),b}(0) \sim \hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k}),$

$\tilde{\mathcal{L}}(0) = \sum_{t=1}^k \log(\sum_{i=1}^N w_t^{(i),b}), \tilde{w}_k^{(i),b}(0) = w_k^{(i),b}$

for $l=1:L$ **do**

$\tilde{\mathbf{q}}_k^{(i),b}(l) \sim \hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k}), \tilde{\mathbf{a}}_k^{(i),b}(l) \sim \hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k})$

$\tilde{w}_{temp}^{(i),b}(l) = \tilde{w}_k^{(i),b}(l-1) \times$

$\times p(z_k | \mathbf{H}_k, \tilde{\mathbf{a}}_k^{(i),b}(l), \tilde{\mathbf{q}}_k^{(i),b}(l), R) / p(z_k | \mathbf{H}_k, \tilde{\mathbf{a}}_k^{(i),b}(l-1),$

$\tilde{\mathbf{q}}_k^{(i),b}(l-1), R)$

$\tilde{\mathcal{L}}_{temp}(l) = \tilde{\mathcal{L}}(l-1)$

$-log(\sum_{i=1}^N \tilde{w}_k^{(i),b}(l-1)) + log(\sum_{i=1}^N \tilde{w}_{temp}^{(i),b}(l))$

 With probability $\min(1, \tilde{\mathcal{L}}_{temp}(l) \setminus \tilde{\mathcal{L}}(l-1))$, set $\hat{a}_{k,j}^b = \tilde{a}_{k,j}^{(i),b}(l)$,

$\tilde{w}_k^{(i),b}(l) = \tilde{w}_{temp}^{(i),b}(l)$

$W_k^{(i),b} = \frac{\tilde{w}_{temp}^{(i),b}(l)}{\sum_{i=1}^N \tilde{w}_{temp}^{(i),b}(l)}$,

$\tilde{\mathcal{L}}(l) = \tilde{\mathcal{L}}_{temp}(l)$

 Otherwise set $\tilde{a}_{k,j}^{(i),b}(l) = \tilde{a}_{k,j}^{(i),b}(l-1)$, $\tilde{\mathcal{L}}(l) = \tilde{\mathcal{L}}(l-1)$

end

end

end

$$\hat{a}_{k,j} = \frac{1}{B} \sum_{b=1}^B \hat{a}_{k,j}^b$$

Algorithm 3: Sequential Markov Chain Monte Carlo

6.9 Particle Parallelization using virtual particle resampling

All the algorithms described in this chapter are quite computationally expensive due to the facts that expensive operations (sampling, importance weights, resampling) have to be performed for each particle and that the number of particles is often high. As a result, an alternative parallelizable algorithm across particles could be beneficial.

The bootstrap procedure can be parallelized without any difficulties and can be easily implemented using MATLAB's *parfor* loop.

In theory, particle filtering can be parallelized for increased computational efficiency. In practice, an issue arises due to the need of the resampling step.

The resampling step is done to avoid sample degeneracy that happens after some steps of the sampling procedure, where the importance will be only on a few particles. Resampling is performed according to the weights the particles have at the time of the resampling and leads the particles to regions that have high likelihood. Nevertheless, resampling causes another issue: Many particles that have low weights end up not being resampled at all. To solve this problem, resampling is performed only when a metric called Effective Sample Size (ESS) is below a certain threshold. The necessity to calculate ESS at every iteration of the filtering procedure renders the parallelization infeasible, as it is a collective operation across all particles that cannot be done in parallel.

To circumvent this issue, an alternative is described here, based on virtual particle (VP) resampling applied on the hybrid self-organized state-space model described in section 6.1.

A resampling strategy that does not require the collective operation of calculating ESS has to be used. To this end, VP resampling, as proposed in [41] will be used. More specifically VP resampling works by treating each particle resampling separately. A set of virtual particles is generated for each particle and the one that gave the highest importance factor is chosen. Importance factor for a given particle is calculated by evaluating $\hat{p}(z_k|\hat{a}_k)$. The algorithm is described in detail in the next page. 2 variants are described, one simpler with the filtered mean of the posterior as the final estimate for the autoregressive parameters and one using the MCMC approach of section 6.8.

Let W denote the number of virtual particles to be generated for each particle and σ the standard deviation which determines how much the generated virtual particles will differ from the original one.

```

for  $b=1:B$  do
  Initialization:  $\mathbf{a}_0^{(i),b} = aryule(Z, p)$ ,  $W_0^{(i),b} = \frac{1}{N}$ ,  $\mathbf{q}_0^{(i),b} = diag(\mathcal{U}(\alpha, \beta))$ 
  Filtering: for  $k=1:T$  do
    Sample  $\mathbf{a}_{k-1+\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
     $\mathbf{q}_{k-1+\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1}^{(i),b}, \mathbf{Q}_q\Delta)$ 
     $\mathbf{a}_{k-1+2\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
     $\mathbf{q}_{k-1+2\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$ 
     $\vdots$ 
     $\mathbf{a}_k^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
     $\mathbf{q}_k^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$ 
     $w_k^{(i),b} = w_{k-1}^{(i),b} p(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R) = w_{k-1}^{(i),b} \mathcal{N}(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R)$ 
    for  $w=1:W$  do
      Sample  $r \sim \mathcal{U}(-1, 1)$ 
       $\hat{\mathbf{a}}_{k,w}^{(i),b} = \mathbf{a}_k^{(i),b} + \sigma r$ 
       $\hat{w}_{k,w}^{(i),b} = \hat{p}(z_k | \hat{\mathbf{a}}_{k,w}^{(i),b})$ 
      if  $\hat{w}_{k,w}^{(i),b} > w_k^{(i),b}$  then
         $\mathbf{a}_k^{(i),b} = \hat{\mathbf{a}}_{k,w}^{(i),b}$ 
         $w_k^{(i),b} = \hat{w}_{k,w}^{(i),b}$ 
      end
    end
     $W_k^{(i),b} = \frac{w_k^{(i),b}}{\sum_{i=1}^N w_k^{(i),b}}$ 
     $\hat{a}_{k,j}^b = \sum_{i=1}^N a_{k,j}^{(i),b} W_k^{(i),b}$ 
  end
end
 $\hat{a}_{k,j} = \frac{1}{B} \sum_{b=1}^B \hat{a}_{k,j}^b$ 

```

Algorithm 4: Parallel SMC using virtual particle resampling

```

for  $b=1:B$  do
    Initialization:  $\mathbf{a}_0^{(i),b} = aryule(Z, p)$ ,  $W_0^{(i),b} = \frac{1}{N}$ ,  $\mathbf{q}_0^{(i),b} = diag(\mathcal{U}(\alpha, \beta))$ 
    Filtering: for  $k=1:T$  do
        Sample  $\mathbf{a}_{k-1+\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
         $\mathbf{q}_{k-1+\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1}^{(i),b}, \mathbf{Q}_q\Delta)$ 
         $\mathbf{a}_{k-1+2\Delta}^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
         $\mathbf{q}_{k-1+2\Delta}^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$ 
         $\vdots$ 
         $\mathbf{a}_k^{(i),b} \sim \mathcal{C}(\mathbf{a}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_{k-1}\Delta)$ 
         $\mathbf{q}_k^{(i),b} \sim \mathcal{N}(\mathbf{q}_{k-1+(\frac{1}{\Delta f_s}-1)\Delta}^{(i),b}, \mathbf{Q}_q\Delta)$ 
         $w_k^{(i),b} = W_{k-1}^{(i),b} p(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R) = W_{k-1}^{(i),b} \mathcal{N}(z_k | \mathbf{H}_k, \mathbf{a}_k^{(i),b}, R)$ 
        for  $w=1:W$  do
            Sample  $r \sim \mathcal{U}(-1, 1)$ 
             $\hat{\mathbf{a}}_{k,w}^{(i),b} = \mathbf{a}_k^{(i),b} + \sigma r$ 
             $\hat{w}_{k,w}^{(i),b} = \hat{p}(z_k | \hat{\mathbf{a}}_{k,w}^{(i),b})$ 
            if  $\hat{w}_{k,w}^{(i),b} > w_k^{(i),b}$  then
                 $\mathbf{a}_k^{(i),b} = \hat{\mathbf{a}}_{k,w}^{(i),b}$ 
                 $w_k^{(i),b} = \hat{w}_{k,w}^{(i),b}$ 
            end
        end
         $W_k^{(i),b} = \frac{w_k^{(i),b}}{\sum_{i=1}^N w_i^{(i),b}}$ 
        Approximate  $p(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k})$  by  $\hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k}) = \sum_{i=1}^N W_k^{(i)} \delta_{\mathbf{a}_k^{(i)}}(\mathbf{a}_k)$ 
        Sample  $\tilde{\mathbf{a}}_k^{(i),b}(0) \sim \hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k})$ ,  $\tilde{q}_k^{(i),b}(0) \sim \hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k})$ 
         $\tilde{\mathcal{L}}(0) = \sum_{t=1}^k \log(\sum_{i=1}^N w_t^{(i),b})$ ,  $\tilde{w}_k^{(i),b}(0) = w_k^{(i),b}$ 
        for  $l=1:L$  do
             $\tilde{\mathbf{q}}_k^{(i),b}(l) \sim \hat{p}(\mathbf{a}_k^b, \mathbf{q}_k^b | z_{1:k})$ 
             $\tilde{w}_{temp}^{(i),b}(l) = \tilde{w}_k^{(i),b}(l-1) \times$ 
             $\times p(z_k | \mathbf{H}_k, \tilde{\mathbf{a}}_k^{(i),b}(l), \tilde{\mathbf{q}}_k^{(i),b}(l), R) / p(z_k | \mathbf{H}_k, \tilde{\mathbf{a}}_k^{(i),b}(l-1), \tilde{\mathbf{q}}_k^{(i),b}(l-1), R)$ 
             $\tilde{\mathcal{L}}_{temp}(l) = \mathcal{L}(l-1) - \log(\sum_{i=1}^N \tilde{w}_k^{(i),b}(l-1)) + \log(\sum_{i=1}^N \tilde{w}_{temp}^{(i),b}(l))$ 
            With probability  $\min(1, \tilde{\mathcal{L}}_{temp}(l) \setminus \tilde{\mathcal{L}}(l-1))$ , set  $\hat{a}_{k,j}^b = \tilde{a}_{k,j}^{(i),b}(l)$ ,
             $W_k^{(i),b} = \frac{\tilde{w}_{temp}^{(i),b}(l)}{\sum_{i=1}^N \tilde{w}_{temp}^{(i),b}(l)}$ ,
             $\tilde{w}_k^{(i),b}(l) = \tilde{w}_{temp}^{(i),b}(l)$ 
             $\tilde{\mathcal{L}}(l) = \tilde{\mathcal{L}}_{temp}(l)$ 
            Otherwise set  $\tilde{\mathbf{a}}_k^{(i),b}(l) = \tilde{\mathbf{a}}_k^{(i),b}(l-1)$ ,  $\tilde{\mathcal{L}}(l) = \tilde{\mathcal{L}}(l-1)$ 
        end
    end
     $\hat{a}_{k,j} = \frac{1}{B} \sum_{b=1}^B \hat{a}_{k,j}^b$ 

```

Algorithm 5: Parallel SMCMC using virtual particle resampling

Chapter 7

Conclusion & Future Research

In conclusion, this thesis has presented a hybrid approach to fitting autoregressive models to non-stationary EEG data. As it has been shown, hybrid models are able to produce smooth spectrograms without underfitting. Two different models have been described, one with Gaussian and one with Cauchy state noise respectively. Gaussian models allow for fast and computationally efficient model fitting procedure but tend to oversmooth. On the other hand, Cauchy hybrid models are able to identify more subtle temporal changes across frequency bands. Gaussian assumptions lead to closed-form solutions and allow the use of Kalman Filter that is perfectly suited for online use and does not impose the high computational burden of Monte Carlo simulations. For this reason both models are presented to allow for an optimal trade-off between the desired accuracy and use of resources one wants to allocate.

A potential extension of the hybrid Cauchy model would be to adapt the model to allow for non-diagonal terms in the covariance matrix with the scale parameters. In the case of the Normal model, estimating the covariance matrix via EM algorithm led to some advantages, as described in page 37. Thus, it is expected that for the Cauchy case, such an extension could be beneficial.

In this hybrid approach, no prior knowledge about the evolution of the parameters in the time intervals between the observations is assumed. As a result, the parameters are assumed to follow a continuous random walk to allow for high adaptability.

The proposed models can become even more potent in calculating frequency spectra if some prior knowledge about the parameters is available. Indeed, having an insight into the expected temporal evolution of the coefficients can be useful to determine an appropriate transition matrix that will be able to model their temporal changes as accurately as possible.

For example, a positive diagonal transition matrix could account for an expected growth in the values of the autoregressive coefficients that will help "guide" the model towards higher values.

On the other hand, predicting *a priori* how the parameters are expected to evolve is not a trivial issue and demands a fair amount of research and experience with EEG datasets and effects of anesthetics on the brain. Nevertheless, it is expected that incorporating such prior information can enhance the model's performance even more.

This way, even in the absence of observations in the intermediate intervals, parameter update will still be possible by the integration of the state and covariance differential equations. I believe that the full potential of the hybrid approach will then be reached: being able to update the estimates even in the absence of new observations by using prior knowledge of the expected parameters' evolution.

Appendix A

MATLAB Routines

As part of this thesis, a MATLAB toolset was developed for the spectral analysis of time-series. In this appendix, a fast description of the created routines and instructions about their use is given.

A.1 Normal State Noise

Routines

1. *HybridKalmanFilter*: Performs the forward pass of the hybrid Kalman Filtering procedure as described in equations (5)-(10) of chapter 2.
2. *HybridKalmanSmoother*: Performs the forward pass of the hybrid Kalman Filtering procedure as described in equations (11),(12).
3. *HybridKalmanEM*: Hybrid Kalman Filter adapted to output the variables necessary to perform the Expectation-Maximization algorithm.
4. *HybridEM*: Expectation-Maximization algorithm to yield optimal estimates for $\Theta = \{\mathbf{Q}, R\}$.
5. *HybridEMwithQ*: Expectation-Maximization algorithm to yield optimal estimates for $\Theta = \{\mathbf{Q}, R\}$, while at the same time calculating the complete log-likelihood \mathcal{Q} .
6. *Model Selection*: Routine to calculate optimal autoregressive order using AIC or BIC, based on results of *HybridEM*.
7. *HybridSpectrogram*: Routine to calculate the time-varying spectrogram of the input time-series.
8. *HybridKalmanGenerative*: Routine to generate ground truth observations for EM algorithm validation.
9. *HybridKalmanFilterOnline*: Hybrid Kalman Filter adapted to run an online EM routine to adaptively estimate $\Theta = \{\mathbf{Q}, R\}$.

10. *ExponentialSmoothing*: Routine that applies exponential smoothing to the estimates of the autoregressive coefficients.

Finally, a routine to combine all was created : *HybridAlgorithm*. In order to generate the spectrogram of a time-series, the user has to just input the time series and this routine applies the algorithm as described in section 2.8. The output is the time-varying spectral estimation of the time-series. Output also includes metrics including AIC scores for the different model orders and EM algorithm convergence. Finally, the temporal estimation of the autoregressive coefficients is plotted.

The user has the liberty to modify the following parameters.

Exponential smoothing of AR coefficients (default:Off)

Sampling rate (default: 250Hz)

Frequency resolution (default: 4 index/Hz)

Maximum of frequency range (default: 50 Hz)

Minimum of frequency range (default: 0.0001 Hz)

Maximum number of EM algorithm iterations (default:50)

Batch size to run EM on (default: 10 seconds)

Tolerance for EM convergence (default: 10^{-3})

Initial observation noise variance for EM iterations (default: 0.2)

Starting coefficient of diagonal covariance matrix for EM iterations (default: 1)

Minimum AR order to be considered in model selection (default: 2)

Maximum AR order to be considered in model selection (default: 20)

Information criterion (default: AIC)

Remove outliers or not (default: On)

Run online EM or not (default: Off)

Buffer size in case of online EM (default 5 min)

In conclusion, the user has to load the data and call the routine *HybridAlgorithm* with the data as input.

A.2 Cauchy State Noise

2 different versions of the model were created: One using the self-organizing state space formulation with adaptive estimation of the scale parameters of the Cauchy distributions of the autoregressive parameters and a simpler, computationally faster one with one constant Cauchy scale parameter for the autoregressive parameters.

Routines

1. *SMC_Hybrid_Spectrogram*: Generates a spectrogram of the input data based on sequential importance resampling.

2. *SO_SMC_Hybrid_Spectrogram*: Generates a spectrogram of the input data based on sequential importance resampling with adaptive estimation of the scale parameters.

Those 2 routines have as input arguments:

y: Univariate Time-series EEG data

p: Autoregressive model order

N: Number of particles for the SMC

R: Variance of Gaussian observation noise

c: Scale parameter of the Cauchy distribution of the AR parameters for *SMC_Hybrid_Spectrogram*

a,b: [a,b] interval to uniformly initialize the scale parameters for *SO_SMC_Hybrid_Spectrogram*

delta: Number of discretization steps between the observations

B: Number of Bootstrap realizations

For the local MCMC search algorithm the following routine was created.

3. *SO_SMC_MCMC_Local*: Returns estimates of locally optimized autoregressive coefficients. In addition to the arguments of *SO_SMC_Hybrid_Spectrogram*, this routine has the following ones.

c_random: Variance of the local random walk

L: Number of steps of MCMC at each time point

4. *SO_SMC_MCMC*: Returns estimates of optimized autoregressive coefficients through Metropolis-Hastings sampling. It has the same arguments as *SO_SMC_MCMC_Local*. In this case *L* refers to the number of steps of MCMC of the global Markov Chain.

Appendix B

Combining Empirical Mode Decomposition with low order Hybrid Autoregressive Models for propofol spectral decomposition

Empirical Mode Decomposition (EMD) is an empirical method that decomposes a time-series signal as a sum of oscillatory components, called Intrinsic Mode Functions (IMF). EMD has been extensively applied to EEG analysis. Applications include source localization, EEG analysis during anesthesia to compare effects of different anesthetics, measuring anesthesia depth [27], [28], [29] etc. Details about the EMD algorithm can be found in [30].

Here, EMD is applied as an efficient way to separate the raw EEG signal under propofol into 2 time-series: a slow oscillation and an α oscillation. Then 2 autoregressive models of order 2 are fit separately in the the 2 time-domain signals and their spectra are combined to yield the overall spectral estimation.

Overall, the algorithm is the following.

[1]: Decompose the raw EEG signal $Z = \{z_k\}_{k=1}^N$ into 10 IMFs using EMD.

[2]: Reconstruct the slow oscillation component and the α oscillation component by adding the IMFs whose mean frequency is in the corresponding band. This way, 2 separate signals, Z_{slow} and Z_α are obtained.

[3]: Fit 2 separate Autoregressive models of order 2 to Z_{slow} and Z_α using the EM algorithm and the hybrid Kalman Filter described in Chapter 2. This way, $A_{slow} = \{\mathbf{a}_k^{slow}\}_{k=1}^N$ and $A_\alpha = \{\mathbf{a}_k^\alpha\}_{k=1}^N$ are obtained.

[4]: Compute the frequency response of the 2 autoregressive models using:

$$H_{slow}(f, t) = \frac{\sqrt{R_{slow}}}{1 - \sum_{k=1}^p a_k^{slow}(t) e^{-i2\pi k \frac{f}{f_s}}}, \text{ and } H_\alpha(f, t) = \frac{\sqrt{R_\alpha}}{1 - \sum_{k=1}^p a_k^\alpha(t) e^{-i2\pi k \frac{f}{f_s}}}, \quad 0 \leq f \leq \frac{f_s}{2}$$

R_{slow} and R_α are set by the user.

[5]: Compute the final spectrogram using $S(f, t) = |H_{slow}(f, t) + H_\alpha(f, t)|^2$

Running the aforementioned algorithm on the raw EEG data under propofol of Fig. 4.1, the following spectrogram is obtained (Fig.B.1).

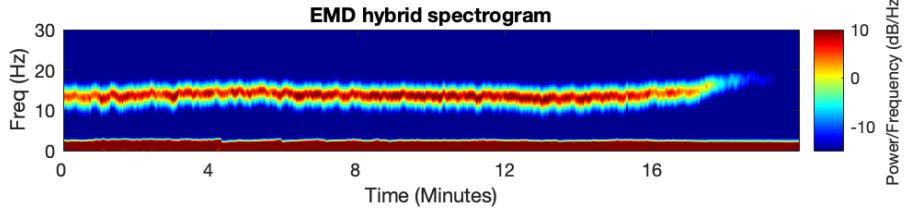


Figure B.1: EMD Hybrid Spectrogram of EEG under propofol

Observing the obtained spectrogram we can conclude that this method is able to efficiently track the slow and α frequency bands.

The main advantage of this method is that it avoids the computationally costly procedure of model selection, where models of different order have to be fit using EM and their respective AIC score has to be calculated. Indeed, it combines two autoregressive models of the lowest possible order ($p = 2$) that is required to represent a peak in the spectrum. In this propofol-oriented model, a priori knowledge about the expected frequency bands to appear is used: The IMFs that belong to the slow and α bands are summed together to yield the corresponding time domain signal. This is why in the spectrogram, power in frequencies outside the two aforementioned bands is very low in comparison to the hybrid spectrogram of Fig.4.5 where a hybrid autoregressive model of order 14 is fit to the raw data. This is a potential disadvantage of this method: It isolates the frequency bands of interest and calculates their temporal evolution. This could lead the model to miss power in the intermediate bands. For example, imagine a situation where the slow oscillation modulates a faster α oscillation. In this case, there would be side lobes along with the main lobe of the α frequency band which this parametric model would miss.

This is a trade-off that has to be made in comparison to the models of higher order that this thesis has investigated: Isolating the bands of interest with EMD and fitting low order autoregressive models can yield an efficient estimation of the temporal evolution of those bands. On the other hand, this method can miss potential interactions between the bands. An autoregressive model of higher order is capable of identifying all such interactions as well as the temporal evolution of the frequency bands of interest, but this comes at a higher computational cost.

In conclusion, an alternative method that uses the hybrid framework is presented here. This method is especially suited for propofol and focuses on estimating the temporal evolution of the clinically important slow and α frequency bands that occur in the EEG of patients under this specific anesthetic.

B.1 MATLAB Routine

In order to use the aforementioned algorithm, the MATLAB routine *EMD_Hybrid_Spectrogram* was created. This function takes as input the raw EEG time series and outputs the calculated spectrogram.

Bibliography

- [1] Thomson, D. J. (1982) "Spectrum estimation and harmonic analysis." Proceedings of the IEEE, 70, 1055–1096
- [2] Jones RH (1981). "Fitting a Continuous Time Autoregression to Discrete Data." Applied Time Series Analysis II, pp. 651–682.
- [3] A.C. Harvey & James H. Stock (1985). "The estimation of higher order continuous time autoregressive models" Econometric Theory, 1, 97-112.
- [4] Schlögl A, Flotzinger D, Pfurtscheller G. (1997) "Adaptive autoregressive modeling used for single-trial EEG classification." Biomed Tech (Berl). 1997 Jun;42(6):162-7.
- [5] Arnold et al. (1998). "Adaptive AR modeling of nonstationary time series by means of Kalman filtering." EEE Trans Biomed Eng. 1998 May;45(5):553-62.
- [6] Michael J. Cassidy and William D. Penny (2002) "Bayesian Nonstationary Autoregressive Models for Biomedical Signal Analysis." IEEE transactions on biomedical engineering, VOL. 49, NO. 10, OCTOBER 2002
- [7] Tarvainen et al. (2004) "Estimation of Nonstationary EEG With Kalman Smoother Approach: An Application to Event-Related Synchronization." IEEE transactions on biomedical engineering, VOL. 51, NO. 3, MARCH 2004
- [8] Nguyen et al. (2008). "Instantaneous Frequency and Amplitude Modulation of EEG in the Hippocampus Reveals State Dependent Temporal Structure." 2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society
- [9] Georgiadis et al. (2009) "Kalman smoother based time-varying spectrum estimation of EEG during single agent propofol anesthesia. 31st Annual International Conference of the IEEE EMBS Minneapolis, Minnesota, USA, September 2-6, 2009
- [10] Mohammad Emtyaz Khan, Deshpande Narayana Dutt (2007) "An Expectation-Maximization Algorithm Based Kalman Smoother Approach for Event-Related Desynchronization estimation from EEG." IEEE transactions on biomedical engineering, VOL. 54, NO. 7, July 2007 1191
- [11] Little, Roderick J.A.; Rubin, Donald B. (1987). Statistical Analysis with Missing Data. Wiley Series in Probability and Mathematical Statistics. New York: John Wiley Sons. pp. 134–136. ISBN 0-471-80254-9.
- [12] Maximum Likelihood from Incomplete Data via the EM Algorithm A. P. Dempster, N. M. Laird, D. B. Rubin Journal of the Royal Statistical Society. Series B (Methodological), Vol.

- [13] Z. Ghahramani and G. E. Hinton, "Parameter estimation for linear dynamical systems," Univ. Toronto, Dept. Comput. Sci, Toronto, ON, Canada, Tech. Rep., 1996.
- [14] Ramsay, James O., Hooker, Giles, Graves, Spencer (2009) "Functional Data Analysis with R and MATLAB", Springer, ISBN:978-0387981840
- [15] R.E. Kalman & R.S. Bucy (1961). "New Results in Linear Filtering and Prediction Theory" J. Basic Eng 83(1), 95-108
- [16] Belcher J, Hampton JS, Tunnicliffe Wilson G (1994). "Parameterization of Continuous Time Autoregressive Models for Irregularly Sampled Time Series Data." Journal of the Royal Statistical Society B, 56, 141–155.
- [17] Akaike H (1974). "A new look at the statistical model identification." IEEE Transactions on Automatic Control, 19(6), 716–723.
- [18] Patrik Axelsson & Fredrik Gustafsson (2015). "Discrete-Time Solutions to the Continuous-Time Differential Lyapunov Equation With Applications to Kalman Filtering." IEEE Transactions on Automatic Control (Volume: 60, Issue: 3, March 2015)
- [19] Yaakov Bar-Shalom, X. Rong Li, Thiagalingam Kirubarajan (2001). "Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software" Wiley, ISBN: 978-0-471-41655-5
- [20] S. M. Pandit & S. M. Wu (1975). "Unique Estimates of the Parameters of a Continuous Stationary Stochastic Process." Biometrika Vol. 62, No. 2 (Aug., 1975), pp. 497-501
- [21] Kim et al (2018). "State-space multitaper time-frequency analysis." Proc Natl Acad Sci U S A. 2018 Jan 2;115(1):E5-E14
- [22] Genshiro Kitagawa & Will Gersch (1996). "Smoothness Priors Analysis of Time Series." Springer, ISBN: 0-387-94819-8
- [23] Purdon PL, Sampson A, Pavone KJ, Brown EN. "Clinical Electroencephalography for Anesthesiologists: Part I: Background and Basic Signatures." Anesthesiology. 2015 Oct;123(4):937-60
- [24] Andrew H. Song, Sourish Chakravarty, Emery N. Brown (2018). "Smoothen State Space Multitaper Spectrogram." 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)
- [25] David Barber (2012). "Bayesian Reasoning and Machine Learning." Cambridge University Press, ISBN: 978-0521518147
- [26] Shumway & Stoffer (2011). "Time series analysis and its applications." Springer, ISBN: 978-1441978646
- [27] Hansen et al. (2019). "Unmixing Oscillatory Brain Activity by EEG Source Local-

ization and Empirical Mode Decomposition." Computational Intelligence and Neuroscience, Volume 2019, Article ID 5618303 <https://doi.org/10.1155/2019/5618303>

[28] Tsai et al. (2016). "Frontal electroencephalogram analysis with ensemble empirical mode decomposition during the induction of general anesthesia." Biomed. Phys. Eng. Express 2 (2016) 065004 doi:10.1088/2057-1976/2/6/065004

[29] Shih et al. (2015). "Instantaneous 3D EEG Signal Analysis Based on Empirical Mode Decomposition and the Hilbert–Huang Transform Applied to Depth of Anaesthesia." Entropy 2015, 17, 928-949; doi:10.3390/e17030928

[30] Mandic et al. (2013). "Empirical Mode Decomposition-Based Time-Frequency Analysis of Multivariate Signals." IEEE SIGNAL PROCESSING MAGAZINE , 10.1109/MSP.2013.2267931

[31] Beck et al. (2018). "State Space Oscillator Models for Neural Data Analysis." Conf Proc IEEE Eng Med Biol Soc. 2018 Jul;2018:4740-4743. doi: 10.1109/EMBC.2018.8513215

[32] Ting et al. (2010). "Spectral estimation of nonstationary EEG using particle filtering with application to event-related desynchronization (ERD)." IEEE Transaction on Biomedical Engineering, vol. 58. no. 3, pp. 321-331, 2011. DOI 10.1109/TBME.2010.2088396

[33] Murata & Hiramatsu (2017). "Non-Gaussian Filters for Nonlinear Continuous-Discrete Models." SICE Journal of Control, Measurement, and System Integration, 2017 Volume 10 Issue 2 Pages 53-61

[34] Kitagawa (1998). "A self-organizing state-space model." Journal of the American Statistical Association; Sep 1998; 93, 443; ABI/INFORM Collection pg. 1203

[35] Liu et al. (2014). "Parallel Resampling for Particle Filters on FPGAs." 2014 International Conference on Field-Programmable Technology ,978-1-4799-6245-7/14/

[36] Murray et al. (2015). "Parallel resampling in the particle filter." arXiv:1301.4019v3

[37] Andrieu et al. (2010) "Particle Markov chain Monte Carlo methods." J. R. Statist. Soc. B (2010) 72, Part 3, pp. 269–342

[38] Luhtala et al. (2016). "An Introduction to Twisted Particle Filters and Parameter Estimation in Non-Linear State-Space Models." IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 64, NO. 18, SEPTEMBER 15, 2016 4875

[39] Dahlin et al. (2019). "Getting Started with Particle Metropolis-Hastings for Inference in Nonlinear Dynamical Models." arXiv:1511.01707v8 12 Mar 2019

[40] Li et al. (2014). "Fight sample degeneracy and impoverishment in particle filters: A review of intelligent approaches." Expert Systems with Applications Volume 41, Issue 8, 15 June 2014, Pages 3944-3954

[41] Schwiegelshohn et al. (2016). "A resampling method for parallel particle filter architectures." Microprocessors and Microsystems 47 (2016) 314–320