LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

INSTITUT FÜR STATISTIK

Matthias Mittermayer

Detection and Tracking of Mobile Channel Impulse Responses

Master's Thesis
in cooperation with the German Aerospace Center (DLR)
Supervision: Prof. Dr. Volker Schmid and Dr. Thomas Jost
Department of Statistics – University of Munich

January 7, 2014

Detection and Tracking of Mobile Channel Impulse Responses

Abstract

The exchange of information in today's society requires for developing more powerful wireless transmission systems like positioning by global navigation satellite systems and mobile communications. However, the available frequency bands are limited with regard to the high data rates and an increased need for spectral efficiency will occur. Thus, a good understanding of the wireless channel is essential, where propagation paths' parameters show a time-variant behavior in terms of assuming a moving receiver. This thesis describes a novel algorithm which is able to both detect individual propagation paths and track how these paths evolve with time. A simulation based on a wireless transmission channel with a moving receiver is given, shows that the algorithm outperforms the standard snapshot-based algorithms in terms of identifying the number of individual paths and estimation accuracy. During the further procedure we assure the algorithm to be even superior to other tracking methods in cognition of individual propagation paths.

Acknowledgments

I would like to express my very sincere gratitude to Prof. Dr. Volker Schmid for the support to make this thesis possible. I am very thankful for his exemplary commitment and flexible supervision throughout the whole process of my work.

Furthermore, I would like to express my honest gratefulness to my adviser at German Aerospace Center DLR, Dr. Thomas Jost for the continuous support, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better adviser.

I would also like to thank my colleagues at the Institute of Communications and Navigation, especially D. Shutin, W. Wang and F. Antreich for fruitful discussions and suggestions to improve my implementation strategy.

Finally and most importantly, I would like to thank my friends and family for their support, their understanding and their endless encouragement throughout the whole years of my studies. Particularly I am deeply grateful to Rebecca for her unlimited belief and backing. I know that this is far from being the case.

Contents

1 Introduction

Signal processing is everywhere. It is one of the most powerful technologies that will shape science and engineering in the twenty-first century. The exchange of information in today's society requires to develop more powerful communication systems. Hereby wireless data transmission is one of the most important fields. Appropriate systems constitute a rapidly growing market, since they allow participants for high mobility. The focus of interest is both applications like positioning by global navigation satellite systems (GPS) and mobile communications. However, the available frequency bands are limited with regard to the high data rates and an increased need for spectral efficiency occurs. Thus, a good understanding of the wireless channel is essential for developing smart antennas, senders, and receivers in order to satisfy the ever-growing demand for fast information (Tschudin (1999)).
Recently, various high resolution methods have been proposed to estimate some of the parameters of impinging plane waves, i.e., their complex amplitude, delay, incidence azimuth, incidence elevation, Doppler frequency and the number of multipath components. It often leads to analytically intractable and computationally very expensive optimization procedures. The problem is often relaxed by assuming that the number of multipath components is fixed, which simplifies optimization in many cases. However, both underspecifying and overspecifying the model order leads to significant performance degradation (Shutin et al. (1997)), (Fleury et al. (1999)).
When we consider a mobile communication channel with a moving receiver it is easily to understand that the parameters will change over time, but not randomly. Taking this fact into account, this thesis deals with two challenges of multipath channel estimation: first, an efficient estimation scheme is considered which works on a snapshot basis able to estimate signal parameters and to perform model order detection. Second, since the parameters of consecutive time steps are highly correlated a method is desired that might include existing information to track multipath components over time.

This work is structured as follows: Chapter 2 addresses the wireless channel and the underlying signal model in the case of mobile channel impulse responses as an extended introduction. The main parts deal with the aforementioned challenges on the parameter estimation scheme. While chapter 3 will present a strategy for detection of all relevant components as well as estimation of their corresponding parameters for fixed snapshot times, chapter 4 is meant for tracking paths over time by the use of the so called Kalman Filter and Kalman Smoother. Finally I will close with a simulation study in order to validate the performance of the proposed algorithm in comparison to already existing implementations.

Throughout this thesis I will make use of the following notation:

- Vectors are represented as boldface lowercase letters e.g. $\boldsymbol{x}$ and are always considered as columns by default.
- Matrices are recognized as boldface uppercase letters, e.g. $\boldsymbol{X}$.
- For vectors and matrices $(\cdot)^T$ and $(\cdot)^H$ stands for the transpose and Hermitian transpose, while $(\cdot)^*$ represents the complex conjugate, respectively.
- An estimate of a random variable $\boldsymbol{x}$ is denoted as $\hat{\boldsymbol{x}}$.
- I use $\mathbb{E}_{p(\cdot)}\left(f(\boldsymbol{x})\right)$ to express the expectation of a function $f(\boldsymbol{x})$ with respect to $p(\cdot)$.
- $\mathbb{E}\left(f(\boldsymbol{x})\right)$ stands for the unconditioned expectation of $f(\boldsymbol{x})$.
- Finally, $x \sim \mathcal{N}\left(\mu, \sigma^2\right)$ and $x \sim \mathcal{CN}\left(\mu, \sigma^2\right)$ denotes a real and complex Gaussian distributed random variable x with mean μ and variance σ^2; $x \sim \mathcal{G}a\left(a, b\right)$ denotes a Gamma probability density function with parameters a and b.

2 The Wireless Channel

2.1 Characteristics of Propagation Paths

Wireless data transmission preserves conditions which are unique. A signal, as it travels through the wireless channel, undergoes many kinds of propagation effects such as reflection, diffraction, scattering, and shadowing, due to the presence of buildings, mountains and other obstructions. A graphical representation of possible phenomena can be seen in fig. 1. Good overview literature which I will refer to in the following is Mitra (2009). Reflection occurs when the waves impinge on objects which are much greater than the wavelength of the traveling wave, e.g., huge buildings, mountains, or the surface of the earth. The signal is reflected and is not as strong as the original, as objects may absorb some of the signal's power. However, reflection helps transmitting signals as soon as no Line-of-Sight (LoS) exists. This is the standard case for wireless data transmission in cities where transmitted signals may bounce off the walls of buildings several times before finally reaching the receiver. Diffraction is a phenomena occurring when the wave interacts with a surface having sharp irregularities and the reason why waves can propagated beyond the horizon, around the curved earth's surface and obstructions, f.e. tall buildings. Thus, the waves will be deflected at an edge and propagate in different directions. Scattering occurs when the medium through the wave is traveling contains objects which are much smaller than the wavelength of the considered wave, leading to a fragmentation of the signal into several weaker outgoing signals. Finally, shadowing occurs when signals get lost by being blocked by objects as mountains or walls.

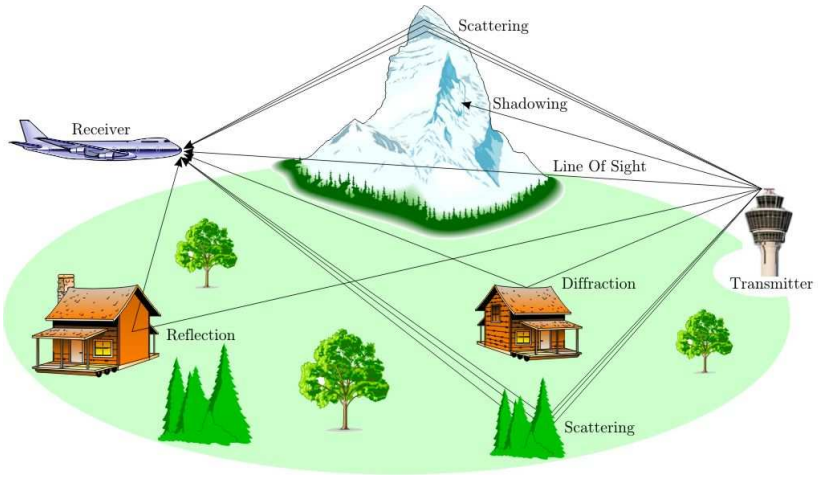


Figure 1: Visualization of propagation effects (Lien (2011))

2.2 The Signal Model

In this chapter we will lay down the basic theoretical background which is necessary to model the underlying multipath channel. The multipath effect occurs when a transmitted signal arrives at the receiver via multiple propagation paths. Due to the phenomena described in section 2.1 of this work, at a specific time point t, each path has a complex amplitude $w_l(t)$, a delay $\tau_l(t)$ and an azimuth angle of arrival $\phi_l(t)$, where we stack $\tau_l(t)$ and $\phi_l(t)$ into the vector $\boldsymbol{\theta}_l(t) = [\tau_l(t), \phi_l(t)]^T$. Generally the considered channel consists of an infinite number of components. Since we want to develop a strategy for detecting the number of paths as well as the corresponding parameters, the task is unsolvable. It was first Turin (1972), who extenuated the problem by assuming a fixed number $L(t)$ of relevant components, what we will also adopt here.

Fig. 2 is a graphical representation of a multipath channel consisting of three components. While the red line visualizes the LoS with its corresponding parameters $\boldsymbol{\theta}_1(t), w_1(t)$, the other two multipaths occur by reflection in this case. It is easily to understand that they differ in terms of delay and azimuth angle of arrival, summarized in $\boldsymbol{\theta}_2(t)$, $\boldsymbol{\theta}_3(t)$.

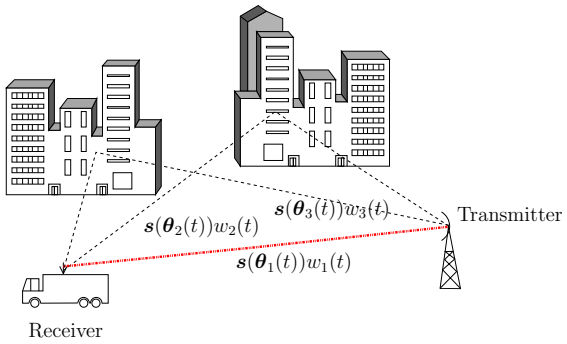


Figure 2: Graphical representation of a multipath channel consisting of three components

One simple and popular model to express the wireless channel is the time-variant channel impulse response denoted by the function $h(t, \tau)$. For all known paths, multipath can be considered as a series of delayed copies of the transmitted signal weighted by amplitudes $w_l(t)$, for $l = 1, ..., L(t)$. Thus, the received signal $\widetilde{y}(t)$ can be expressed in time domain by

$$\widetilde{y}(t) = b(t) * h(t, \tau) = \int_{-\infty}^{\infty} h(t, \tau) b(t - \tau) d\tau,$$

where $*$ denoting the convolution. The impulse response function may be considered as being the system response at time t to a unit impulse $\delta(t)$ applied in the past. If we assume that contributions with different delays are uncorrelated plus that the statistical properties of the channel do not change over time, we can write $h(t, \tau)$ as

$$h(t, \tau) = \sum_{l=1}^{L(t)} w_l(t)\delta(\tau - \tau_l(t)),$$

or redefined by the use of the Fourier transform to switch from time to frequency domain

$$H(t, f_m) = \sum_{l=1}^{L(t)} w_l(t) \exp\left(-j2\pi f_m \tau_l(t)\right),$$

where f_m represents the frequency of the m-th frequency bin, $m = 0, ..., M-1$.

In the following we will restrict the further process on Single-Input Multiple-Output (SIMO) inspection, i.e. one transmitter and multiple receiver antennas. Furthermore, we assume all elements to be placed on the x/y-plane, and we may only estimate one angle, i.e the azimuth angle of arrival $\phi_l(t)$. For the later implementation, we consider only linear antenna arrays. Nevertheless, the algorithms and models can be easily extended to consider also other structures that the elevation and azimuth angle may be estimated separately.

For further calculation we stack the received signals of all antennas into one vector, thus the number of total samples N is $M \times A$, with M denoting the frequency samples and A the number of antennas. The received signal $\boldsymbol{y}(t)$ can then be written in frequency domain as

$$\boldsymbol{y}(t) = [\boldsymbol{y}(t, f_0)^T, ..., \boldsymbol{y}(t, f_{M-1})]$$

with

$$\boldsymbol{y}(t, f_m) = \sum_{l=1}^{L(t)} u(\tau_l(t))\boldsymbol{c}(\phi_l(t))w_l(t) + \boldsymbol{\xi}(t). \tag{2.1}$$

Thereby, $\boldsymbol{\xi}(t)$ denotes a zero mean circular complex normal vector for the measurement noise, $u(\tau_l(t))$ stands for the frequency response and $\boldsymbol{c}(\phi_l(t))$ represents

the steering vector, where the latter two can be stated as

$$u(\tau_l(t)) = \exp(-j2\pi f_m \tau_l(t))$$
$$\boldsymbol{c}(\phi_l(t)) = \exp\left(-j2\pi \boldsymbol{p}_n \cos\left(\phi_l(t)\right)\right),$$

where $\boldsymbol{p}_n$ denoting the antenna positions.

Finally we define

$$u(\tau_l(t))\boldsymbol{c}(\phi_l(t)) := \boldsymbol{s}(\boldsymbol{\theta}_l(t)), \tag{2.2}$$

where $\boldsymbol{s}(\boldsymbol{\theta}_l(t))$ is a non-linear parametrized vector by delay $\tau_l(t)$ and azimuth angle of arrival $\phi_l(t)$.

Plugging in eq. (2.2) into eq. (2.1) results in the final form of the signal model in the SIMO case which we will deal with throughout the whole thesis.

$$\boldsymbol{y}(t) = \sum_{l=1}^{L(t)} \boldsymbol{s}(\boldsymbol{\theta}_l(t)) w_l(t) + \boldsymbol{\xi}(t) = \boldsymbol{S}(\boldsymbol{\Theta}(t))\boldsymbol{w}(t) + \boldsymbol{\xi}(t). \tag{2.3}$$

In what follows in the consecutive work, we want to set up a strategy to detect the number of relevant components as well as the parameter values for complex amplitude $w_l(t)$, delay $\tau_l(t)$, and azimuth angle of arrival $\phi_l(t)$ for each multipath l.

3 Detection and Estimation of Multipath Components

3.1 Introduction to Multipath Channel Estimation

In this chapter, our goal is to estimate the parameters of the underlying signal model (see eq. 2.3) in an adequate way. For the moment we restrict ourselves on estimation processes at fixed snapshot time, so the time index index t is ignored for the current consideration. We base our work on the so called Variational Bayesian Space-Alternating Generalized Expectation-Maximization (VB-SAGE) algorithm introduced in Shutin and Fleury (2011) and Shutin et al. (2013).
A received signal vector $\boldsymbol{y}$ which considers all antennas can be represented as a superposition of the unknown number L of multipath components and its signal parameters $w_l, \boldsymbol{\theta}_l$. In the assumed model, w_l denotes the complex amplitude as an unknown weight or gain for the component l. Furthermore $\boldsymbol{s}(\boldsymbol{\theta}_l)$ is a vector which is parametrized non-linear by the set $\boldsymbol{\theta}_l$ of dispersion parameters, e.g. delay τ_l and azimuth angle of arrival ϕ_l, thus $\boldsymbol{\theta}_l = [\tau_l, \phi_l]^T$. Finally, $\boldsymbol{\xi}$ represents a zero mean circular complex normal vector for the noise with covariance $\boldsymbol{\Sigma}$. All together:

$$\boldsymbol{y} = \sum_{l=1}^{L} \boldsymbol{s}(\boldsymbol{\theta}_l) w_l + \boldsymbol{\xi} = \boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w} + \boldsymbol{\xi}. \tag{3.1}$$

Clearly, the received signal $\boldsymbol{y}$ is observable, but neither the finite number L of multipath components nor the signal parameters belonging to one specific path l are known. The challenge of multipath channel estimation is to find adequate estimates for the complex amplitudes w_l, the parameters $\boldsymbol{\theta}_l$ and last but not least the model order L.
Experimental evidence shows that model order selection schemes not designed carefully, may lead to a wrong detection of the number of relevant propagation paths. In case of overestimating the number of paths, fictive components without any physical meaning will be introduced and their parameters estimated. Vice versa, if too less multipath components are detected, important information gets lost.
For a fixed model order the classical maximum likelihood approach to the estimation of parameters involves maximization of the multidimensional parameter likelihood $p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w})$ given the signal measurements $\boldsymbol{y}$. As mentioned before, $\boldsymbol{S}(\boldsymbol{\Theta})$ is parametrized non-linear by delay and azimuth angle of arrival for each path. For non-linear type of optimizations the estimation of these signal parameters has often been solved using Expectation-Maximization (EM) type of algorithms, due to the non-linearity of the assumed model with respect to the parameter set $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1^T, \cdots, \boldsymbol{\theta}_L^T]^T$. Unfortunately these methods are only applicable when the order L is known and fixed, a requirement that is rarely satisfied in practice.

Within the VB-SAGE framework a so called precision or sparsity parameter $\boldsymbol{\alpha}$ is introduced in Shutin and Fleury (2011) and Shutin et al. (2013) as a constraint on the model parameters, whose contribution is following: Including the precision parameter avoids a wrong detection of the number of relevant paths. The gains $\boldsymbol{w}$ are constrained using a Bayesian parametric prior $p(\boldsymbol{w}|\boldsymbol{\alpha}) \sim \mathcal{CN}(\mathbf{0}, \boldsymbol{\alpha}^{-1})$, which is a Gaussian probability density function with zero mean and precision parameters $\boldsymbol{\alpha}$ being inversely proportional to the width of the actual probability density function. Such form of the prior allows controlling the contribution of the weight w_l for each component l: a large value of α_l will drive the corresponding weight w_l towards zero (because of $\mathbb{E}(w_l) = 0$) what will result in identifying the component as a fictive one, which has to be removed within the parameter estimation process. Thus, the VB-SAGE includes a detection scheme which eliminates the drawback of standard EM-based algorithms that requires for pre-knowledge about the model order L for optimization, since we are now able to detect the number of relevant components within the framework.

The current chapter is organized as follows: Introductory it is illustrated how the classical Space-Alternating Generalized Expectation-Maximization algorithm (SAGE) works for fixed model order L and afterwards we proceed with setting up the VB-SAGE framework that builds on this EM-based estimation scheme. In the main part it is explained how the detailed estimation of multipath parameters is carried out. This should also include the precision parameter approach to identify the number of relevant components. Finally we close with a simulation study in order to evaluate the performance of the proposed algorithm.

3.2 Classical SAGE Approach as a Variant of EM

The EM algorithm is a well-known numerical procedure which can facilitate maximizing the log-likelihood function in cases of incomplete data. An EM algorithm iteratively alternates between an Expectation- and a Maximization-step. While the former one calculates the conditional expectation of the complete-data log-likelihood, the second one simultaneously maximizes that expectation with respect to all of the unknown parameters. The algorithm iterates back and forth, using the current parameter estimates to decompose the observed data better and thus increase the likelihood of the next parameter estimates. Under regularity conditions, the algorithm converges to a stationary point of the likelihood function where each iteration cycle increases the likelihood of the estimated parameters (Feder and Weinstein (1988)).

One of its variants, the SAGE algorithm (Fessler and Hero (1994)) updates the parameters sequentially in small groups - however it preserves the stability of EM.

Instead of estimating all parameters at once, each iteration of SAGE consists of C sets of parameters. The parameter subset c is then updated by maximizing the conditional expectation of the log-likelihood of the augmented data, corresponded to this subset (Chung and Boehme (2001)). In the simplest form SAGE allows even for one-dimensional optimization, what makes the algorithm easy to use. This advantage is the particular reason why SAGE is one of the widely-used parameter estimation frameworks in signal processing although it is not the answer to everything. SAGE as well as the classical EM is only applicable if the number of signal components L is known or pre-defined. For this reason the VB-SAGE algorithm is proposed which allows for including a model selection criteria to the classical SAGE approach.

3.3 Variational Bayesian Parameter Inference

Let's begin with the assumed signal model introduced in eq. (3.1). Including the precision parameter $\boldsymbol{\alpha}$ to the model it can be represented by the following graph.

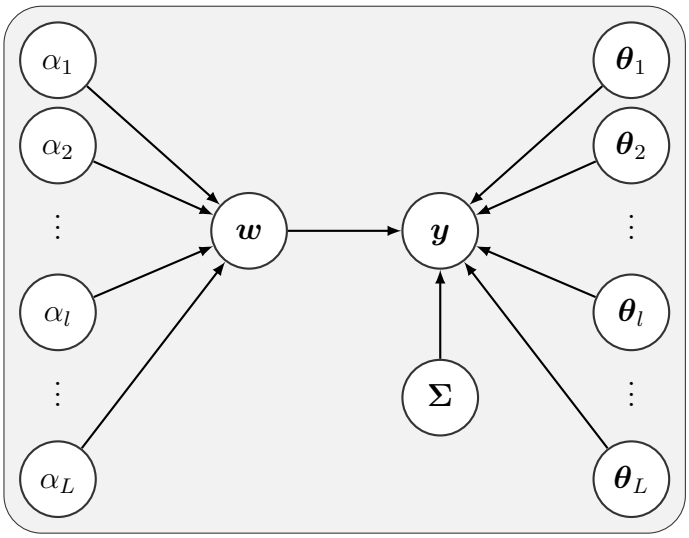


Figure 3: Graphical model representing the underlying signal model (3.1) with L components

According to this, the estimation problem can be described by

$$\underbrace{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}_{\text{Posterior}} \propto \underbrace{p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma})}_{\text{Likelihood}} \times \underbrace{p(\boldsymbol{w}|\boldsymbol{\alpha}) \times p(\boldsymbol{\alpha}) \times p(\boldsymbol{\Theta}) \times p(\boldsymbol{\Sigma})}_{\text{Prior}}, \tag{3.2}$$

with

$$p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \sim \mathcal{CN}\left(\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}, \boldsymbol{\Sigma}\right) \tag{3.3}$$

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) \sim \mathcal{CN}\left(\boldsymbol{0}, \mathrm{diag}(\boldsymbol{\alpha})^{-1}\right) \tag{3.4}$$

$$p(\boldsymbol{\alpha}) \sim \prod_{l=1}^{L} \mathcal{G}a\left(a_l, b_l\right), \tag{3.5}$$

where $\mathrm{diag}(\boldsymbol{\alpha})$ represents a diagonal matrix whose diagonal is the vector $\boldsymbol{\alpha}$. The choice of the prior $p(\boldsymbol{\Theta})$ is set to a dirac distribution in the context of this work. Our task is to evaluate the posterior of interest in an adequate way. At this point we will make use of Variational Bayesian methods, introduced in Beal (2003). This is a family of techniques that allow analytical approximations of the posterior $p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ just using a simpler approximating probability density function $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$. In the following, we will do a step by step derivation of the most important key facts on Variational Bayesian Inference using the SAGE algorithm to get a deeper understanding of the basic ideas.

Let us rewrite $\log p(\boldsymbol{y})$ as:

$$\log p(\boldsymbol{y}) = \log\left\{\frac{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}, \boldsymbol{y})}{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}\right\} \tag{3.6}$$

Now, we add $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ to the former equation. Note, that we do not have to make any assumptions on the choice of $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ for the moment. The basic idea of Variational Bayesian methods is to decompose $\log p(\boldsymbol{y})$:

$$\log p(\boldsymbol{y}) = \log\left\{\frac{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}, \boldsymbol{y})}{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})} \times \frac{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}\right\}$$

$$\log p(\boldsymbol{y}) = \log\left\{\frac{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}, \boldsymbol{y})}{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}\right\} + \log\left\{\frac{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}\right\}$$

Building the expectation over $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ on both sides will not effect the left part because of $\log p(\boldsymbol{y})$ being independent of the parameters $\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}$ and $\boldsymbol{\Sigma}$.

$$\log p(\boldsymbol{y}) = \mathop{\mathbb{E}}_{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}, \boldsymbol{y})}{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}\right\} + \mathop{\mathbb{E}}_{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}\right\}$$

$$\log p(\boldsymbol{y}) = \underbrace{\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}, \boldsymbol{y})}{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}\right\}}_{\text{Variational Lower Bound}} + \underbrace{\mathrm{D}_{\mathrm{KL}}(q(\cdot)\|p(\cdot))}_{\text{Kullback-Leibler divergence} \geq 0}. \tag{3.7}$$

In this connection, the second part of eq. (3.7) denotes the Kullback-Leibler divergence as a non-symmetric measure of the difference between the the probability density function $p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ and its approximation $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$. The Kullback-Leibler divergence is always non-negative and zero if and only if $p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y}) = q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ almost everywhere. Hence, if $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ is a 'good' approximation for $p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ ensures a small value of the KL divergence, otherwise the value will accordingly increase. Thus, it is sufficient to evaluate only the Variational Lower Bound (see eq. 3.8) for the maximization process:

$$\log p(\boldsymbol{y}) \geq \underbrace{\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})} \log \left\{ \frac{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}, \boldsymbol{y})}{q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})} \right\}}_{\text{Variational Lower Bound}} \tag{3.8}$$

Finally Variational inference for finding the optimal $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ is carried out by maximizing the Variational Lower Bound stated above. This will simultaneously force the aforementioned Kullback-Leibler divergence to be minimal due to the direct connection of both parts in eq. (3.7) and therefore, $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ represents the estimation for the posterior distribution $p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$. The significance of transforming eq. (3.6) to eq. (3.8) is that, for a suitable choice for $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$, the Variational Lower Bound in eq. (3.8) may be tractable to compute, even though the original model evidence function $p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ is not (Bishop and Tipping (2000)). Note that eq. 3.8 is used a bit 'sloppy' in this context. We will give a more precise formulation in the next section.

3.4 The Variational Bayesian SAGE Algorithm

Essentially, the VB-SAGE algorithm uses Variational Bayesian inference techniques, explained in detail in Beal (2003) and Tipping (2001), by approximating the posterior probability density function $p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$ of interest with an approximating probability density function $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})$, which we will assume to factor according to the graphical model given in fig. 3:

$$q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y}) = q(\boldsymbol{w}|\boldsymbol{\alpha}, \boldsymbol{y}) q(\boldsymbol{\Sigma}|\boldsymbol{y}) \prod_{l=1}^{L} q(\alpha_l|\boldsymbol{y}) q(\boldsymbol{\theta}_l|\boldsymbol{y}) \tag{3.9}$$

The optimal $q(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|y)$ is then found by maximizing eq. (3.8), what is shown in Appendix A.1 to be equivalent to maximizing

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log \left\{ \frac{c_0 \times \exp\left\{\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}) \times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\right\}\right\}}{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} - c_1 \right\},$$

where c_0, c_1 are constants. To simplify the notation we rewrite the Variational Lower Bound as

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \log \left\{ \frac{\widetilde{p}(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \right\}, \tag{3.10}$$

without respecting the constant term c_1 and consider two cases for the maximization process:

Case 1

$q(\boldsymbol{\vartheta}|\boldsymbol{y})$ is modelled as a dirac, thus $\delta(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}})$, then the Variational Lower Bound in (3.10) is maximized when

$$\widetilde{\boldsymbol{\vartheta}} = \mathop{\mathrm{argmax}}_{\boldsymbol{\vartheta}} \left\{\widetilde{p}(\boldsymbol{\vartheta})\right\}. \tag{3.11}$$

Case 2

$q(\boldsymbol{\vartheta}|\boldsymbol{y})$ is modelled as $\boldsymbol{\vartheta} \sim \mathcal{N}(\boldsymbol{\mu}_\vartheta, \boldsymbol{\Sigma}_\vartheta)$ and $\widetilde{p}(\boldsymbol{\vartheta}) \propto \mathcal{N}(\widetilde{\boldsymbol{\mu}}_\vartheta, \widetilde{\boldsymbol{\Sigma}}_\vartheta)$, then the Variational Lower Bound in eq. (3.10) is maximized when

$$q(\boldsymbol{\vartheta}|\boldsymbol{y}) = \mathcal{N}(\widetilde{\boldsymbol{\mu}}_\vartheta, \widetilde{\boldsymbol{\Sigma}}_\vartheta). \tag{3.12}$$

A detailed proof for the maximization equations of both cases can be found in Appendix A.2 of this work.

Just like SAGE, the VB-SAGE algorithm uses its advantages compared to standard EM approach by not maximizing all parameters at once but maximizing in a sequential manner.
For the model with L signal components we start with $l = 1$ and update the variational parameters related to the l-th component, i.e, we update the corresponding parameters, assuming that the parameters for the other components are known and fixed. The parameters for the component $l = 2$ are updated in the same fashion, and so on, until every component is considered. This procedure

of updating all parameters for all L multipath components constitutes one single update cycle of the algorithm, which are repeated anew until convergence. One of the key features of variational methods is that the factors in eq. (3.9) can be updated in any order.

In what follows, we consider the expressions for the variational parameters of the l-th component only in order to simplify the notation. This estimation principle is performed for all other multipath components $k \neq l$ in an analogue way.

3.4.1 Estimation of Signal Parameters $\boldsymbol{\Theta}$

Let us first have a look on variational inference of $\boldsymbol{\Theta} = [\boldsymbol{\tau}^T, \boldsymbol{\phi}^T]^T$ denoting the parameters for delay and azimuth angle of arrival. Since $q(\boldsymbol{\Theta}|\boldsymbol{y})$ is modelled as a dirac, thus $\delta(\boldsymbol{\Theta} - \widehat{\boldsymbol{\Theta}})$, we have to apply case one in eq. (3.11) to maximize the Variational Lower Bound given in eq. (3.10):

$$\{\widetilde{p}(\boldsymbol{\vartheta})\} := \mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}) \times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\}$$

$$\begin{aligned}\widehat{\boldsymbol{\Theta}} &= \mathop{\operatorname{argmax}}_{\boldsymbol{\Theta}} \left\{ \mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\} + \log\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\} \right\} \\ &= \mathop{\operatorname{argmax}}_{\boldsymbol{\Theta}} \left\{ \int_{\boldsymbol{w}} [\log\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\} + \log\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\}] \times p(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y}) \partial\boldsymbol{w} \right\}\end{aligned}$$

In what follows, we use $p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}) \sim \mathcal{CN}\left(\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w},\boldsymbol{\Sigma}\right)$ stated in eq. (3.3) and define $p(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y}) \sim \mathcal{CN}(\widehat{\boldsymbol{w}},\widehat{\boldsymbol{\Phi}})$. The detailed derivation for estimating the complex amplitudes parameters $\widehat{\boldsymbol{w}}$, $\widehat{\boldsymbol{\Phi}}$ can be found in section 3.4.3 of this work.

$$\begin{aligned}\widehat{\boldsymbol{\Theta}} = \mathop{\operatorname{argmax}}_{\boldsymbol{\Theta}} \Big\{ \int_{\boldsymbol{w}} \log\Big\{ c \times \exp\Big[-(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})^H \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}) \Big] \Big\} \times \\ p(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y}) \partial\boldsymbol{w} + \log\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\} \Big\} \quad \text{(where } c \text{ is a constant)} \\ = \mathop{\operatorname{argmax}}_{\boldsymbol{\Theta}} \Big\{ \int_{\boldsymbol{w}} \Big[\log c - \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \boldsymbol{w}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \boldsymbol{w} - \\ \boldsymbol{w}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \boldsymbol{w} \Big] \times p(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y}) \partial\boldsymbol{w} + \log\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\} \Big\}\end{aligned}$$

$$= \underset{\boldsymbol{\Theta}}{\operatorname{argmax}} \Big\{ \log c + \log\left\{p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma})\right\} - \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \widehat{\boldsymbol{w}}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}} - \int_{\boldsymbol{w}} \boldsymbol{w}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \boldsymbol{w} \times p(\boldsymbol{w}|\boldsymbol{\alpha}, \boldsymbol{y}) \partial \boldsymbol{w} \Big\}$$

Using below-mentioned transformation for multivariate Gaussians with random variable $\boldsymbol{x} \sim \mathcal{CN}(\widehat{\boldsymbol{x}}, \boldsymbol{\Sigma}_x)$ and symmetric matrix $\boldsymbol{A}$ in Petersen and Pedersen (2012)

$$\mathbb{E}(\boldsymbol{x}^H \boldsymbol{A} \boldsymbol{x}) = \operatorname{tr}(\boldsymbol{A} \boldsymbol{\Sigma}_x) + \widehat{\boldsymbol{x}}^H \boldsymbol{A} \widehat{\boldsymbol{x}},$$

which can be adopted to the case at hand as

$$\int_{\boldsymbol{w}} \boldsymbol{w}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \boldsymbol{w} \times p(\boldsymbol{w}|\boldsymbol{\alpha}, \boldsymbol{y}) \partial \boldsymbol{w} = \operatorname{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{\Phi}}) + \widehat{\boldsymbol{w}}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}}.$$

Therefore, we proceed with

$$\begin{aligned}
\widehat{\boldsymbol{\Theta}} &= \underset{\boldsymbol{\Theta}}{\operatorname{argmax}} \Big\{ \log c + \log\left\{p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma})\right\} - \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \widehat{\boldsymbol{w}}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \\
&\qquad \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}} - \operatorname{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{\Phi}}) + \widehat{\boldsymbol{w}}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}} \Big\} \\
&\propto \underset{\boldsymbol{\Theta}}{\operatorname{argmax}} \Big\{ \log\left\{p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma})\right\} - (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}})^H \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}}) - \\
&\qquad \operatorname{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{\Phi}}) \Big\}. \qquad (3.13)
\end{aligned}$$

With restricting to an estimator for the l-th component only, $\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}}$ changes to $\widehat{\boldsymbol{x}}_l - \boldsymbol{s}(\boldsymbol{\theta}_l) \widehat{w}_l$, where we define

$$\widehat{\boldsymbol{x}}_l := \boldsymbol{y} - \sum_{k=1, k \neq l}^{L} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_k) \widehat{w}_k. \qquad (3.14)$$

In Shutin et al. (2013) it is described that this formulation takes the correlations between individual multipath components into account, effectively penalizing the

estimator $\widehat{\boldsymbol{\Theta}}$. We now seek to reformulate eq. 3.13 in terms of considering an expression for one single component in its final form

$$\widehat{\boldsymbol{\theta}}_l = \underset{\boldsymbol{\theta}_l}{\operatorname{argmax}} \Big\{ \log p(\boldsymbol{\theta}_l) - (\widehat{\boldsymbol{x}}_l - \boldsymbol{s}(\boldsymbol{\theta}_l)\widehat{w}_l)^H \boldsymbol{\Sigma}^{-1} (\widehat{\boldsymbol{x}}_l - \boldsymbol{s}(\boldsymbol{\theta}_l)\widehat{w}_l) - \sum_{k=1, k\neq l}^{L} 2\Re\Big\{\widehat{\Phi}_{kl}\boldsymbol{s}(\boldsymbol{\theta}_k)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\boldsymbol{\theta}_k)\Big\} - \widehat{\Phi}_{ll}\boldsymbol{s}(\boldsymbol{\theta}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\boldsymbol{\theta}_l)\Big\}, \tag{3.15}$$

where $\Re\{\cdot\}$ denotes the real part operator. Additionally, $\widehat{\Phi}_{kl}$, $\widehat{\Phi}_{ll}$ stand for the corresponding entries of the complex amplitude covariance matrix $\widehat{\boldsymbol{\Phi}}$, when restricting to the k-th and l-th column and row.

Due to the non-linear dependence of $\boldsymbol{s}(\boldsymbol{\theta}_l)$ on $\boldsymbol{\theta}_l$, eq. (3.15) has to be optimized numerically, e.g. using successive line searches, where each element of $\widehat{\boldsymbol{\theta}}_l$ is determined separately or using a joint search as an alternative in which all elements of $\widehat{\boldsymbol{\theta}}_l$ are computed jointly. Notice that the same assumption of numerical optimization underpins the classical SAGE-based estimation of $\boldsymbol{\theta}_l$ (Shutin and Fleury (2011)).

3.4.2 Estimation of the Noise Covariance Matrix $\boldsymbol{\Sigma}$

Within the Variational Bayesian SAGE algorithm we need to develop an estimation expression for the covariance matrix $\boldsymbol{\Sigma}$. Therefore, we maximize the Variational Lower Bound given in eq. (3.10) in an analogue way as described in the former section:

$$\{\widetilde{p}(\boldsymbol{\vartheta})\} := \underset{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}{\mathbb{E}} \log\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}) \times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\}$$

$$\begin{aligned}\widehat{\boldsymbol{\Sigma}} &= \underset{\boldsymbol{\Sigma}}{\operatorname{argmax}} \left\{ \underset{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}{\mathbb{E}} \log\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\} + \log\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\} \right\} \\ &= \underset{\boldsymbol{\Sigma}}{\operatorname{argmax}} \left\{ \int_{\boldsymbol{w}} [\log\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\} + \log\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\}] \times p(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})\partial\boldsymbol{w} \right\}\end{aligned}$$

In this connection we assume the distribution for $p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})$ factors as in section 3.4.1 and make use of the transformation for multivariate Gaussians stated in

Petersen and Pedersen (2012) again. Then, further calculation is straightforward:

$$
\begin{aligned}
\widehat{\boldsymbol{\Sigma}} = \underset{\boldsymbol{\Sigma}}{\operatorname{argmax}} \Big\{ & \int_{\boldsymbol{w}} \log \Big\{ \pi^{-N} (\det(\boldsymbol{\Sigma}))^{-1} \times \\
& \exp \Big[- (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})^H \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}) \Big] \Big\} \times \\
& p(\boldsymbol{w}|\boldsymbol{\alpha}, \boldsymbol{y}) \partial \boldsymbol{w} + \log \{ p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \} \Big\} \\
= \underset{\boldsymbol{\Sigma}}{\operatorname{argmax}} \Big\{ & - N \log \pi - \log \{ \det(\boldsymbol{\Sigma}) \} + \log \{ p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \} - \\
& \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \widehat{\boldsymbol{w}}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}} - \\
& \int_{\boldsymbol{w}} \boldsymbol{w}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \boldsymbol{w} \times p(\boldsymbol{w}|\boldsymbol{\alpha}, \boldsymbol{y}) \partial \boldsymbol{w} \Big\} \\
= \underset{\boldsymbol{\Sigma}}{\operatorname{argmax}} \Big\{ & - N \log \pi - \log \{ \det(\boldsymbol{\Sigma}) \} + \log \{ p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \} - \\
& \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \widehat{\boldsymbol{w}}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} + \boldsymbol{y}^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}} - \\
& \operatorname{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \boldsymbol{\Phi}) + \widehat{\boldsymbol{w}}^H \boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}} \Big\} \\
\propto \underset{\boldsymbol{\Sigma}}{\operatorname{argmax}} \Big\{ & - \log \{ \det(\boldsymbol{\Sigma}) \} + \log \{ p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \} - \\
& (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}})^H \boldsymbol{\Sigma}^{-1} (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}}) - \operatorname{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \boldsymbol{\Phi}) \Big\}.
\end{aligned}
\tag{3.16}
$$

In what follows we assume the noise covariance matrix represented by

$$
\boldsymbol{\Sigma} = \boldsymbol{I} \sigma^2, \tag{3.17}
$$

meaning $\boldsymbol{\Sigma}$ is expected as a diagonal matrix of size $(N \times N)$ with identical elements σ^2. Thus the noise variance is assumed to be equal for all input samples N. Using this eq. (3.17), our maximization problem in eq. (3.16) can be reformulated and extended to:

$$
\begin{aligned}
\widehat{\sigma}^2 = \underset{\sigma^2}{\operatorname{argmax}} \Big\{ & - \log \big\{ (\sigma^2)^N \big\} + \log \big\{ p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \sigma^2) \big\} - \\
& (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}})^H (\sigma^2)^{-1} (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta}) \widehat{\boldsymbol{w}}) - \operatorname{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H (\sigma^2)^{-1} \boldsymbol{S}(\boldsymbol{\Theta}) \boldsymbol{\Phi}) \Big\} \quad \Leftrightarrow
\end{aligned}
$$

$$\frac{\partial}{\partial \sigma^2}\Big\{ -\log\left\{(\sigma^2)^N\right\} + \log\left\{p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \sigma^2)\right\} - (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}})^H(\sigma^2)^{-1}(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}}) - \mathrm{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H(\sigma^2)^{-1}\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{\Phi})\Big\} \stackrel{!}{=} 0 \quad \Leftrightarrow$$

$$-N(\sigma^2)^{-1} = (\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}})^H(\sigma^2)^{-2}(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}}) + \mathrm{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H(\sigma^2)^{-2}\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{\Phi})$$

Note that since $\log\{p(\boldsymbol{\Theta}, \boldsymbol{\alpha}, \sigma^2)\}$ is assumed to be uninformative, we can omit this part in the last line. We proceed with using σ^2 being a scalar to set up the final equation to calculate the estimator for σ^2:

$$\widehat{\sigma}^2 = \frac{(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}})^H(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}}) + \mathrm{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{\Phi})}{N}. \tag{3.18}$$

3.4.3 Estimation of Complex Amplitudes Parameters $\boldsymbol{w}, \boldsymbol{\Phi}$

In this section we are going to find the estimation expressions for the complex amplitude vector $\boldsymbol{w}$ and its respective covariance matrix $\boldsymbol{\Phi}$ within a single cycle of the algorithm. Note that separated estimates for magnitude and phase can be easily derived by transforming the resulting expressions for the complex amplitude. Before starting with the actual proof, the final equations within the VB-SAGE algorithm are given by:

$$\widehat{\boldsymbol{w}} = \left(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) + \mathrm{diag}(\boldsymbol{\alpha})\right)^{-1}\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y} \tag{3.19}$$

$$\widehat{\boldsymbol{\Phi}} = \left(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) + \mathrm{diag}(\boldsymbol{\alpha})\right)^{-1}. \tag{3.20}$$

To derive these expressions we maximize the Variational Lower Bound (see eq. (3.10)) by applying the second case stated in eq. (3.12):

$$q(\boldsymbol{w}|\boldsymbol{\alpha}, \boldsymbol{y}) = p(\boldsymbol{w}|\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}, \boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha})}{\int\limits_{\boldsymbol{w}} p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha})\partial\boldsymbol{w}}, \tag{3.21}$$

with using of eqs. (3.3) and (3.4) results

$$p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \sim \mathcal{CN}\left(\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}, \boldsymbol{\Sigma}\right)$$

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) \sim \mathcal{CN}\left(\mathbf{0}, \mathrm{diag}(\boldsymbol{\alpha})^{-1}\right).$$

According to Tipping (2001), the denominator in eq. (3.21) can be reformulated as

$$\begin{aligned}\int_{\boldsymbol{w}} p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha}) \partial \boldsymbol{w} = \\ p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) = \\ (2\pi)^{\frac{N}{2}} (\boldsymbol{\Sigma} + \boldsymbol{S}(\boldsymbol{\Theta}) \,\mathrm{diag}(\boldsymbol{\alpha})^{-1} \boldsymbol{S}(\boldsymbol{\Theta})^H)^{\frac{1}{2}} \times \\ \exp\left\{-\boldsymbol{y}^H (\boldsymbol{\Sigma} + \boldsymbol{S}(\boldsymbol{\Theta}) \,\mathrm{diag}(\boldsymbol{\alpha})^{-1} \boldsymbol{S}(\boldsymbol{\Theta})^H) \boldsymbol{y}\right\}\end{aligned}$$

what is used to set up the desired probability density function

$$\begin{aligned}q(\boldsymbol{w}|\boldsymbol{\alpha}, \boldsymbol{y}) = p(\boldsymbol{w}|\boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}, \boldsymbol{y}) = \\ (2\pi)^{-\frac{N+1}{2}} |\widehat{\boldsymbol{\Phi}}|^{\frac{1}{2}} \exp\left\{-(\boldsymbol{w} - \widehat{\boldsymbol{w}})^H \widehat{\boldsymbol{\Phi}}^{-1} (\boldsymbol{w} - \widehat{\boldsymbol{w}})\right\},\end{aligned}$$

with $\widehat{\boldsymbol{w}}$ and $\widehat{\boldsymbol{\Phi}}$ as given in eqs. (3.19) and (3.20).

3.4.4 Estimation of the Precision Parameter $\boldsymbol{\alpha}$

In this section we want to come up with an expression for estimating the vector of precision parameters within the VB-SAGE framework. Therefore, we maximize the Variational Lower Bound (see eq. (3.10)) by applying the second case given in eq. (3.12):

$$q(\boldsymbol{\alpha}|\boldsymbol{y}) = p(\boldsymbol{\alpha}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\Sigma}, \boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha}) \times p(\boldsymbol{\alpha})}{\int_{\boldsymbol{\alpha}} p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha}) \times p(\boldsymbol{\alpha}) \partial \boldsymbol{\alpha}}. \quad (3.22)$$

In contrast to the derivation of complex amplitudes parameters in section 3.4.3, we only seek to find an adequate estimator for the vector of precision parameters $\boldsymbol{\alpha}$, i.e. an expression for the respective covariance matrix is not needed within the framework. Therefore, we can simplify eq. (3.22) to

$$q(\boldsymbol{\alpha}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha}) \times p(\boldsymbol{\alpha}).$$

In this connection we assume $p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma})$ modelled as in eq. (3.3) and $p(\boldsymbol{w}|\boldsymbol{\alpha})$ as stated in eq. (3.4) respectively. Furthermore we consider below-mentioned distribution (see also eq. (3.5)) for the precision parameter:

$$p(\boldsymbol{\alpha}) \sim \prod_{l=1}^{L} \mathcal{G}a\left(a_l, b_l\right),$$

where a_l and b_l are both set to 10^{-7} for all components to make the hyper-prior being non-informative.

By substituting $p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma})$, $p(\boldsymbol{w}|\boldsymbol{\alpha})$ and $p(\boldsymbol{\alpha})$ with their distributions plus expanding the resulting posterior density function in the same way as in chapter 3.4.3 will yield a "'pseudo"'-likelihood function $\ell(\boldsymbol{\alpha}|\boldsymbol{y})$, which to be used for developing an estimation expression of the precision parameter:

$$\begin{aligned}
&\log\Big\{q(\boldsymbol{\alpha}|\boldsymbol{y})\Big\} \\
&\quad\propto \log\Big\{p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha}) \times p(\boldsymbol{\alpha})\Big\} \\
&\quad\propto \log\Big\{\exp\left[-(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})^H \boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})\right] \times \\
&\qquad\quad \exp\left[-\boldsymbol{w}^H \operatorname{diag}(\boldsymbol{\alpha})\boldsymbol{w}\right] \times \\
&\qquad\quad \operatorname{diag}(\boldsymbol{\alpha})\Big\} \\
&\quad= -\boldsymbol{y}^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y} + \boldsymbol{w}^H\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y} - \boldsymbol{w}^H\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w} + \\
&\qquad \boldsymbol{y}^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w} - \boldsymbol{w}^H \operatorname{diag}(\boldsymbol{\alpha})\boldsymbol{w} + \log(\operatorname{diag}(\boldsymbol{\alpha})) \\
&\quad:= \ \ell(\boldsymbol{\alpha}|\boldsymbol{y})
\end{aligned}$$

In what follows is known as maximum a posteriori estimation (MAP), what can be seen as a regularization of ML estimation in the Bayesian approach. Anyway MAP estimation is not very representative for Bayesian methods in general. This is because MAP estimates are as well as ML estimates point estimates, whereas Bayesian techniques are characterized by the use of distributions to summarize information and draw inferences. The method of maximum a posterior estimation estimates $\boldsymbol{\alpha}$ as the mode of the posterior distribution by first maximization and last solving for $\widehat{\boldsymbol{\alpha}}$.

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmax}}\, \ell(\boldsymbol{\alpha}|\boldsymbol{y}) \qquad \Leftrightarrow$$

$$\frac{\partial \ell(\boldsymbol{\alpha}|\boldsymbol{y})}{\partial \boldsymbol{\alpha}} = -\boldsymbol{w}^H \boldsymbol{I} \boldsymbol{w} + \mathrm{diag}(\boldsymbol{\alpha})^{-1} \stackrel{!}{=} 0 \qquad \Leftrightarrow$$
$$\mathrm{diag}(\widehat{\boldsymbol{\alpha}}) = (\boldsymbol{w}^H \boldsymbol{w})^{-1} = \left(|\boldsymbol{w}|^2\right)^{-1} \tag{3.23}$$

Due to $\boldsymbol{w}$ is constraint by $\boldsymbol{\alpha}$ the expectation of $|\boldsymbol{w}|^2$ has to be computed by applying the expanded expression for the covariance

$$\mathbb{E}_{\boldsymbol{\alpha}}\left(|\boldsymbol{w}|^2\right) = \mathrm{Cov}_{\boldsymbol{\alpha}}\left(\boldsymbol{w}\right) + \left(\mathbb{E}_{\boldsymbol{\alpha}}\left(\boldsymbol{w}\right)\right)^2 = \widehat{\boldsymbol{\Phi}} + |\widehat{\boldsymbol{w}}|^2. \tag{3.24}$$

Plugging in eq. (3.24) into eq. (3.23) finally leads to the following update expression:

$$\mathrm{diag}(\widehat{\boldsymbol{\alpha}}) = (\widehat{\boldsymbol{\Phi}} + |\widehat{\boldsymbol{w}}|^2)^{-1}. \tag{3.25}$$

Eq. (3.25) can be reformulated to get the parameter estimate α_l for one specific component l

$$\widehat{\alpha_l} = \frac{1}{\widehat{\Phi}_{ll} + |\widehat{w}_l|^2}, \tag{3.26}$$

where $\widehat{\Phi}_{ll}$ is the l-th element on the main diagonal of the complex amplitude covariance matrix $\widehat{\boldsymbol{\Phi}}$, and $\widehat{w}_l$ is the l-th element of the vector $\widehat{\boldsymbol{w}}$. Note that $\widehat{\alpha_l}$ uses the parameter estimates $\widehat{\Phi}_{ll}$, $\widehat{w}_l$ of the same iteration. Therefore, the estimates of a fixed component l are performed successively ad infinitum while keeping the other estimates fixed. Note that eq. (3.26) will play a decisive role in deciding if a multipath component should be kept within the Variational Bayesian SAGE framework or not, what is part of a separate paragraph of this work (see section 3.4.5).

3.4.5 Model Order Detection

Within the VB-SAGE framework we introduced the precision parameter $\boldsymbol{\alpha}$ in order to avoid an wrong estimation of the number of relevant propagation paths. Therefore, we will give a short overview how $\boldsymbol{\alpha}$ can be used as a model selection criteria, i.e to decide if an estimated path is merely fictive and so to remove from the model or does actually exist and should be included in the estimation scheme.

The resulting test (see Shutin et al. (2013)) to keep a component can be stated as

$$\omega_l^2 > \varsigma_l, \tag{3.27}$$

with

$$\varsigma_l = \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) - \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l}) \widehat{\boldsymbol{\Phi}}_{-l} \boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) \right)^{-1} \tag{3.28}$$

$$\omega_l^2 = \left(\varsigma_l \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} - \varsigma_l \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l}) \widehat{\boldsymbol{\Phi}}_{-l} \boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l})^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} \right)^2, \tag{3.29}$$

where $\widehat{\boldsymbol{\Phi}}_{-l}$ and $\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l})$ are respective values without considering the l-th component.

For simplicity we will assume that only one multipath component exists in the model in order to derive the selection criteria. The basic intuition remains unchanged for more than one single path and can be found in Shutin et al. (2011a) and Shutin et al. (2011b). Consequently eqs. (3.28) and (3.29) reduce to

$$\varsigma_l = \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) \right)^{-1}$$

$$\omega_l^2 = \left(\varsigma_l \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{y} \right)^2$$

since no other but the l-th component is available and therefore $\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l}) = 0$. Applying eq. (3.14), $\boldsymbol{y}$ can be set to $\widehat{\boldsymbol{x}}_l$ respectively. Let us now reformulate eq. (3.27) in order to uncover the connection to the precision parameter $\boldsymbol{\alpha}$.

$$\begin{aligned}
\omega_l^2 > \varsigma_l \qquad & \Leftrightarrow \\
\left(\varsigma_l \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l \right)^2 > \varsigma_l \qquad & \Leftrightarrow \\
\varsigma_l^2 \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l \right)^2 > \varsigma_l \qquad & \Leftrightarrow \\
\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l \right)^2 > \varsigma_l^{-1} \qquad & \Leftrightarrow \\
\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l \right)^2 > \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) \qquad & (3.30)
\end{aligned}$$

In fact the precision parameter $\boldsymbol{\alpha}$ is deeply connected with the extended test in eq. (3.30). To point this out, one has to take the estimator for α_l (see chapter 3.4.4)

$$\widehat{\alpha}_l = \frac{1}{\widehat{\Phi}_{ll} + |\widehat{w}_l|^2} \tag{3.31}$$

and plugging in the expressions of $\widehat{\Phi}_{ll}$, $\widehat{w}_l$ given in section 3.4.3 with respect to the assumption of one component only

$$\widehat{\Phi}_{ll} = \left(\boldsymbol{s}(\boldsymbol{\theta}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\boldsymbol{\theta}_l) + \widehat{\alpha}_l\right)^{-1} \tag{3.32}$$

$$\begin{aligned}\widehat{w}_l &= \widehat{\Phi}_{ll} \boldsymbol{s}(\boldsymbol{\theta}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l \\ &= \frac{\boldsymbol{s}(\boldsymbol{\theta}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l}{\boldsymbol{s}(\boldsymbol{\theta}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\boldsymbol{\theta}_l) + \widehat{\alpha}_l}.\end{aligned} \tag{3.33}$$

Note that the updating steps given in eqs. (3.31), (3.32) and (3.33) are repeated, while keeping $\widehat{\boldsymbol{x}}_l$ and $\widehat{\boldsymbol{\theta}}_l$ fixed to generate a sequence $\{\widehat{\alpha}_l^{[m]}\}_{m>0}$. Plugging in eqs. (3.32), (3.33) into eq. (3.31) and solving for $\widehat{\alpha}_l^{[\infty]}$ as the stationary point of $\{\widehat{\alpha}_l^{[m]}\}_{m>0}$ leads to the estimation expression for α_l, when considering the existence of one single multipath.

$$\begin{aligned}\frac{1}{\widehat{\alpha}_l^{[\infty]}} &= \left(\frac{\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l}{\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) + \widehat{\alpha}_l^{[\infty]}}\right)^2 + \frac{1}{\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) + \widehat{\alpha}_l^{[\infty]}} \\ &= \frac{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l\right)^2}{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) + \widehat{\alpha}_l^{[\infty]}\right)^2} + \frac{\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) + \widehat{\alpha}_l^{[\infty]}}{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) + \widehat{\alpha}_l^{[\infty]}\right)^2}\end{aligned}$$

$$\begin{aligned}\widehat{\alpha}_l^{[\infty]} &= \frac{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) + \widehat{\alpha}_l^{[\infty]}\right)^2}{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l\right)^2 + \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) + \widehat{\alpha}_l^{[\infty]}} \\ &= \frac{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right)^2 + \left(2\widehat{\alpha}_l^{[\infty]} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right) + \left(\widehat{\alpha}_l^{[\infty]}\right)^2}{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l\right)^2 + \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) + \widehat{\alpha}_l^{[\infty]}} \quad \Leftrightarrow\end{aligned}$$

$$\begin{aligned}&\widehat{\alpha}_l^{[\infty]} \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l\right)^2 + \widehat{\alpha}_l^{[\infty]} \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right) + \widehat{\alpha}_l^{[\infty]} \widehat{\alpha}_l^{[\infty]} = \\ &\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right)^2 + \left(2\widehat{\alpha}_l^{[\infty]} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right) + \widehat{\alpha}_l^{[\infty]} \widehat{\alpha}_l^{[\infty]} \quad \Leftrightarrow\end{aligned}$$

$$\widehat{\alpha}_l^{[\infty]} \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \widehat{\boldsymbol{x}}_l\right)^2 = \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right)^2 + \left(\widehat{\alpha}_l^{[\infty]} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H \boldsymbol{\Sigma}^{-1} \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right) \quad \Leftrightarrow$$

$$\widehat{\alpha}_l^{[\infty]}\left[\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\widehat{\boldsymbol{x}}_l\right)^2-\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right]=\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right)^2 \quad \Leftrightarrow$$

$$\widehat{\alpha}_l^{[\infty]}=\frac{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right)^2}{\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\boldsymbol{\Sigma}^{-1}\widehat{\boldsymbol{x}}_l\right)^2-\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)}$$

By definition $\widehat{\alpha}_l^{[\infty]}>0$ which is satisfied if, and only, if for the denominator holds

$$\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\boldsymbol{\Sigma}^{-1}\widehat{\boldsymbol{x}}_l\right)^2>\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) \tag{3.34}$$

what equals eq. (3.30) as an reformulation of eq. (3.27) and thus the test $\omega^2>\varsigma$ itself. As a preliminary result let us point out that the whole precision parameter scheme is 'encoded' in the test $\omega^2>\varsigma$ that determines the convergence of α_l update: if $\widehat{\alpha}_l$ diverges, thus $\omega^2<\varsigma$ and the corresponding component is removed from the model.

To close this chapter we will give a more intuitive interpretation of the derived results. Since we assume the one component case, the covariance noise matrix $\boldsymbol{\Sigma}$ simplifies to σ^2 and allows for extending eq. (3.30) with $\widehat{\boldsymbol{x}}_l=w_l\boldsymbol{S}(\widehat{\boldsymbol{\Theta}})$

$$\begin{aligned}
\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\widehat{\boldsymbol{x}}_l\right)^2&>\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) && \Leftrightarrow\\
\sigma^{-4}\left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\widehat{\boldsymbol{x}}_l\right)^2&>\sigma^{-2}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) && \Leftrightarrow\\
\sigma^{-4}(N)^2w_l^2&>\sigma^{-2}N && \Leftrightarrow\\
w_l^2&<\frac{\sigma^2}{N}. && (3.35)
\end{aligned}$$

We obtain as a final result that the component will be kept if the squared complex amplitude is above the average noise level, otherwise it is removed.

3.4.6 Summary of the VB-SAGE Algorithm

Finally the VB-SAGE algorithm can be set up by putting together all the puzzle pieces of estimation equations. The proposed algorithm updates the factors in eq. (3.9) in groups, where the l-th group contains factors $\{q(\boldsymbol{\theta}_l), q(\boldsymbol{w}), q(\alpha_l)\}$: starting with $q(\boldsymbol{\theta}_l)$, we then update $q(\boldsymbol{w})$ and $q(\alpha_l)$. If the estimate of $\widehat{\alpha}_l$ diverges, the

corresponding multipath component is removed from the model; otherwise, its parameters are updated, and the next component $l+1$ is considered. The realization of the algorithm includes two steps, which are carried out in a sequential manner: the initialization algorithm 1 and update algorithm 2, respectively. In the following we want to consider the characteristics of both types of algorithms, before showing up some differences plus important key facts on both implementations.

The initialization of VB-SAGE uses a simple bottom-up strategy by starting with an empty model, i.e assuming all variational parameters to be zero. The first component is initialized by setting $\widehat{\boldsymbol{x}}_1 = \boldsymbol{y}$ and applying the initialization loop shown in Algorithm 1. The obtained estimate $\widehat{\boldsymbol{\theta}}_l$ is plugged in eqs. (3.28), (3.29) to determine whether the actual component should be kept in the model. When the test $\omega_l^2 > \varsigma_l$ fails, the initialization stops. In this implementation we will make use of this condition plus limiting the number of initialization iterations to $L_{\max}$. Thus, the algorithm will stop if the pruning condition $\omega_l^2 > \varsigma_l$ fails or if the number of maximal components $L_{\max}$ is reached at some iteration.
Algorithm 2 shows one update cycle for a single multipath component l. In opposition to the initialization strategy another stopping criterion has to be defined, since the pruning condition $\omega_l^2 > \varsigma_l$ only allows for removing components from the model but not for quitting the algorithm. In this implementation we use the following simple criterion: The update iterations are terminated when first the number of estimated signal components stabilizes and second the maximum change of the components in $\{\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}\}$ between two consecutive update cycles is less than 10^{-4}.

In the end we want to summarize what VB-SAGE can do for us. In contrast to the classical SAGE algorithm, VB-SAGE does not only allow for estimation of all parameters considering the underlying signal model given in eq. (3.1), but also for model order selection by deciding if a signal component should be kept.

Up to now we are only considering the case at fixed snapshot time t. In a world with a moving receiver which we will assume in chapter 4, the estimation process has to be carried out for consecutive snapshots to track how paths evolve with time. Therefore, the Kalman Filter will be used. Prior to this we would like to evaluate the performance of the VB-SAGE scheme in the next section.

Algorithm 1: Initialization of VB-SAGE

Set $l \leftarrow 0$, $\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}) \leftarrow [\,]$, $\widehat{\boldsymbol{\alpha}} \leftarrow [\,]$, $\widehat{\boldsymbol{w}} \leftarrow [\,]$, $\widehat{\boldsymbol{\Phi}} \leftarrow [\,]$, *Continue* $\leftarrow$ *true*

while *Continue* = *true* **do**

$l = l + 1$

Compute $\widehat{\boldsymbol{x}}_l$ from (3.14): $\widehat{\boldsymbol{x}}_l = \boldsymbol{y} - \sum_{k=1,k\neq l}^{L} \widehat{w}_k \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_k)$

Compute $\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)$ from (3.15):

$$\widehat{\boldsymbol{\theta}}_l = \text{argmax}_{\boldsymbol{\theta}_l} \Big\{ \log p(\boldsymbol{\theta}_l) - (\widehat{\boldsymbol{x}}_l - \boldsymbol{s}(\boldsymbol{\theta}_l)\widehat{w}_l)^H \boldsymbol{\Sigma}^{-1} (\widehat{\boldsymbol{x}}_l - \boldsymbol{s}(\boldsymbol{\theta}_l)\widehat{w}_l) - \sum_{k=1,k\neq l}^{L} 2\Re\left\{\Phi_{kl}\boldsymbol{s}(\boldsymbol{\theta}_k)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\boldsymbol{\theta}_k)\right\} - \Phi_{ll}\boldsymbol{s}(\boldsymbol{\theta}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\boldsymbol{\theta}_l)\Big\}$$

Compute ς_l from (3.28):

$$\varsigma_l = \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) - \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\widehat{\boldsymbol{\Theta}})\widehat{\boldsymbol{\Phi}}\boldsymbol{S}(\widehat{\boldsymbol{\Theta}})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right)^{-1}$$

Compute ω_l^2 from (3.29):

$$\omega_l^2 = \left(\varsigma_l\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y} - \varsigma_l\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\widehat{\boldsymbol{\Theta}})\widehat{\boldsymbol{\Phi}}\boldsymbol{S}(\widehat{\boldsymbol{\Theta}})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y}\right)^2$$

if $\omega_l^2 > \varsigma_l$ **then**

Add a new component l and update ...

$\widehat{\alpha}_l$ by: $\widehat{\alpha}_l = (\omega^2 - \varsigma_l)^{-1}$

$\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}) = [\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}), \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)]$

$\widehat{\boldsymbol{\Phi}}$ from (3.20): $\widehat{\boldsymbol{\Phi}} = \left(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) + \text{diag}(\boldsymbol{\alpha})\right)^{-1}$

$\widehat{\boldsymbol{w}}$ from (3.19): $\left(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) + \text{diag}(\boldsymbol{\alpha})\right)^{-1}\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y}$

$\widehat{\boldsymbol{\Sigma}}$ from (3.18):

$\boldsymbol{\Sigma} = \text{diag}(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}})^H(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}}) + \text{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{\Phi})N^{-1})$

else

Reject component l and stop initialization

Set number of components as: $L = l - 1$

Continue = *false*

end

end

Algorithm 2: Update of VB-SAGE

Set $l \leftarrow 0$, *Continue* $\leftarrow$ *true*

while *Continue* = *true* **do**

$l = l + 1$

Compute $\widehat{\boldsymbol{x}}_l$ from (3.14): $\widehat{\boldsymbol{x}}_l = \boldsymbol{y} - \sum_{k=1,k\neq l}^{L} \widehat{w}_k \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_k)$

Set $\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l}) = \boldsymbol{S}(\widehat{\boldsymbol{\Theta}}) \backslash \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)$

Set $\widehat{\boldsymbol{\Phi}}_{-l} = \left[\widehat{\boldsymbol{\Phi}} - \frac{\widehat{\boldsymbol{\Phi}}\boldsymbol{e}_l\boldsymbol{e}_l^H\widehat{\boldsymbol{\Phi}}}{\boldsymbol{e}_l^H\widehat{\boldsymbol{\Phi}}\boldsymbol{e}_l}\right]_{\overline{ll}}$ (See details in [a])

Compute ς_l from (3.28):

$$\varsigma_l = \left(\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l) - \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l})\widehat{\boldsymbol{\Phi}}_{-l}\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)\right)^{-1}$$

Compute ω_l^2 from (3.29):

$$\omega_l^2 = \left(\varsigma_l\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y} - \varsigma_l\boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l})\widehat{\boldsymbol{\Phi}}_{-l}\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y}\right)^2$$

if $\omega^2 > \varsigma_l$ **then**

Keep component l and update ...

$\widehat{\alpha}_l$ by: $\widehat{\alpha}_l = (\omega^2 - \varsigma_l)^{-1}$

$\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}) = [\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l}), \boldsymbol{s}(\widehat{\boldsymbol{\theta}}_l)]$

$\widehat{\boldsymbol{\Phi}}$ from (3.20): $\widehat{\boldsymbol{\Phi}} = \left(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) + \mathrm{diag}(\boldsymbol{\alpha})\right)^{-1}$

$\widehat{\boldsymbol{w}}$ from (3.19): $\left(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) + \mathrm{diag}(\boldsymbol{\alpha})\right)^{-1}\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y}$

$\widehat{\boldsymbol{\Sigma}}$ from (3.18):

$\boldsymbol{\Sigma} = \mathrm{diag}(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}})^H(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}}) + \mathrm{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{\Phi})N^{-1})$

else

Remove component l and set ...

$\boldsymbol{S}(\widehat{\boldsymbol{\Theta}}) = \boldsymbol{S}(\widehat{\boldsymbol{\Theta}}_{-l})$

$\boldsymbol{\alpha} = \boldsymbol{\alpha} \backslash \widehat{\alpha}_l$

$\widehat{\boldsymbol{\Phi}} = \widehat{\boldsymbol{\Phi}}_{-l}$

$\widehat{\boldsymbol{w}} = \left(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) + \boldsymbol{\alpha}\right)^{-1}\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y}$

$\boldsymbol{\Sigma} = \mathrm{diag}(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}})^H(\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})\widehat{\boldsymbol{w}}) + \mathrm{tr}(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{\Phi})N^{-1})$

end

end

[a] Here we use $\boldsymbol{e}_l$ as a canonical vector with length L, i.e. $\boldsymbol{e}_l = [0, \ldots, 0, 1, 0, \ldots, 0]^T$. Furthermore $[\boldsymbol{A}]_{\overline{ll}}$ denotes a matrix obtained by deleting the l-th row and l-th column from the matrix $\boldsymbol{A}$.

3.5 Performance of the VB-SAGE Algorithm

To conclude this chapter a discussion about the algorithms performance should be carried out. Therefore, a simulation study is appropriate. In the following we use two different definitions of the mean squared error (MSE). Generally the MSE of any parameter vector $\boldsymbol{\vartheta}$ is given by

$$\mathrm{MSE}(\widehat{\boldsymbol{\vartheta}}) = \mathrm{Cov}(\widehat{\boldsymbol{\vartheta}}) + \left(\mathrm{Bias}(\widehat{\boldsymbol{\vartheta}})\right)^2 \tag{3.36}$$

$$= \mathbb{E}\left(|\widehat{\boldsymbol{\vartheta}} - \boldsymbol{\vartheta}|^2\right) \tag{3.37}$$

If an estimator $\widehat{\boldsymbol{\vartheta}}$ is unbiased $\mathrm{Cov}(\widehat{\boldsymbol{\vartheta}})$ can be compared with the Cramer-Rao Lower Bound (CRLB) which is determined by the inverse of the Fisher-Information matrix $[\mathcal{I}(\boldsymbol{\vartheta})]_{ij}$, otherwise one has to respect the Bias for derivation of a lower bound for $\mathrm{Cov}(\widehat{\boldsymbol{\vartheta}})$ as well. In what follows we compute an adequate, theoretical lower bound for each delay, azimuth angle of arrival, and complex amplitude. Subsequently these bounds are compared with the second definition of the MSE stated in eq. (3.37) which makes use of the true and in fact unknown parameter values to evaluate the algorithms performance.

3.5.1 Lower Bound for Delay $\boldsymbol{\tau}$ and Azimuth Angle of Arrival $\boldsymbol{\phi}$

We start with computing an adequate, theoretical lower bound for delay $\boldsymbol{\tau}$ and azimuth angle of arrival $\boldsymbol{\phi}$ by evaluation of following posterior of interest which is given in eq. (3.2):

$$\underbrace{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})}_{\text{Posterior}} \propto \underbrace{p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma})}_{\text{Likelihood}} \times \underbrace{p(\boldsymbol{w}|\boldsymbol{\alpha}) \times p(\boldsymbol{\alpha}) \times p(\boldsymbol{\Theta}) \times p(\boldsymbol{\Sigma})}_{\text{Prior}}. \tag{3.38}$$

In order to derive separated results for delay and azimuth angle of arrival, which are summarized in $\boldsymbol{\Theta} = [\boldsymbol{\tau}^T, \boldsymbol{\phi}^T]^T$, we set $\boldsymbol{S}(\boldsymbol{\Theta}) := \boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})$. Since we are only interested in evaluating the posterior in terms of $\boldsymbol{\Theta}$, eq. (3.38) reduces to

$$p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{\Theta}), \tag{3.39}$$

where we assume $p(\boldsymbol{\Theta})$ as a dirac distribution. Thus evaluation of the posterior is straightforward by using the classical maximum likelihood approach, due to eq. (3.39) depending on the likelihood only, when we restrict $p(\boldsymbol{\Theta})$ being a dirac distribution.

$$
\begin{aligned}
\log\Big\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})\Big\} &\propto \log\Big\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\Big\} \\
&\propto \log\Big\{\exp\Big[-\left(\boldsymbol{y}-\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w}\right)^H\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{y}-\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w}\right)\Big]\Big\} \\
&= -\left(\boldsymbol{y}-\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w}\right)^H\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{y}-\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w}\right) \\
&= -\boldsymbol{y}^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y}+(\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y}-(\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w}+ \\
&\quad \boldsymbol{y}^H\boldsymbol{\Sigma}^{-1}\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w} \;:=\; \ell(\boldsymbol{\tau},\boldsymbol{\phi}|\boldsymbol{y}) \qquad (3.40)
\end{aligned}
$$

Computing the Fisher-Information matrix for the delay $[\mathcal{I}(\boldsymbol{\tau})]_{ij}$ and azimuth angle of arrival $[\mathcal{I}(\boldsymbol{\phi})]_{ij}$ is straightforward by

$$
[\mathcal{I}(\boldsymbol{\tau})]_{ij} = -\mathbb{E}\left\{\frac{\partial\ell(\boldsymbol{\tau},\boldsymbol{\phi}|\boldsymbol{y})}{\partial\tau_i\partial\tau_j}\right\} \qquad \text{and} \qquad [\mathcal{I}(\boldsymbol{\phi})]_{ij} = -\mathbb{E}\left\{\frac{\partial\ell(\boldsymbol{\tau},\boldsymbol{\phi}|\boldsymbol{y})}{\partial\phi_i\partial\phi_j}\right\}.
$$

Since we do only consider the likelihood to derive the CRLB's, the results are standard in signal processing which can be found f.e. in Kay (1993),Tschudin et al. (1998) and Antreich et al. (2008):

$$
\begin{aligned}
[\mathcal{I}(\boldsymbol{\tau})]_{ij} &= \boldsymbol{w}^H\boldsymbol{c}(\boldsymbol{\phi})^H\boldsymbol{u}(\boldsymbol{\tau})^H(-j\omega_n)^H\boldsymbol{\Sigma}^{-1}(-j\omega_n)\boldsymbol{u}(\boldsymbol{\tau})\boldsymbol{c}(\boldsymbol{\phi})\boldsymbol{w} \\
[\mathcal{I}(\boldsymbol{\phi})]_{ij} &= \boldsymbol{w}^H(2j\pi\sin(\boldsymbol{\phi})\boldsymbol{c}(\boldsymbol{\phi})\odot\boldsymbol{p}_n)^H\boldsymbol{u}(\boldsymbol{\tau})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{u}(\boldsymbol{\tau})(2j\pi\sin(\boldsymbol{\phi})\boldsymbol{c}(\boldsymbol{\phi})\odot\boldsymbol{p}_n)\boldsymbol{w}
\end{aligned} \qquad (3.41)
$$

where $\boldsymbol{p}_n$ denoting the antenna positions and ω_n as representative for the frequency axis.

3.5.2 Lower Bound for Complex Amplitude $\boldsymbol{w}$

The derivation of a lower bound for the complex amplitude $\boldsymbol{w}$ can be done in an analogue way as for delay $\boldsymbol{\tau}$ and azimuth angle of arrival $\boldsymbol{\phi}$. Consider the posterior given in eq. (3.2) and with its supposed likelihood (3.3) and priors (3.4). In contrast to section 3.5.1, we do make prior assumptions i.e, we assume

$$
p(\boldsymbol{w}|\boldsymbol{\alpha}) \sim \mathcal{CN}\left(\boldsymbol{0},\operatorname{diag}(\boldsymbol{\alpha})^{-1}\right).
$$

Therefore, the posterior of interest (in terms of $\boldsymbol{w}$) factors as follows

$$p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha}),$$

and the inference on this probability density function is carried out by using a Bayesian approach: In the case at hand the posterior density function, which includes the Bayesian prior $p(\boldsymbol{w}|\boldsymbol{\alpha})$, is the one to evaluate. Therefore, the desired Bayesian Cramer-Rao Lower Bound (BCRLB) $[\mathcal{I}(\boldsymbol{w})]_{ij}^{-1}$ must not be conflicted with the CRLB in the frequentistic approach when using the likelihood function only. The respective BCRLB can be derived in two ways: First, just using the posterior density function and second, computing the frequentistic Fisher-Information matrix for likelihood and prior separately plus adding both results (van Trees and Bell (2007), Tichavský et al. (1998)). In the following we will keep to the first proposal.

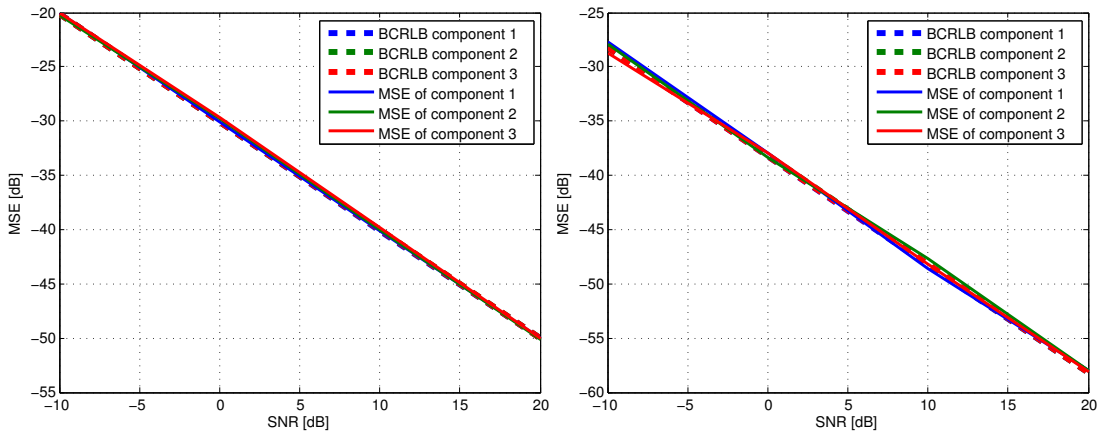
$$\begin{aligned}
\log\Big\{p(\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}|\boldsymbol{y})\Big\} &\propto \log\Big\{p(\boldsymbol{y}|\boldsymbol{\Theta}, \boldsymbol{w}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}) \times p(\boldsymbol{w}|\boldsymbol{\alpha})\Big\} \\
&\propto \log\Big\{\exp\Big[-\left(\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}\right)^H \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}\right)\Big]\times \\
&\qquad \exp\left[-\left(\boldsymbol{w}^H(\mathrm{diag}(\boldsymbol{\alpha})^{-1})^{-1}\boldsymbol{w}\right)\right]\Big\} \\
&= -\left(\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}\right)^H \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}\right) - \boldsymbol{w}^H \mathrm{diag}(\boldsymbol{\alpha})\boldsymbol{w} \\
&= -\boldsymbol{y}^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y} + \boldsymbol{w}^H\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y} - \boldsymbol{w}^H\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w} + \\
&\qquad \boldsymbol{y}^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w} - \boldsymbol{w}^H\mathrm{diag}(\boldsymbol{\alpha})\,\boldsymbol{w} \;:=\; \ell(\boldsymbol{w}|\boldsymbol{y}),
\end{aligned} \tag{3.42}$$

and computation of $[\mathcal{I}(\boldsymbol{w})]_{ij}^{-1}$ is straightforward:

$$\begin{aligned}
[\mathcal{I}(\boldsymbol{w})]_{ij}^{-1} &= \Big(-\mathbb{E}\left\{\frac{\partial \ell(\boldsymbol{w}|\boldsymbol{y})}{\partial w_i^* \partial w_j}\right\}\Big)^{-1} \\
&= \Big(-\mathbb{E}\left\{\frac{\partial}{\partial w_j}\left[\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{y} - \boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w} - \mathrm{diag}(\boldsymbol{\alpha})\boldsymbol{w}\right]\right\}\Big)^{-1} \\
&= \Big(-\mathbb{E}\Big\{-\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) - \mathrm{diag}(\boldsymbol{\alpha})\Big\}\Big)^{-1} = \\
&= \Big(\boldsymbol{S}(\boldsymbol{\Theta})^H\boldsymbol{\Sigma}^{-1}\boldsymbol{S}(\boldsymbol{\Theta}) + \mathrm{diag}(\boldsymbol{\alpha})\Big)^{-1}.
\end{aligned} \tag{3.43}$$

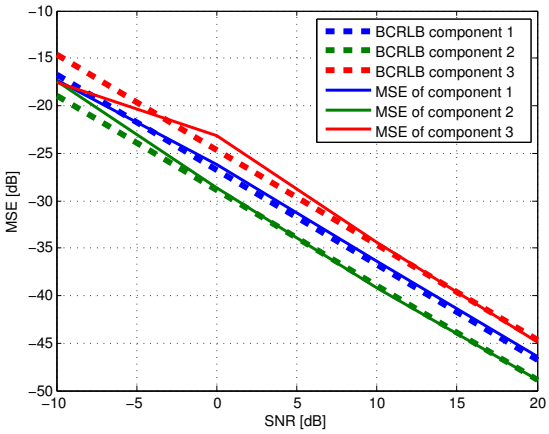
3.5.3 Simulations

For verification of VB-SAGE, we study the performance of the proposed estimation scheme using synthetic data generated accordingly to model (3.1). We assume a static scenario at a fixed snapshot time t and non-moving senders and receiver. The receiver is equipped by a 3-element linear antenna array along the x-direction, where an inter-element spacing of 0.45λ is applied with λ is denoting the wavelength. The carrier frequency has been fixed to 500 MHz and the bandwidth of the transmitted signal to 100 MHz. We use 5000 simulations witch each 101 samples per antenna. The number of existing multipath components is set to 3, with time delays $\boldsymbol{\tau} = [50.83, 37.23, 37.71]$ in samples and azimuth angle of arrival $\boldsymbol{\phi} = [0.17, 0.21, 0.22]$ in radians. Finally all propagation paths are assumed to have equal power.



(a) Complex amplitude

(b) Delay



(c) Azimuth angle of arrival

Figure 4: Comparison of theoretical lower bounds and VB-SAGE mean squared error for complex amplitude, delay and azimuth angle of arrival

Fig. 4 visualizes the resulting performance of VB-SAGE by a graphical comparison of the two MSE definitions in the last section. Note that we are only considering the VB-SAGE update algorithm for performance check by initializing with the true parameter setting. While the dashed lines are representatives for the theoretical (Bayesian) Cramer-Rao Lower Bounds (BCRLB) derived in eqs. (3.41) - (3.43), when taking the true parameter values as a basis, the continuous lines stand for the mean squared error $\mathrm{MSE}(\hat{\boldsymbol{\vartheta}}) = \mathbb{E}(|\widehat{\boldsymbol{\vartheta}} - \boldsymbol{\vartheta}|^2)$ with $\boldsymbol{\vartheta} \in \{\boldsymbol{w}, \boldsymbol{\tau}, \boldsymbol{\phi}\}$ defined as squared difference of estimated and true parameter values.

The results show that at least for the complex amplitude and delay the VB-SAGE update algorithm hits the respective lower bounds. When considering the case for azimuth angle of arrival (in lower SNR levels), the graphic show divergence for component 2 (green) and component 3 (red). At first glance this result seems to argue against good estimation performance of the VB-SAGE algorithm. However, Tschudin (1999) shows that this outcome is not an exceptional circumstance, if two multipath components cannot clearly be distinguished in their azimuth angle of arrival value (in fact components 2 and 3 are similar with values of 0.21 and 0.22). In particular since at least for the first component the VB-SAGE estimation even hits the CRLB, we do not apply hypercritical standards for this case.

4 Tracking of Multipath Components

4.1 Motivation

The VB-SAGE framework (see chapter 3) is a suggested solution for estimation of the model order L and its corresponding signal parameters when restricting to fixed snapshot situations. In a world with a moving receiver which we will assume in the following, the scenario changes to a dynamic one with multiple snapshots, where the estimation of signal parameters is known as tracking.
A straightforward approach for estimation of the unknown parameters is just to apply the VB-SAGE algorithm for all consecutive time steps. But there is a snag in performing the estimation process this way. The VB-SAGE framework itself is not able to use information of past estimation results, because the initialization step starts without any input except the current measurements. It is easily to understand that all prior information, which is available, should be included in any case in order to increase the accurateness of the estimates.
The receiver moves with a certain and especially known speed towards a given direction. This means that both the number of components L and the signal parameters do not change randomly in the space of two subsequent time steps but are highly correlated. By way of example suppose a delay $\tau_l(t)$ for one signal component at snapshot time t. While the receiver is slowly changing its position, $\tau_l(t)$ does only marginally change. Thus, the difference of the subject delay $\tau_l(t)$ and $\tau_l(t+1)$ is limited in value as well, if we fix the position of the sender. As a result we have to implement a principle which allows using the prior information within the tracking process to reduce the imprecision of the estimates.
Therefore, we will go on with introducing the discrete Kalman Filter, which was first published by Rudolf E. Kalman in 1960 to provide a widely-used method that covers the requirements for including prior information to the estimation scheme and adopt it to the multipath channel estimation problem. Subsequent an extension of the Kalman Filter is illustrated as a smoothing algorithm, which additional allows to use measurements derived later than the current time point to obtain information about the signal parameters. This is to smooth the estimates in order to reduce the noise part in order to achieve better performance. Finally the chapter closes with a summary of the tracking algorithm for multipath components.

4.2 The Discrete Kalman Filter

The Kalman filter performs by using a form of feedback control: it estimates the signal parameters at some arbitrary time and then obtains feedback in the form of noisy measurements. As such, the Kalman filter equations can be separated into two groups: time update equations and measurement update equations. The former can also be thought of as predictor equations, while the latter can be thought of as update equations. On the one hand time update equations are responsible

for projecting forward (in time) the current signal parameters and its error covariance estimates at snapshot $t-1$ to get the a priori estimates for the next time step t. On the other hand the measurement update equations are responsible for the feedback mechanism. In particular new noisy measurements are included into the a priori estimate to receive an improved a posteriori estimate. We will consider the case of filtering the multipath components problem shortly, but we must first set up the basic Kalman filter recursive equations, whose derivation can be found for example in Brown and Hwang (1985) as well as Anderson and Moore (2005).

At this point we assume that we have an initial estimate of the process at some point in time $t-1$, which is based on all our knowledge about the process prior to $t-1$. This a priori estimation is denoted as $\widehat{\boldsymbol{x}}(t-1)$ for the process state vector and $\widehat{\boldsymbol{P}}(t-1)$ for its covariance structure respectively. The time update equations or prediction equations for the time step t are given by

$$\begin{aligned} &\text{State vector prediction:} && \widehat{\boldsymbol{x}}^{-}(t) = \boldsymbol{A}\widehat{\boldsymbol{x}}(t-1) \\ &\text{Covariance structure prediction:} && \widehat{\boldsymbol{P}}^{-}(t) = \boldsymbol{A}\widehat{\boldsymbol{P}}(t-1)\boldsymbol{A}^{T} + \boldsymbol{Q}, \end{aligned} \tag{4.1}$$

where the 'super minus' is a reminder that this is our best estimate prior to assimilating the measurements at t. Furthermore $\boldsymbol{A}$ stands for the transition matrix which can be interpreted in what physical way the state vector and the respective covariance structure do change from the initial time step to the next one. Finally $\boldsymbol{Q}$ denotes the system noise matrix which gives information about the underlying noise variance of the system model. These equations give the best prediction of the state vector and its relating covariance matrix without including any new information.

With the assumption of $\widehat{\boldsymbol{x}}^{-}(t)$, we now seek to use the new measurement $\boldsymbol{y}(t)$ to improve the state vector prediction.

$$\begin{aligned} &\text{State vector update:} && \widehat{\boldsymbol{x}}(t) = \widehat{\boldsymbol{x}}^{-}(t) + \widehat{\boldsymbol{K}}(t)\left(\boldsymbol{y}(t) - \boldsymbol{H}\widehat{\boldsymbol{x}}^{-}(t)\right) \\ &\text{Covariance structure update:} && \widehat{\boldsymbol{P}}(t) = \left(\boldsymbol{I} - \widehat{\boldsymbol{K}}(t)\boldsymbol{H}\right)\widehat{\boldsymbol{P}}^{-}(t), \end{aligned} \tag{4.2}$$

with the so called Kalman gain

$$\widehat{\boldsymbol{K}}(t) = \widehat{\boldsymbol{P}}(t)\boldsymbol{H}^{T}\left(\boldsymbol{H}\widehat{\boldsymbol{P}}(t)\boldsymbol{H}^{T} + \widehat{\boldsymbol{R}}(t)\right)^{-1}, \tag{4.3}$$

and a measurement matrix $\boldsymbol{H}$ as well as a the measurement noise covariance structure $\widehat{\boldsymbol{R}}(t)$. Fig. 5 provides the underlying Bayesian Network (or Hidden Markov Model).

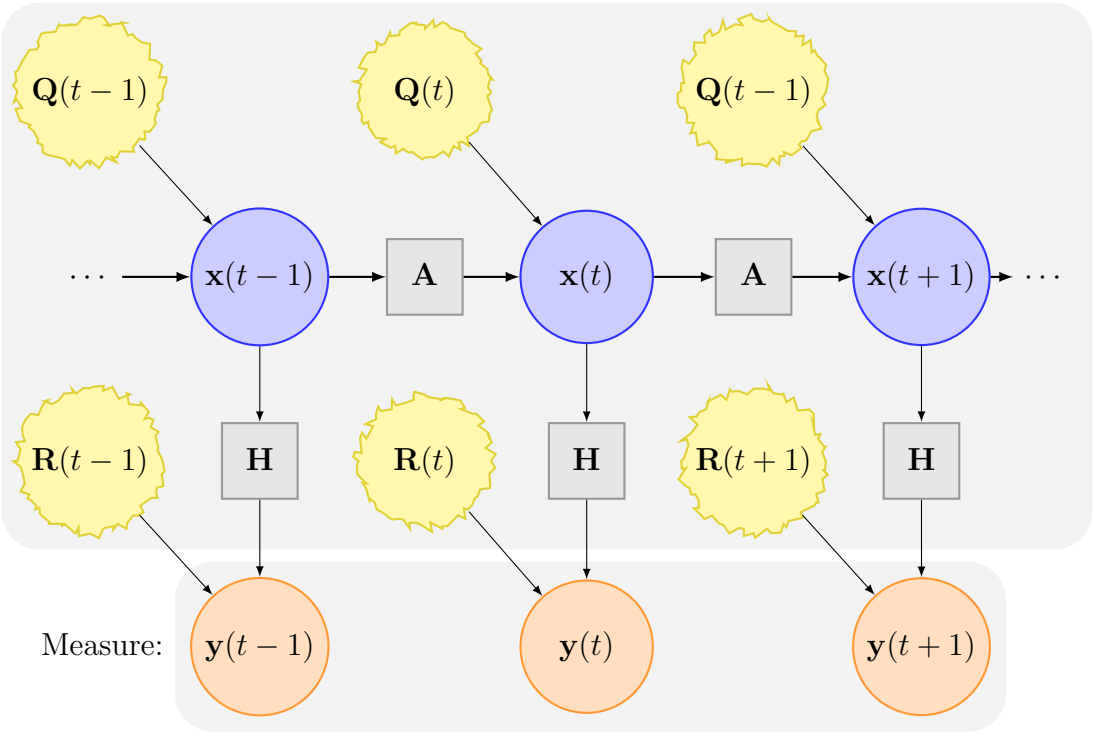


Figure 5: Underlying Bayesian Network

At each time step the state vector $\mathbf{x}(t-1)$ is propagated to the new state estimation $\widehat{\boldsymbol{x}}^{-}(t)$ by multiplication with the constant state transition matrix $\mathbf{A}$. Additionally, the states $\mathbf{x}(t)$ are influenced by the system noise matrix $\mathbf{Q}$. The system state cannot be measured directly. The measurements $\mathbf{y}(t)$ consists of the information contained within the state vector $\mathbf{x}(t)$ multiplied by the measurement matrix $\mathbf{H}$, and the additional measurement noise variance $\mathbf{R}(t)$.

The following two sections are meant to remove possible ambiguity or confusion with respect to the Kalman system model. Therefore, the Kalman equations in the case of signal modeling are set up and involved variables are explained in more detail.

4.2.1 Kalman Prediction Equations

Compared to the one-snapshot based view of the VB-SAGE framework, the considered Kalman Filter should not only provide results for complex amplitude but also separated for magnitude and phase. Additionally a tracking of the noise variance σ^2 over time is included, in order to increase the accurateness of the estimates. Therefore, some changes in implementation will occur which we are going to be explained in detail at appropriate location. For simplicity the states $\{\boldsymbol{a}(t), \boldsymbol{\varphi}(t), \boldsymbol{\tau}(t), \boldsymbol{\phi}(t), \sigma^2(t)\}$ of magnitude, phase, delay, azimuth angle of ar-

rival and noise variance and the differences between two consecutive time steps $\{\Delta\boldsymbol{a}(t), \Delta\boldsymbol{\varphi}(t), \Delta\boldsymbol{\tau}(t), \Delta\boldsymbol{\phi}(t)\}$ are represented in the condensed state vector form $\boldsymbol{\zeta}(t)$. Note that since L multipath components are assumed, the dimension of $\boldsymbol{\zeta}$ is $(8L + 1 \times 1)$. Furthermore the differences are all set to zero for the initial time step $t = 1$.

The Kalman predictions in eq. (4.1) can then be reformulated as

$$\begin{aligned} &\text{State vector prediction:} && \widehat{\boldsymbol{\zeta}}^{-}(t) = \boldsymbol{A}\widehat{\boldsymbol{\zeta}}(t-1) \\ &\text{Covariance structure prediction:} && \widehat{\boldsymbol{P}}^{-}(t) = \boldsymbol{A}\widehat{\boldsymbol{P}}(t-1)\boldsymbol{A}^{T} + \boldsymbol{Q}. \end{aligned} \tag{4.4}$$

In what follows we have to make assumptions about the shape of the transition matrix $\boldsymbol{A}$ and the system noise variance $\boldsymbol{Q}$. Therefore, we will use expressions that turned out to work quite well, as applied in Jost, Wang, Fiebig and Pérez-Fontán (2012). For a start regard the state vector prediction with the initial states $\widehat{\boldsymbol{\zeta}}(t-1)$ which are transformed by the transition matrix $\boldsymbol{A}$ to the next time step prediction $\widehat{\boldsymbol{\zeta}}^{-}(t)$, what can be stated in extensive form. For simplicity we assume the one component case, thus $l = 1$:

$$\begin{pmatrix} a(t) \\ \varphi(t) \\ \tau(t) \\ \phi(t) \\ \sigma^2(t) \\ \Delta a(t) \\ \Delta\varphi(t) \\ \Delta\tau(t) \\ \Delta\phi(t) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} a(t-1) \\ \varphi(t-1) \\ \tau(t-1) \\ \phi(t-1) \\ \sigma^2(t-1) \\ \Delta a(t-1) \\ \Delta\varphi(t-1) \\ \Delta\tau(t-1) \\ \Delta\phi(t-1) \end{pmatrix}$$

$$= \begin{pmatrix} a(t-1) \\ \varphi(t-1) + T\Delta\varphi(t-1) \\ \tau(t-1) + T\Delta\tau(t-1) \\ \phi(t-1) \\ \sigma^2(t-1) \\ \Delta a(t-1) \\ \Delta\varphi(t-1) \\ \Delta\tau(t-1) \\ \Delta\phi(t-1) \end{pmatrix}. \tag{4.5}$$

As result, the state predictions for magnitude, azimuth angle of arrival and noise variance do only depend on their respective state values of the initial time step $t-1$, for phase and delay a change by $T\Delta\boldsymbol{\varphi}(t-1)$ and $T\Delta\boldsymbol{\tau}(t-1)$ for physical reasons is additional included.

In the same manner we are able to set up the prediction equation for the covariance state structure by adding the system noise variance matrix $\boldsymbol{Q}$, which is assumed to be fixed at every snapshot time. For clarity issues we do resign for pure broad mathematical presentation, the interpretation can be adopted from the state vector prediction in analogue way.

4.2.2 Kalman Update Equations

Now let's inspect the Kalman Filters update equations in the case of signal modeling. With using the Kalman predictions given in eq. (4.4) we now seek to include the new measurement $\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t)$, representing the VB-SAGE estimations referred to chapter 3.4 made out of the received signal vector $\boldsymbol{y}(t)$. Thereby we use the predictions $\widehat{\boldsymbol{\zeta}}^{-}(t)$ and $\widehat{\boldsymbol{P}}^{-}(t)$ as input for updating the the VB-SAGE estimates $\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t)$. That guarantees for finding the next local maximas what ensures an automatic association of multipath components, i.e. of the prediction $\widehat{\boldsymbol{\zeta}}^{-}(t)$ and the new measurements $\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t)$. Since we split up the random variable for complex amplitude $\boldsymbol{w}(t)$ into magnitude $\boldsymbol{a}(t)$ and phase $\boldsymbol{\varphi}(t)$ within the Kalman Filter framework, the respective VB-SAGE state estimation $\widehat{\boldsymbol{w}}(t)$ has to be adjusted, what can be easily achieved by transformation. Remember the Kalman update equations for state vector and covariance structure:

State vector update: $$\widehat{\boldsymbol{\zeta}}(t) = \widehat{\boldsymbol{\zeta}}^{-}(t) + \widehat{\boldsymbol{K}}(t)\left(\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t) - \boldsymbol{H}\widehat{\boldsymbol{\zeta}}^{-}(t)\right)$$

Covariance structure update: $$\widehat{\boldsymbol{P}}(t) = \left(\boldsymbol{I} - \widehat{\boldsymbol{K}}(t)\boldsymbol{H}\right)\widehat{\boldsymbol{P}}^{-}(t)$$

with the Kalman gain: $$\widehat{\boldsymbol{K}}(t) = \widehat{\boldsymbol{P}}(t)\boldsymbol{H}^{T}\left(\boldsymbol{H}\widehat{\boldsymbol{P}}(t)\boldsymbol{H}^{T} + \widehat{\boldsymbol{R}}(t)\right)^{-1}. \qquad (4.6)$$

Since we already explained the intuition of the VB-SAGE estimations $\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t)$, the Kalman predictions $\widehat{\boldsymbol{\zeta}}^{-}(t)$, $\widehat{\boldsymbol{P}}^{-}(t)$, we need to survey the measurement matrix $\boldsymbol{H}$, the Kalman gain $\widehat{\boldsymbol{K}}(t)$ which includes for the additional measurement noise variance $\widehat{\boldsymbol{R}}(t)$ corresponding to $\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t)$. The measurement matrix is assumed to be identical for every time instant t, thus $\boldsymbol{H}$ is fixed. Referring to Jost, Wang, Fiebig and Pérez-Fontán (2012) $\boldsymbol{H}$ factors, by assuming the one component case again, as follows, since only the states but not their differences are estimated by the VB-SAGE algorithm:

$$\boldsymbol{H} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Next, we will have a look on the Kalman gain $\widehat{\boldsymbol{K}}(t)$. We will pass on the derivation (therefore, see f.e. Brown and Hwang (1985) as well as Anderson and Moore (2005)), but at least a brief interpretation should be given. Before we move on, let's look at this equations in detail. First note that the gain changes at every iteration. For this reason it should be written with a subscript (i.e., t). Next, and more significantly, we can examine what happens as each of the two non-fixed terms $\widehat{\boldsymbol{P}}(t)$ and $\widehat{\boldsymbol{R}}(t)$ in eq. (4.6) are varied.
If the a priori estimation error expressed by the state covariance matrix prediction $\widehat{\boldsymbol{P}}(t)$ is very small compared to $\widehat{\boldsymbol{R}}(t)$, $\widehat{\boldsymbol{K}}(t)$ is correspondingly very small. Thus we will ignore the current VB-SAGE estimates and simply use past estimates to form the update. In other words, we keep more trust in the past estimates than in the new VB-SAGE measurements. Vice versa, if $\widehat{\boldsymbol{P}}(t)$ is very large (so that the measurement noise term $\widehat{\boldsymbol{R}}(t)$ is unimportant) then $\widehat{\boldsymbol{K}}(t) \approx \boldsymbol{H}^{-1}$. This, in effect, tells us to throw away the a priori estimates and use the current (measured) VB-SAGE values to estimate the states. In other words again, we have more trust in the new VB-SAGE measurements than in the past measurements.

However, there is still a problem because an adequate estimation for the a priori measurement noise covariance matrix $\widehat{\boldsymbol{R}}(t)$ is needed within the Kalman gain expression. Therefore, our next task will be to come up with an estimation for that. In that case we can make use of former results given in section 3.5. Simulations showed consistency of the variances of the VB-SAGE estimations for complex amplitude, delay, and azimuth angle of arrival with the (Bayesian) CRLB determined by the respective information matrix.
This conception needs only to be extended to the covariance case (with replacing the complex amplitude variance with their magnitude and phase values) plus including the noise variance σ^2. Therefore, we evaluate the logarithmic posterior given in eq. (3.2) with respect to $\widetilde{\boldsymbol{\zeta}}(t) := \{\boldsymbol{a}(t), \boldsymbol{\varphi}(t), \boldsymbol{\tau}(t), \boldsymbol{\phi}(t), \sigma^2(t)\}$. Note that compared to $\boldsymbol{\zeta}(t)$, we do not consider the state differences in $\widetilde{\boldsymbol{\zeta}}(t)$. Since we do not assume a special structure on the prior noise variance distribution $p(\sigma^2)$, this part can be ignored. The derivation of the covariance matrix $\widehat{\boldsymbol{R}}(t)$ is straightforward by computing the respective information matrix $[\mathcal{I}(\boldsymbol{\zeta})]_{ij}$. Note that we do not calculate the expectation here, since only one measurement is available at each time instant t.

$$\left[\mathcal{I}(\widetilde{\boldsymbol{\zeta}})\right]_{ij} = -\left\{\frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial \widetilde{\boldsymbol{\zeta}}_i^* \partial \widetilde{\boldsymbol{\zeta}}_j}\right\}$$

$$= -\left(\frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial a_i^* \partial a_j} \quad \frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial \varphi_i^* \partial \varphi_j} \quad \frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial \tau_i \partial \tau_j} \quad \frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial \phi_i \partial \phi_j} \quad \frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial (\sigma^2)_i^* \partial (\sigma^2)_j}\right)^T,$$

with

$$\frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial \tau_i \partial \tau_j}, \frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial \phi_i \partial \phi_j} \quad \text{from (3.41)},$$

and

$$\begin{aligned}\frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial a_i^* \partial a_j} = \frac{\partial}{\partial a_j}\Big[&(\exp(j\boldsymbol{\varphi}))^H \boldsymbol{S}(\boldsymbol{\Theta})^H (\sigma^2)^{-1} \boldsymbol{y} - \\ &(\exp(j\boldsymbol{\varphi}))^H \boldsymbol{S}(\boldsymbol{\Theta})^H (\sigma^2)^{-1} \boldsymbol{S}(\boldsymbol{\Theta})(\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l)) - \\ &(\exp(j\boldsymbol{\varphi}))^H \boldsymbol{\alpha}\,(\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l))\Big] \\ = -(\exp(j\boldsymbol{\varphi}))^H &\boldsymbol{S}(\boldsymbol{\Theta})^H (\sigma^2)^{-1} \boldsymbol{S}(\boldsymbol{\Theta})(\exp(j\boldsymbol{\varphi})) - (\exp(j\boldsymbol{\varphi}))^H \boldsymbol{\alpha}(\exp(j\boldsymbol{\varphi})),\end{aligned} \tag{4.7}$$

$$\begin{aligned}\frac{\partial \ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial \varphi_i^* \partial \varphi_j} = \frac{\partial}{\partial \varphi_j}\Big[&(j\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l))^H \boldsymbol{S}(\boldsymbol{\Theta})^H (\sigma^2)^{-1} \boldsymbol{y} - \\ &(j\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l))^H \boldsymbol{S}(\boldsymbol{\Theta})^H (\sigma^2)^{-1} \boldsymbol{S}(\boldsymbol{\Theta})(\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l)) - \\ &(j\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l))^H \boldsymbol{\alpha}(\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l))\Big] \\ = -(j\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l))^H &\boldsymbol{S}(\boldsymbol{\Theta})^H (\sigma^2)^{-1} \boldsymbol{S}(\boldsymbol{\Theta})(j\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l)) - \\ &(j\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l))^H \boldsymbol{\alpha}(j\boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l)),\end{aligned} \tag{4.8}$$

where we take the logarithmic posterior (3.42) and define $\boldsymbol{w} = \boldsymbol{a}_l \exp(j\boldsymbol{\varphi}_l)$ as a component-wise multiplication for calculation of eqs. (4.7), (4.8).

Obviously in eqs. (4.7) and (4.8) one has to respect the prior $p(\boldsymbol{w}|\boldsymbol{\alpha})$ since magnitude and phase are transformations of $\boldsymbol{w}$. Therefore, the resulting cells of the inverse information matrix $[\mathcal{I}(\widetilde{\boldsymbol{\zeta}})]_{ij}^{-1}$ must be interpreted as Bayesian Cramer-Rao Bounds.

Finally we need to derive the expression for the noise variance, where we first set up the respective log-likelihood

$$\ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y}) = \Big\{ -\log\big\{(\sigma^2)^N\big\} - (\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})^H(\sigma^2)^{-1}(\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})\Big\},$$

and continue with calculation the final expression

$$\begin{aligned}\frac{\partial\ell(\widetilde{\boldsymbol{\zeta}}|\boldsymbol{y})}{\partial(\sigma^2)_i^*\partial(\sigma^2)_j} &= \frac{\partial}{\partial(\sigma^2)_j}\Big[-N(\sigma^2)^{-1} + (\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})^H(\sigma^2)^{-2}(\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})\Big] \\ &= N(\sigma^2)^{-2} - (\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w})^H(\sigma^2)^{-3}(\boldsymbol{y}-\boldsymbol{S}(\boldsymbol{\Theta})\boldsymbol{w}).\end{aligned}$$

Note that the resulting information matrix $[\mathcal{I}(\widetilde{\boldsymbol{\zeta}})]_{ij}$ has dimension $[(4L+1)\times 4L+1)]$ due to magnitude, phase, delay and azimuth angle of arrival values are calculated for each multipath component $l = 1, ..., L$, the noise variance value only once. The target measurement noise covariance matrix $\widehat{\boldsymbol{R}}(t)$ is derived in a well known manner as $[\mathcal{I}(\widetilde{\boldsymbol{\zeta}})]_{ij}^{-1}$.

4.2.2.1 Adjustments for Model Order Change

At this point one may think we are able to put together all the puzzle pieces of Kalman prediction and update equations and perform the Kalman Filter for consecutive snapshots. However, it will become apparent that the present version of Kalman does not allow to change the model order L within two time instants. Both Kalman prediction and Kalman update equations are only applicable for a fixed model order. Before moving on with criteria to decide whether new paths are added or removed, one has to set up the adjusted Kalman equations.

Assume the case when one multipath component is removed from the model within two consecutive time steps, thus $L(t) = L(t-1) - 1$. Since the decision about eliminating the respective propagation path at t can only be made out of the new measurements $\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t)$, the Kalman prediction in eq. (4.4) does not change in the beginning. After the decision process has been performed one simply corrects $\widehat{\boldsymbol{\zeta}}^-(t)$ and $\widehat{\boldsymbol{P}}^-(t)$ for the corresponding entries of the 'killed' component by $\widehat{\boldsymbol{\zeta}}(t-1) = (\widehat{\boldsymbol{\zeta}}(t-1))_{\bar{l}}$, and $\widehat{\boldsymbol{P}}(t-1) = (\widehat{\boldsymbol{P}}(t-1))_{\overline{ll}}$, where $(\widehat{\boldsymbol{\zeta}}(t-1))_{\bar{l}}$,

$(\widehat{\boldsymbol{P}}(t-1))_{\overline{ll}}$ denoting the state vector and covariance structure update at $t-1$ without respecting the removed component l.
In the same way one adjusts the Kalman update. Note that the VB-SAGE update measurements $\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t)$ and the respective noise variance $\widehat{\boldsymbol{R}}(t)$ already show correct dimensions since the removing criterion is performed prior to the VB-SAGE update step.

Let's now consider the case when a now component is initialized at t, thus $L(t) = L(t-1) + 1$. The Kalman prediction equations do change (after evaluation the new measurements) in the identical manner as demonstrated in the case when one multipath component was removed, i.e. we correct $\widehat{\boldsymbol{\zeta}}^{-}(t)$ and $\widehat{\boldsymbol{P}}^{-}(t)$ for the new component by including the VB-SAGE measurements of the new multipath for both state vector and covariance structure respectively.
Finally, the adjusted Kalman update equations can be derived analogue to the former case by eliminating one component.

After these mathematical adaptions are described, we can now think about an implementation strategy for removing and adding multipath components to the Kalman Filter in the next sections.

4.2.2.2 Test to Remove Paths

The VB-SAGE algorithm provides a model selection criteria (see section 3.4.5) inside its implementation, i.e. it is possible to decide if an estimated propagation path is merely fictive and so to remove from the model or does actually exist and should be included in the estimation process. This idea could be adopted in the Kalman Filter as well.

However, a different approach is presented for following reasons: The VB-SAGE model selection criteria does only depend on one single snapshot without any input of former time point results. Since the Kalman Filter gives a prediction for both complex amplitude and its corresponding noise variance we should use this information. The aim is to guarantee better performance of the model selection criteria, which is based on statistical hypothesis testing, i.e. decision theory, used in the field of signal processing f.e. in Salmi et al. (2009).

The basic idea is quite simple and can be formulated as a binary choice problem, explained in detail f.e in Hänsler (2001)

$$H_0 : \widehat{w}_l \sim \mathcal{CN}(0, (\sigma^{-})^2) \qquad \text{vs.} \qquad H_1 : \widehat{w}_l \sim \mathcal{CN}(\widehat{w}_l^{-}, (\sigma_{w_l}^{-})^2) \tag{4.9}$$

where $(\sigma^-)^2$ stands for the Kalman prediction of the noise variance and $\widehat{w}_l^-$, $(\sigma_{w_l}^-)^2$ for the Kalman prediction of mean and variance of the complex amplitude for the considered component l.

In statistical detection, one works with a favored assumption. This favored assumption is called the null hypothesis, which is denoted by H_0. The 'alternative' to the null hypothesis is, naturally, called the alternative hypothesis H_1. In this context we state H_0 in that way meaning the VB-SAGE update of the complex amplitude does not account for improving our model, i.e. $\widehat{w}_l$ does not represent an existing multipath component, the estimated performance is actually pure measurement noise.
In opposition H_1 expresses for the VB-SAGE update $\widehat{w}_l$ having an in fact physical distribution with mean $\widehat{w}_l^-$ and variance $(\sigma_{w_l}^-)^2$ given by Kalman prediction, meaning the component l does de facto exist within the model and should be kept. Therefore, we have to develop a criterion which allows for testing if H_0 or H_1 is accepted.
It should be mentioned that one does not 'prove' anything. Rather, we are only looking at consistency or inconsistency between the VB-SAGE measurement for complex amplitude and the proposed hypotheses. If $\widehat{w}_l$ is consistent with H_0, then instead of proving the hypothesis true, the proper interpretation is that there is no reason to doubt that $\widehat{w}_l$ occurs due to the measurement noise.

Therefore, we define following equation as our testing criterion which is based on the likelihood-ratio:

H_0 holds if:

$$\frac{p\left(\widehat{w}_l|H_0\right)}{p\left(\widehat{w}_l|H_1\right)} \quad = \quad \frac{p\left(\widehat{w}_l|0,(\sigma^-)^2\right)}{p\left(\widehat{w}_l|\widehat{w}_l^-,(\sigma_{w_l}^-)^2\right)} > 1 \tag{4.10}$$

After setting up the actual test criterion we need to compute the distributions $p\left(\widehat{w}_l|H_0\right)$ and $p\left(\widehat{w}_l|H_1\right)$. If we assume the current VB-SAGE estimation $\widehat{w}_l$ belonging to the noise distribution under H_0, the result for the respective distribution is straightforward:

$$p\left(\widehat{w}_l|0,(\sigma^-)^2\right) = \frac{1}{\pi(\sigma^-)^2} \ \exp\left(-\widehat{w}_l^H(\sigma^-)^{-2}\widehat{w}_l\right).$$

It turns out that the derivation of $p\left(\widehat{w}_l|H_1\right)$ is more difficult to treat, since the Kalman Filter does not provide predictions for $\widehat{w}_l^-$ and $(\sigma_{w_l}^-)^2$ directly. Available are predictions for magnitude and phase. Therefore, we have to transform the

random variables to derive the desired distribution. Hereby we will make use of Weickert (2010) within eqs. (4.11) - (4.12).

Therefore, we take $w_l = \Re(w_l) + j\,\Im(w_l)$, where $\Re$, $\Im$ denoting the real and imaginary part, and define

$$\begin{aligned} \Re(\widehat{\boldsymbol{w}}) &:= \boldsymbol{w}_R = T_R(\boldsymbol{a}, \boldsymbol{\varphi}) = \boldsymbol{a}\ \cos(\boldsymbol{\varphi}) \\ \Im(\widehat{\boldsymbol{w}}) &:= \boldsymbol{w}_I\ = T_I(\boldsymbol{a}, \boldsymbol{\varphi})\ = \boldsymbol{a}\ \sin(\boldsymbol{\varphi}). \end{aligned} \tag{4.11}$$

Since the existence of the inversion is proofed in Weickert (2010) we can formulate

$$\begin{aligned} \boldsymbol{a} &= T_R^{-1}(\boldsymbol{w}_R, \boldsymbol{w}_I) = (\boldsymbol{w}_R^2 + \boldsymbol{w}_I^2)^{\frac{1}{2}} \\ \boldsymbol{\varphi} &= T_I^{-1}(\boldsymbol{w}_R, \boldsymbol{w}_I) = \arctan\left(\frac{\boldsymbol{w}_I}{\boldsymbol{w}_R}\right). \end{aligned}$$

Computation of the respective Jacobian matrix

$$\begin{aligned} \boldsymbol{J} &= \begin{pmatrix} \dfrac{\partial T_R^{-1}(\boldsymbol{w}_R, \boldsymbol{w}_I)}{\partial \boldsymbol{w}_R} & \dfrac{\partial T_R^{-1}(\boldsymbol{w}_R, \boldsymbol{w}_I)}{\partial \boldsymbol{w}_I} \\ \dfrac{\partial T_I^{-1}(\boldsymbol{w}_R, \boldsymbol{w}_I)}{\partial \boldsymbol{w}_R} & \dfrac{\partial T_I^{-1}(\boldsymbol{w}_R, \boldsymbol{w}_I)}{\partial \boldsymbol{w}_I} \end{pmatrix} = \begin{pmatrix} \dfrac{\boldsymbol{w}_R}{(\boldsymbol{w}_R^2 + \boldsymbol{w}_I^2)^{\frac{1}{2}}} & \dfrac{\boldsymbol{w}_I}{(\boldsymbol{w}_R^2 + \boldsymbol{w}_I^2)^{\frac{1}{2}}} \\ -\dfrac{\boldsymbol{w}_I}{\boldsymbol{w}_R^2 + \boldsymbol{w}_I^2} & \dfrac{\boldsymbol{w}_R}{\boldsymbol{w}_R^2 + \boldsymbol{w}_I^2} \end{pmatrix} \\ &= \begin{pmatrix} \cos(\boldsymbol{\varphi}) & \sin(\boldsymbol{\varphi}) \\ -\dfrac{\sin(\boldsymbol{\varphi})}{\boldsymbol{a}} & \dfrac{\cos(\boldsymbol{\varphi})}{\boldsymbol{a}} \end{pmatrix} \end{aligned} \tag{4.12}$$

with the determinant of the Jacobian matrix

$$\begin{aligned} |\boldsymbol{J}| &= \cos(\boldsymbol{\varphi})\ \frac{\cos(\boldsymbol{\varphi})}{\boldsymbol{a}} + \sin(\boldsymbol{\varphi})\ \frac{\sin(\boldsymbol{\varphi})}{\boldsymbol{a}} = \frac{(\cos(\boldsymbol{\varphi}))^2 + (\sin(\boldsymbol{\varphi}))^2}{\boldsymbol{a}} \\ &= \frac{1}{\boldsymbol{a}}. \end{aligned}$$

Finally one receives the desired probability density function for the complex amplitude by applying the general case set up in Peebles (2001) to the problem at hand.

$$
\begin{aligned}
f_{\boldsymbol{w}_R,\boldsymbol{w}_I}\left(\boldsymbol{w}_R,\boldsymbol{w}_I\right) &= f_{\boldsymbol{a},\boldsymbol{\varphi}}\left(\boldsymbol{a},\boldsymbol{\varphi}\right)\left|\boldsymbol{J}\right| \\
&= \frac{f_{\boldsymbol{a},\boldsymbol{\varphi}}\left(\left(\boldsymbol{w}_R^2+\boldsymbol{w}_I^2\right)^{\frac{1}{2}},\arctan\left(\frac{\boldsymbol{w}_I}{\boldsymbol{w}_R}\right)\right)}{\boldsymbol{a}} \\
&= \frac{f_{\boldsymbol{a},\boldsymbol{\varphi}}\left(\left(\boldsymbol{a}\cos(\boldsymbol{\varphi})^2+\boldsymbol{a}\sin(\boldsymbol{\varphi})^2\right)^{\frac{1}{2}},\arctan\left(\frac{\sin(\boldsymbol{\varphi})}{\cos(\boldsymbol{\varphi})}\right)\right)}{\boldsymbol{a}} \\
&= \frac{f_{\boldsymbol{a},\boldsymbol{\varphi}}\left(\left(\boldsymbol{a}\right)^{\frac{1}{2}},\boldsymbol{\varphi}\right)}{\boldsymbol{a}}
\end{aligned}
$$

The basic intuition of the test (4.10) is demonstrated by a fictitious graphical representation given in fig. 6. While the blue line is a representative for the physical existing complex amplitude likelihood under H_1, the red line stand for the noise distribution under H_0. If the VB-SAGE update for $\widehat{w}_l$ is given in $[0.7, 1.3]$ (on the x-axis) it is more likely to state H_1 to be true and we will keep the component l, otherwise H_0 is assumed to be true and we will remove the component from the model.

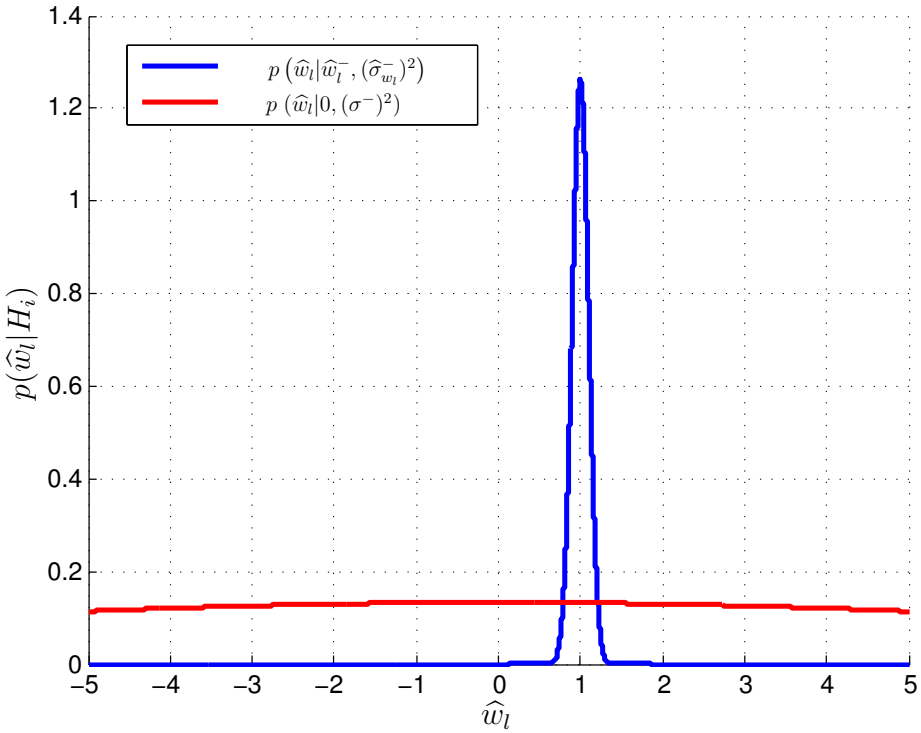


Figure 6: Graphical representation of testing criterion to kill paths within the Kalman Filter

Assume our decision has been performed by applying eq. (4.10) and propagation paths have been killed. We are compelled to correct the dimensions of the Kalman prediction and Kalman update as described in section 4.2.2.1. Since paths have been removed from the model but the estimates of remaining signal components have not been updated, we need to perform the VB-SAGE update algorithm (see page 26) again before going on with the Kalman Filter processing of the following snapshot.

4.2.2.3 Test to Include Paths

For the sake of completeness we seek for a criterion which allows to detect new multipath components, if they appear at a time point t. Unfortunately the presented strategy in section 4.2.2.2 can not be adopted to this case. The Kalman Filter does not provide a prediction for non existing paths, since the prior state vector $\widehat{\boldsymbol{\zeta}}(t-1)$ for the new component is not available. However we are able to define a modified testing strategy by taking the Kalman prediction for the noise variance $(\sigma^-)^2$ into account. Prior to this we have to estimate the complex amplitude of the new component, which has to be tested. Therefore, we extend the VB-SAGE initialization algorithm which now does not operate with a 'bottom-up' strategy as its background algorithm. Thus no empty model is assumed for a start but all existing multipath components with their corresponding signal parameters are taken as input. Thus we fix the number of existing propagation paths to L and set $l = L$. The initialization cycles for new components $l+1, l+2, ...$ are performed in the well-known manner and the complex amplitude for the new component is estimated. We now have to validate if the respective component should be kept by applying following Neyman-Person Test:

$$H_0 : \widehat{w}_l \sim \mathcal{CN}(0, (\sigma^-)^2) \quad \text{vs.} \quad H_1 : \widehat{w}_l \nsim \mathcal{CN}(0, (\sigma^-)^2).$$

If H_0 holds, the component will not be added, since the VB-SAGE estimate $\widehat{w}_l$ is inside the quantile-given-interval of the assumed noise distribution $F_{w_l}(0, (\sigma^-)^2)$ on a significance level (or false alarm rate) of 5 percent.

$$q_{(0.025)} F_{w_l}(0, (\sigma^-)^2) < \widehat{w}_l < q_{(0.975)} F_{w_l}(0, (\sigma^-)^2).$$

After completing the test one needs to correct the dimensions for Kalman prediction and Kalman update as described in section 4.2.2.1 again, followed by performing the VB-SAGE update algorithm for analogue justification as in the case when propagation paths were removed from the model (see section 4.2.2.2).

4.3 Kalman Smoothing

4.3.1 Filtering vs. Smoothing

Smoothing differs from filtering in that the information for estimating the signal parameters at time point t need not become available at t. Thus measurements derived later than t can be used in obtaining information about the parameters at current time point t. This means data can not be processed online, as compared to the filtering case, but the disadvantage has to be weighted against the ability to use more measurements than in the filtering case. For this reason, smoothing may produce more accurate estimates than the simple filtering process (Anderson and Moore (2005)).

In the last sections, we addressed the filtering problem. To proceed, suppose that we are given a set of data over the time interval $0 < t \leq T$. Smoothing is a non-real-time operation in that it involves estimation of the state vector $\boldsymbol{\zeta}(t)$ for $0 < t \leq T$, using all the available data, past as well as future. In what follows, we assume that the final time T is fixed. To determine the state estimates $\widehat{\boldsymbol{\zeta}}(t)$ for $0 < t \leq T$, we need to account for past data $\boldsymbol{y}(p)$ defined by $0 < p \leq t$, and future data $\boldsymbol{y}(k)$ defined by $t < k \leq T$. The estimation pertaining to the past data, which we refer to as forward filtering, was presented in section 4.2. To deal with the issue of state estimation pertaining to the future data, we use backward filtering, which starts at the final time T and runs backwards (Haykin (2001)). Let $\widehat{\boldsymbol{\zeta}}^f(t)$ and $\widehat{\boldsymbol{\zeta}}^b(t)$ denote the state estimates obtained from the forward and backward recursions, respectively. Given these two estimates, the next issue to be considered is how to combine them into an overall smoothed estimate $\widehat{\boldsymbol{\zeta}}(t)$, which accounts for data over the entire time interval. Note that the symbol $\widehat{\boldsymbol{\zeta}}(t)$ used for the smoothed estimate in this section is not to be confused with the filtered (i.e., a posteriori) forward estimate used in former sections.

4.3.2 Forward-backward Kalman Smoothing

A Kalman Smoother uses the regular Kalman filter equations which are employed together with a separate set of equations used for a backward calculation. This backward calculation uses the future measurement data to derive correcting terms for the regular forward Kalman filter estimates. The forward filter is operated in the usual way, giving a posterior state and covariance estimate of the interior point of interest, $\widehat{\boldsymbol{\zeta}}^f(t)$ and $\widehat{\boldsymbol{P}}^f(t)$. The backward running Kalman filter starts with the last measurement, ends on the first measurement, and is initialized with a prior estimate of infinite covariance (which assures that the initial state estimate has no influence). It differs from the forward Kalman filter by performing the prediction stage using the following equations set up in Storve (2012). Note that in this connection, the time index t is used in the other way round as $t = T, ..., 1$.

$$\begin{aligned} &\text{State vector prediction:} && \widehat{\boldsymbol{\zeta}}^{-b}(t) = \boldsymbol{A}^{-1}\widehat{\boldsymbol{\zeta}}^{b}(t-1) \\ &\text{Covariance structure prediction:} && \widehat{\boldsymbol{P}}^{-b}(t) = \boldsymbol{A}^{-1}\left(\widehat{\boldsymbol{P}}^{b}(t-1) + \boldsymbol{Q}\right)\left(\boldsymbol{A}^{-1}\right)^{T}, \end{aligned} \tag{4.13}$$

where $\boldsymbol{A}$, $\boldsymbol{Q}$ are defined for the application to signal processing in the same manner as stated in section 4.2.1. Note that $\boldsymbol{A}^{-1}$ is the inverse of the state-transition matrix used in the forward running Kalman filter in this connection. The update step is performed in the usual way. The backward-running Kalman filter stops at the time step before (in backward direction) the point of interest, afterwards a prediction step is performed. These two estimates $\widehat{\boldsymbol{\zeta}}^{f}(t)$ and $\widehat{\boldsymbol{\zeta}}^{b}(t)$ are then fused by the following covariance intersection algorithm: Given the two estimates $(\widehat{\boldsymbol{\zeta}}^{f}(t), \widehat{\boldsymbol{P}}^{f}(t))$ and $(\widehat{\boldsymbol{\zeta}}^{b}(t), \widehat{\boldsymbol{P}}^{b}(t))$, the fused estimate is computed as in Storve (2012):

$$\begin{aligned} \widehat{\boldsymbol{P}}(t) &= \left((\widehat{\boldsymbol{P}}^{f}(t))^{-1} + (\widehat{\boldsymbol{P}}^{b}(t))^{-1}\right)^{-1} \\ \widehat{\boldsymbol{\zeta}}(t) &= \widehat{\boldsymbol{P}}\left((\widehat{\boldsymbol{P}}^{f}(t))^{-1}\widehat{\boldsymbol{\zeta}}^{f}(t) + (\widehat{\boldsymbol{P}}^{b}(t))^{-1}\widehat{\boldsymbol{\zeta}}^{b}(t)\right). \end{aligned}$$

This is done for all time-steps and yields the smoothed estimates $\widehat{\boldsymbol{\zeta}}(t)$.

4.3.3 Implementation Strategy

For implementation, we will keep to the simplest form of a forward-backward Kalman Smoother, which assumes a fixed and finite set of measurements and is, therefore, restricted to post-processing only. The forward filter is operated in the usual way as stated in section 4.2. The backward filter can be classified as a fixed-interval smoother, where we restrict the backward prediction and update stage on only one snapshot past (in time) the interior point of interest. The aim is solely to evaluate if an additional annexation of a smoothing algorithm can improve the signal parameters estimates. More (computational) complex forms of smoothing strategies are outside the scope of this work, however this can be easily extended if additional benefit is shown.

4.4 Summary of the Algorithm

Finally we can put together all results. In this work a two-stage estimation algorithm is proposed. It is composed of an outer stage keeping track of time-variations in terms of number of paths and paths' parameters, and an inner stage

applying a snapshot-based Bayesian estimator. While the inner stage is performed by using the Variational Bayesian Space-Alternating Generalized Expectation-Maximization (VB-SAGE) algorithm (see section 3), the outer stage makes use of Kalman Filtering plus Kalman Smoothing of the parameters. This approach we denote as Bayesian Estimation and Kalman Tracking/Smoothing (BEKS) algorithm, which is summarized in fig. 7.

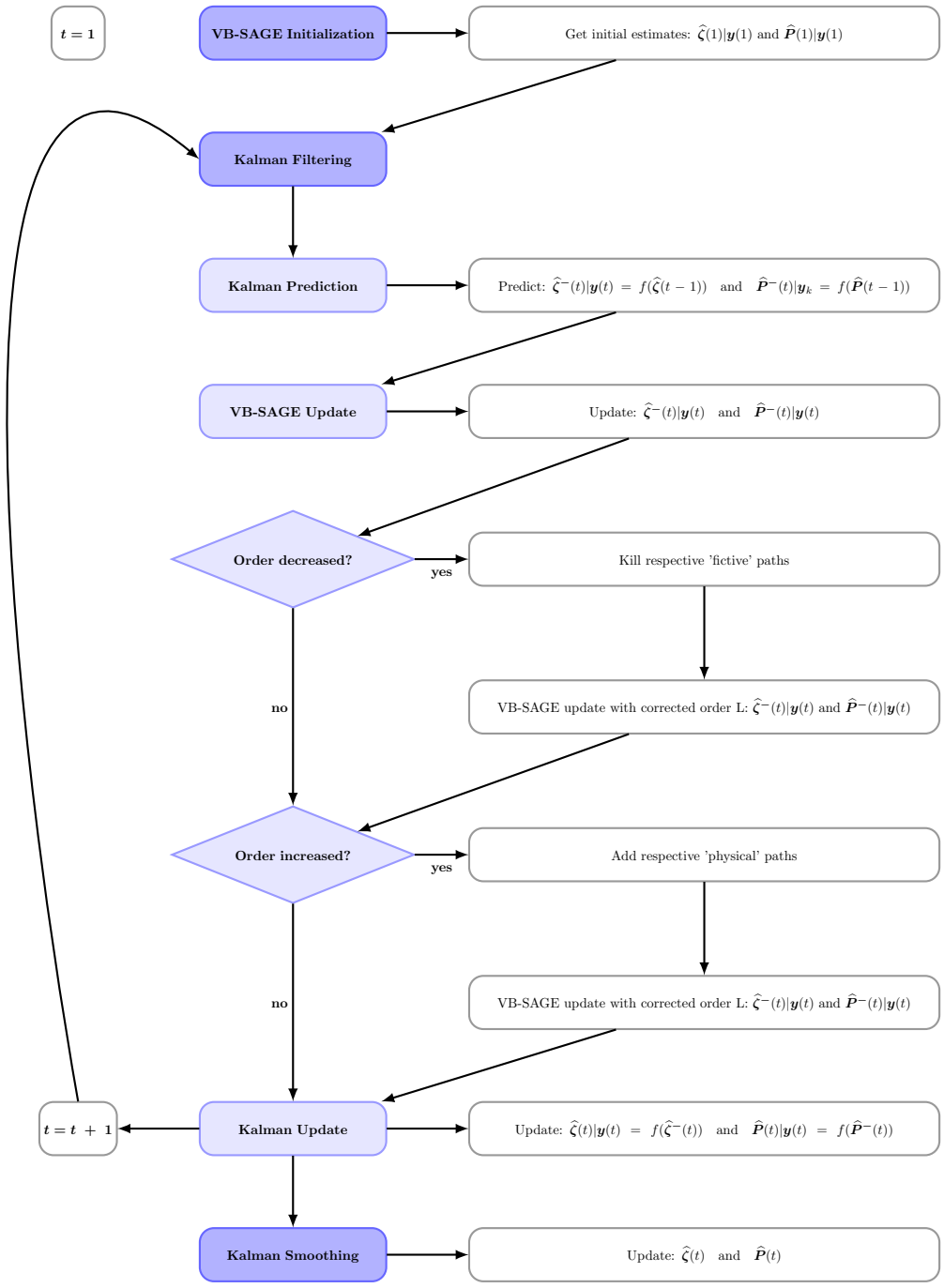


Figure 7: Graphical representation of Bayesian Estimation and Kalman Tracking/Smoothing algorithm

5 Simulation Results

A verification of BEKS should be performed by a comparison of results relating the snapshot based implementations of SAGE and VB-SAGE as well as the Kalman Enhanced Super Resolution Tracking algorithm KEST introduced by Jost, Wang, Fiebig and Pérez-Fontán (2012). KEST works in a similar way compared to BEKS as a two-stage estimator. While the inner stage makes use of the SAGE algorithm for snapshot-based estimation, the outer stage applies several Kalman Filters for tracking which are running in parallel each with a different model order. After all Kalman Filters have calculated their current estimates, a decision among all results is performed. Note that any model order change with respect to the last time step is penalized within this implementation.

To check the performance of BEKS, simulations in a geometry-based artificial scenario have been exercised; consisting of 7 static reflectors and a moving receiver. The direct Line-of-Sight (LoS) has been omitted and the single antenna transmitter is assumed to be stationary in a far field condition. The receiver is equipped by a 5-element linear antenna array and is moving along a trajectory in x-direction. A measurement has been simulated every 0.1 m on the receiver trajectory. All elements within the scenario are placed on the x/y-plane. An inter-element spacing of 0.45λ is applied to the antennas, with λ is denoting the wavelength. A snapshot at every 0.1 m on the trajectory has been generated with 101 samples per antenna, so the number of overall samples is 505. The bandwidth of the transmitted signal has been fixed to 100 MHz and the carrier frequency to 500 MHz. For simulation all paths have equal power and the Signal-to-Noise Ratio (SNR) is calculated as $\mathbb{E}_t||\boldsymbol{y}(t)||^2/(505 \cdot \sigma^2)$, where σ^2 is the variance of the white receiver noise. An overview of the scenario is shown in fig. 8. Furthermore fig. 9 visualizes the delay and azimuth angle of arrival change for each of the reflected paths in the simulated time frame of 500 snapshots.

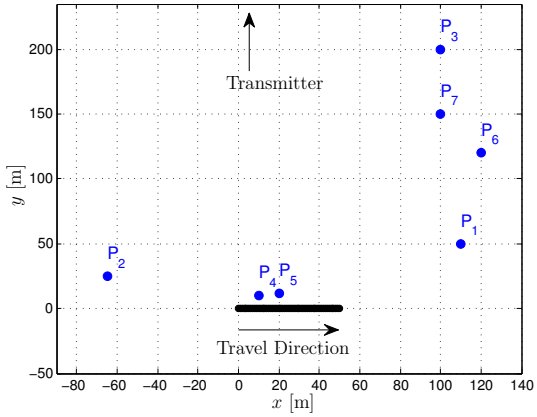
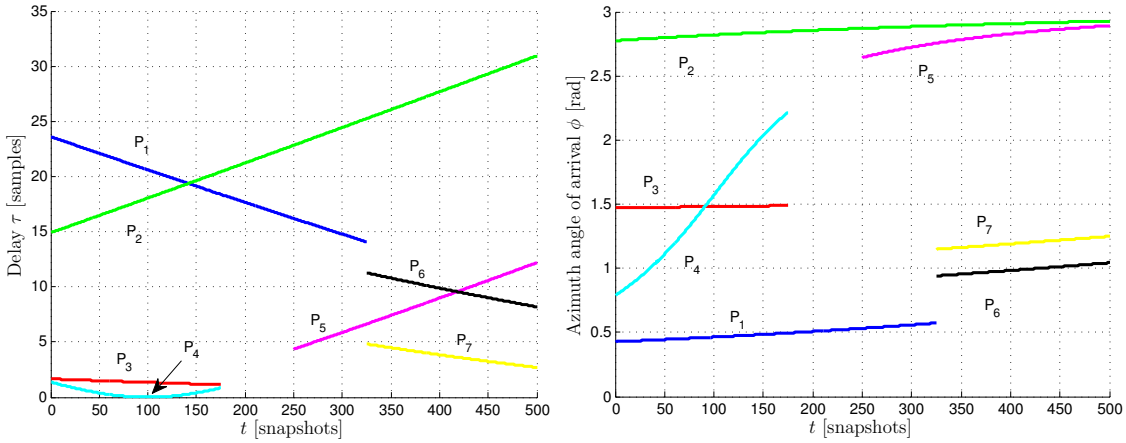


Figure 8: Scenario Visualization



(a) Szenario paths and delay

(b) Szenario paths and azimiuth angle of arrival

Figure 9: Visualization of delay and azimuth angle of arrival to the LoS path delay over time t for each path

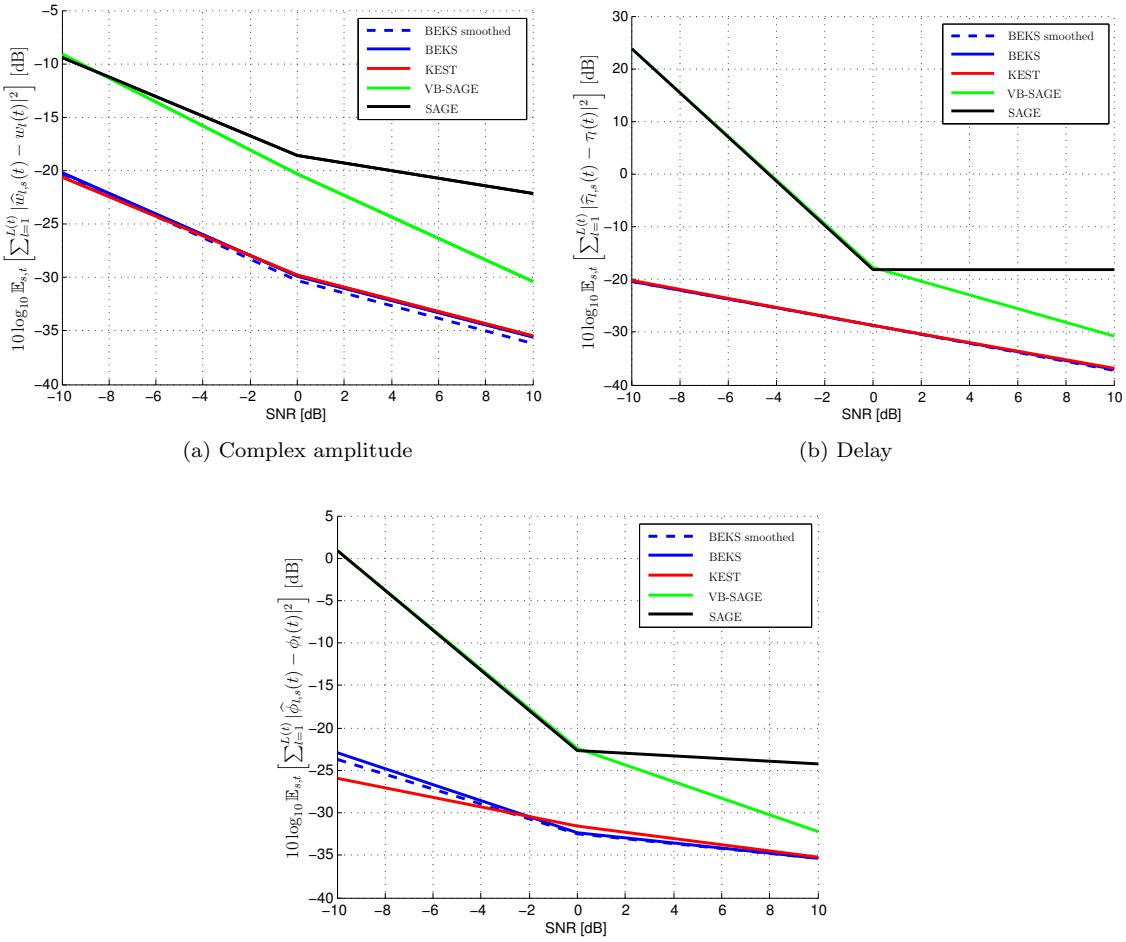
5.1 Estimation Performance

To verify the estimation performance 100 Monte Carlo runs were performed. Thereby we use the MSE

$$10\log_{10}\mathbb{E}_{s,t}\left[\sum_{l=1}^{L(t)}|\widehat{\vartheta}_{l,s}(t)-\vartheta_l(t)|^2\right]\ \text{[dB]}\quad \vartheta\in\{w,\tau,\phi\}$$

of estimated $\widehat{\vartheta}$ and true parameters ϑ over all time points t as well as all Monte Carlo runs s as performance criterion. To guarantee uniqueness of the MSE, all considerations are based on true, fixed model order L. Fig. 10 shows the performance in terms of complex amplitude $\boldsymbol{w}$, delay $\boldsymbol{\tau}$, and azimuth angle of arrival $\boldsymbol{\phi}$ estimation error. Note that in the delay case, the lines for BEKS and KEST are overlapping.

For a start a general evaluation of the tracking algorithms BEKS/KEST and the snapshot-based methods SAGE/VB-SAGE should be carried out. Obviously the tracking algorithms outperform the snapshot-based schemes in all verifications. Clearly, BEKS and KEST can track the propagation paths more accurately than the snapshot-based algorithms, since they make use of past snapshot results within the Kalman filtering process. In what follows we will have a closer look on estimation performance for all types of algorithms.



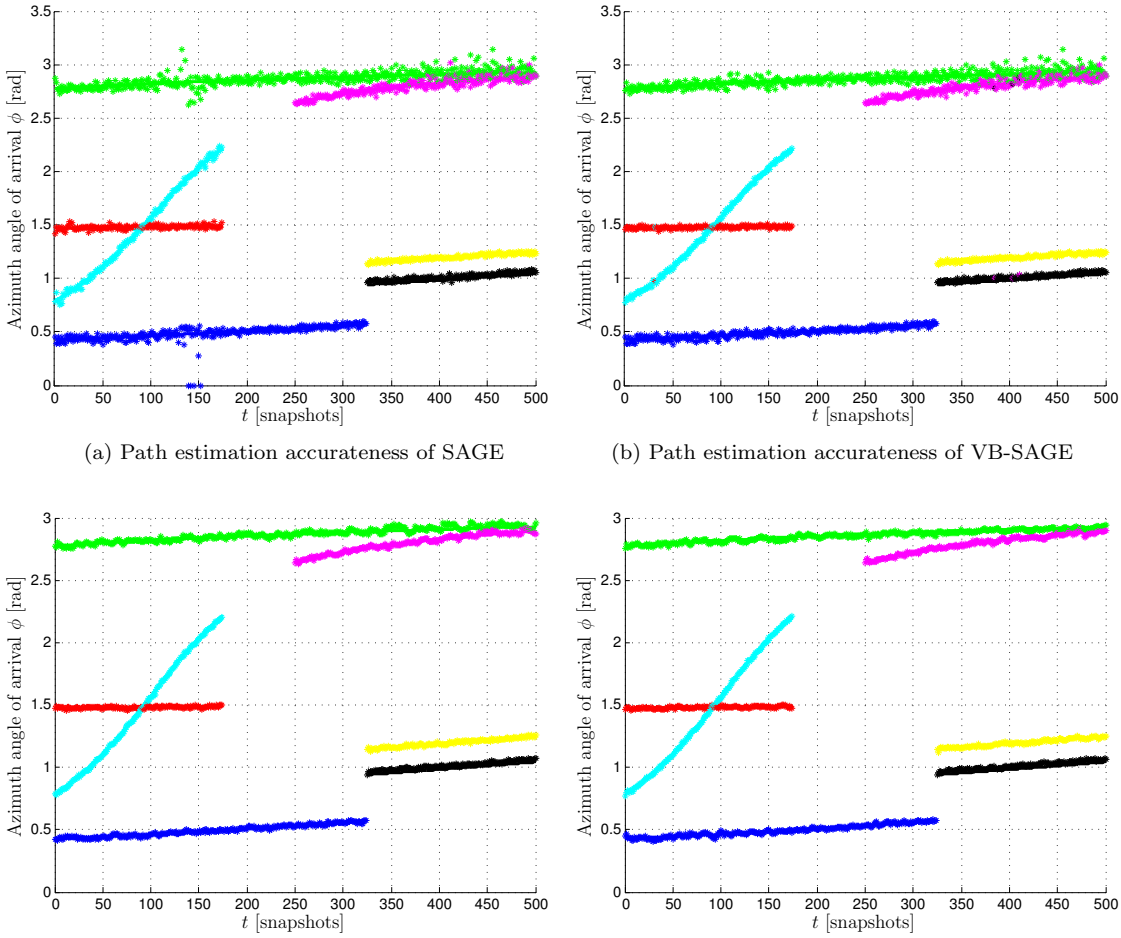
(a) Complex amplitude

(b) Delay

(c) Azimuth angle of arrival

Figure 10: Visualization of the performance in terms of delay, azimuth angle of arrival, and complex amplitude estimation error between the tracking algorithms BEKS/KEST and the snapshot-based methods SAGE/VB-SAGE

Considering the snapshot-based approaches SAGE and VB-SAGE first. The corresponding results show differences between both methods, since the latter one is more accurate in higher SNR regions. Fig. 11 visualizes as an example the estimated angle of arrivals for each multipath component at SNR of 10 dB of one Monte Carlo run. It can be seen that the estimation results for SAGE (a) are more imprecise (compared to VB-SAGE (b)) at time points 120 to 170, where the chosen scenario comprises an intersection of multipath components 1 and 2 in delay at $t = 140$. Thereby SAGE is unreliable to estimate the parameters of two propagation paths as long as they are overlapping in their delay parameter. Obviously SAGE is in contrast to VB-SAGE more prone to handle the parameter estimation right what affects as the angle of arrival estimation accurateness leading to a higher MSE.



(a) Path estimation accurateness of SAGE

(b) Path estimation accurateness of VB-SAGE

(c) Path estimation accurateness of KEST

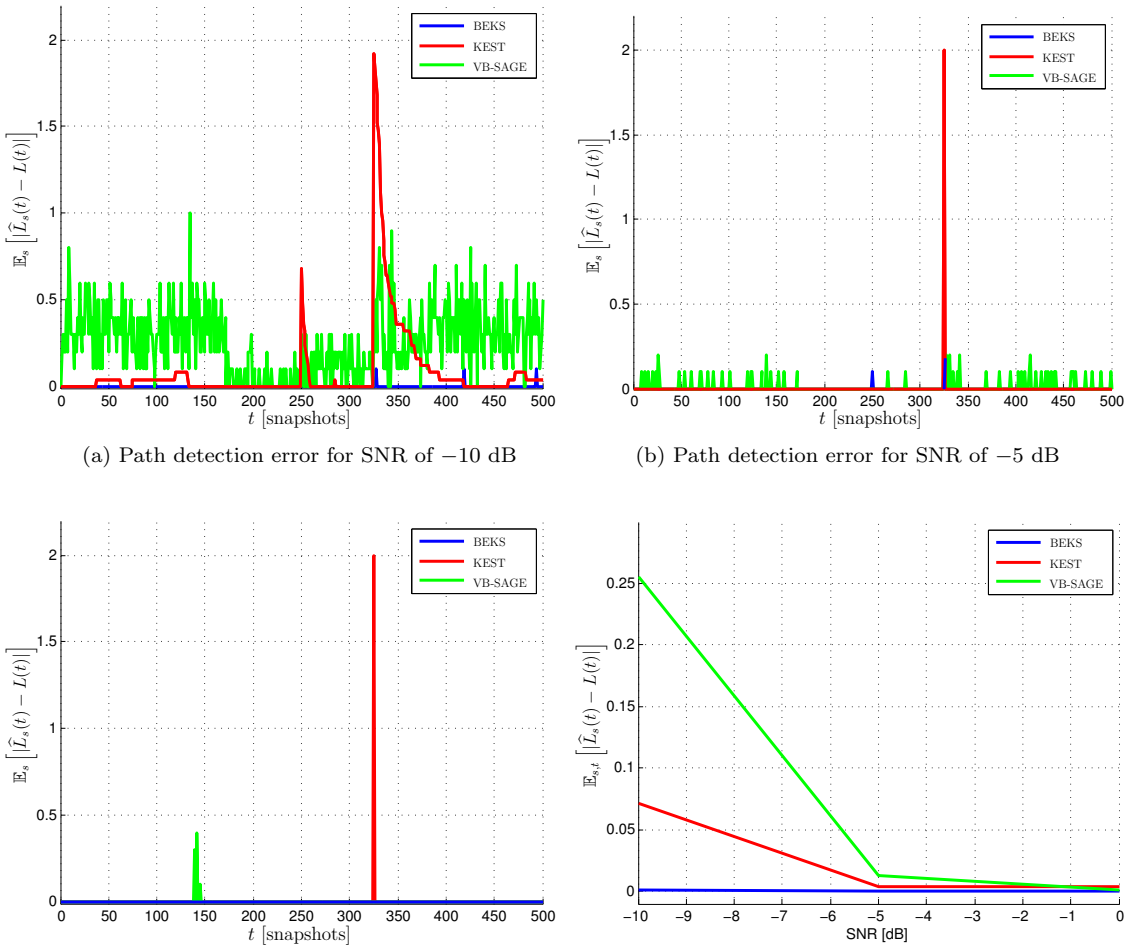
(d) Path estimation accurateness of BEKS

Figure 11: Visualization of the path estimation accurateness for azimuth angle of arrival at fixed model order at SNR of 10 dB

Next, we characterize the tracking algorithms BEKS and KEST. Both tracking algorithms show nearly equal performance in matters of estimation accuracy. First, applying the proposed BEKS algorithm with an included smoothing procedure shows a marginally increase of estimation accurateness compared to using the Kalman Filtering version only. Thus, the smoothing version performs better. Secondly, the BEKS algorithm provides similar results compared to KEST. The resulting estimation error of the complex amplitude seems to be lower by using BEKS than KEST; vice versa in the case of azimuth angle of arrival estimation error. Finally both methods can not be distinguished from one another for delay consideration.

5.2 Detection of Relevant Paths

For a comparison of BEKS with some selected algorithms in terms of propagation path detection quality 100 Monte Carlo runs s were performed for each snapshot t. Since the classical SAGE approach does not come with an integrated model order selection criteria, it is ignored in further considerations.



(a) Path detection error for SNR of −10 dB

(b) Path detection error for SNR of −5 dB

(c) Path detection error for SNR of 0 dB

(d) Overall path detection error

Figure 12: Visualization of path detection error of the snapshot-based VB-SAGE and the tracking algorithms BEKS/KEST

The mean absolute difference of number of true $L(t)$ and estimated $\widehat{L}(t)$ multipath components

$$\mathbb{E}_s\left[|\widehat{L}_s(t) - L(t)|\right]$$

for SNR's -10, -5, and 0 (in dB) is visualized in fig. 12 (a) - (c) for the snapshot-based VB-SAGE algorithm as well as the tracking algorithms BEKS and KEST,

while fig. 12 (d) shows the overall error in detected number of paths

$$\mathbb{E}_{s,t}\left[|\widehat{L}_s(t) - L(t)|\right]$$

for all SNR's.

The corresponding results show that the tracking methods BEKS and KEST clearly outperform the snapshot-based VB-SAGE algorithm, with BEKS exhibiting the best performance in lower SNR's, while the algorithms performance can not be distinguished with increasing SNR (see fig. 12 (d)).
Let's now have a closer look on path detection errors, starting with the case an SNR of -10 dB (fig. 12 (a)): It can bee seen that VB-SAGE is unreliable to estimate the number of relevant components within the whole scenario, while BEKS and KEST are only error-prone at snapshots $t = 250$ and $t = 325$ when the model order changes or when paths are close together in their azimuth angle of arrival value at $t \geq 420$. Thereby BEKS shows even better performance than KEST.
A picture similar to that is observed at an SNR of -5 dB (fig. 12 (b)). Generally, the error decreases for all types of algorithms, however both tracking algorithms only fail at corresponding model order change time points. Note that in the case of $t = 325$, KEST solely exhibits its 'natural error' since the algorithm is only capable of removing or(!) including components. This seems to be particularly problematic in situations when paths are removed and added simultaneously, since KEST will not capture the model order change in a correct way.
Finally, BEKS is without any fault at an SNR of 0 dB (fig. 12 (c)), KEST only shows the aforementioned error at $t = 325$, while the snapshot based VB-SAGE algorithm is inaccurate in the delay intersection of multipath components 1 and 2 around $t = 140$.

6 Conclusion and Outlook

In this work the two-stage Bayesian Estimation and Kalman Tracking/Smoothing (BEKS) algorithm is proposed which is able to both detect individual propagation paths and track how these paths evolve with time. It is composed of an outer stage keeping track of time-variations in terms of number of paths and paths' parameters, and an inner stage applying a snapshot-based Bayesian estimator. While the inner stage is performed by using a Variational Bayesian Space-Alternating Generalized Expectation-Maximization (VB-SAGE) algorithm, the outer stage makes use of Kalman Filtering plus Kalman Smoothing of the parameters.
Simulation based on a geometrical scenario with a moving receiver is given, showing that the algorithm outperforms the standard snapshot-based algorithms in terms of identifying the number of individual paths and estimation accuracy, since any snapshot-based method does not include prior information. Furthermore, the proposed BEKS algorithm is shown to be even superior to Kalman Enhanced Super Resolution Tracking algorithm (KEST) in cognition of individual propagation paths. In addition there are two advantages of using the novel algorithm instead of KEST: First, while KEST runs several Kalman Filters with a different model order in parallel, BEKS does only require for one, what should be computational less complex. Second, the proposed algorithm allows for adding and removing multipath components at the same snapshot, while KEST is only capable of removing or(!) including components. This seems to be particularly problematic in situations when paths are removed and added simultaneously, since the latter approach will not capture the model order change in a correct way.

Despite everything the existing status quo of BEKS is not the end of the road. The algorithm applies model order decisions by testing if one specific path amplitude is above the noise floor. This approach could be extended by using a combined version of paths' amplitudes, respecting the correlation between all existing multipath components. Furthermore one could also think about including a penalization strategy for model order changes in order to avoid paths being switched off and on again what guarantees for an improved tracking of multipaths over time.
A further aspect is to include a dense multipath component (DMC), i.e. a colored noise term which may be modelled by a multichannel auto-regressive process (Jost, Wang, Shutin and Antreich (2012)), a multichannel moving average process, or using a Kronecker Structure (Salmi et al. (2009)). The current implementation assumes a white noise process, which is insufficient compared to measurement data. The DMC count for spatial as well as temporal properties of a random scattering process in wireless channels (Salmi (2012)). Last but not least it is indispensable to check, if the algorithm is able to capture the dynamic behavior of the propagation channel by using real measurement data.

A Proofs

A.1 Transformation of the Variational Lower Bound

In this section we want to proof the equivalency of two expressions for the Variational Lower Bound, introduced in chapters 3.3 and 3.4, i.e.

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})}{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\} \stackrel{!}{=} \\
&\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{c_0 \times \exp\left\{\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}) \times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\right\}\right\}}{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} - c_1\right\},
\end{aligned} \tag{A.1}$$

where c_0, c_1 are constants.

Now, let's going on with inspecting the left hand expression of the Variational Lower Bound in eq. (A.1). With using eq. (3.9), the denominator of the Variational Lower Bound can then be written as

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})\right\} = \\
&\mathop{\mathbb{E}}_{q(\boldsymbol{\alpha}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{\Sigma}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})\right\}\right\}\right\}\right\}.
\end{aligned}$$

In the same manner we express (with $\boldsymbol{\vartheta} = \{\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}\}$)

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})\right\} = \\
&\int_{\boldsymbol{\Theta}}\int_{\boldsymbol{w}}\int_{\boldsymbol{\alpha}}\int_{\boldsymbol{\Sigma}}\Big[\log\left\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y}) \times p(\boldsymbol{w}|\boldsymbol{\alpha}) \times p(\boldsymbol{\alpha}) \times p(\boldsymbol{\Theta}) \times p(\boldsymbol{\Sigma})\right\} \times \\
&\quad q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y}) = \\
&q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})q(\boldsymbol{\Sigma}|\boldsymbol{y})\prod_{l=1}^{L} q(\alpha_l|\boldsymbol{y})q(\boldsymbol{\theta}_l|\boldsymbol{y})\Big]\partial\boldsymbol{\vartheta} = \\
&\mathop{\mathbb{E}}_{q(\boldsymbol{\alpha}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{\Sigma}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})\right\}\right\}\right\}\right\}
\end{aligned}$$

for the numerator of the left hand term of the Variational Lower Bound and combine both results in its temporary form:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})}{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\} = \\
&\mathop{\mathbb{E}}_{q(\boldsymbol{\alpha}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{\Sigma}|\boldsymbol{y})}\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{\frac{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})}{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\}\right\}\right\}\right\}. \qquad \text{(A.2)}
\end{aligned}$$

In what follows we do some mathematical transformations to rewrite the Variational Lower Bound. Therefore, we consider only the last part of eq. (A.2) for a start:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{\frac{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})}{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\} = \\
&\log\left\{\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})\right\}\right\}\right\} - \mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha}\boldsymbol{y})} \log\left\{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})\right\} = \\
&\log\left\{\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})\right\}\right\}\right\} - \log\left\{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})\right\} - k_0 = \\
&\log\left\{\frac{\exp\left\{\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})\right\}\right\}}{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\} - k_0 \qquad \text{(A.3)}
\end{aligned}$$

In the second and third line of eq. (A.3) we use eq. (3.9), i.e.

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha}\boldsymbol{y})} \log\left\{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})\right\} = \\
&\underbrace{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})\right\}}_{:=k_0,\text{ constant}} + \log\left\{q(\boldsymbol{\Theta}|\boldsymbol{y})\right\} + \log\left\{q(\boldsymbol{\alpha}|\boldsymbol{y})\right\} + \log\left\{q(\boldsymbol{\Sigma}|\boldsymbol{y})\right\} = \\
&\log\left\{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})\right\} + k_0.
\end{aligned}$$

Now we do again restrict ourselves on a sub view of the derived eq. (A.3) by considering the numerator only:

$$\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})\right\}\right\} =$$

$$\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{w},\boldsymbol{y})\times p(\boldsymbol{w}|\boldsymbol{\alpha})\times p(\boldsymbol{y})\right\}\right\} =$$

$$\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{w},\boldsymbol{y})\right\}+\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{w}|\boldsymbol{\alpha})\right\}+\log\left\{p(\boldsymbol{y})\right\}\right\}$$

Since $\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{w}|\boldsymbol{\alpha})\right\}$ and $\log\left\{p(\boldsymbol{y})\right\}$ are constant (k_1) in terms of $\boldsymbol{\Theta},\boldsymbol{\alpha}$, and $\boldsymbol{\Sigma}$ (see therefore eq. (A.2)), we proceed with using

$$p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{w},\boldsymbol{y})=\frac{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})}{p(\boldsymbol{y})}$$

to

$$\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})\right\}\right\} =$$

$$\exp\left\{k_1\right\}\times\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\right\}+\log\left\{p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\right\}-\log\left\{p(\boldsymbol{y})\right\}\right\}$$

Again, $\log\left\{p(\boldsymbol{y})\right\}$ is constant in terms of $\boldsymbol{\Theta},\boldsymbol{\alpha}$, and $\boldsymbol{\Sigma}$ and is combined with k_1 to k_2.

$$\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})\right\}\right\} =$$

$$\exp\left\{k_2\right\}\times\exp\left\{\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\right\}\right\} \tag{A.4}$$

Finally we plug eq. (A.4) into eq. (A.3)

$$\mathop{\mathbb{E}}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{\frac{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})}{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\} =$$

$$\log\left\{\frac{\exp\left\{k_2\right\}\times\exp\left\{\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})}\log\left\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma})\times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\right\}\right\}}{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\}-k_0, \tag{A.5}$$

and the resulting eq. (A.5) into eq. (A.2) in the final form of the transformed

Variational Lower Bound

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{p(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma},\boldsymbol{y})}{q(\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\} =$$

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{c_0 \times \exp\left\{\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}) \times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\right\}\right\}}{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\} - c_1, \tag{A.6}$$

with $c_1 = k_0$ and $c_0 = \exp\{k_2\}$.

A.2 Maximization of the Variational Lower Bound

Assume the transformed Variational Lower Bound

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})} \log\left\{\frac{c_0 \times \exp\left\{\mathbb{E}_{q(\boldsymbol{w}|\boldsymbol{\alpha},\boldsymbol{y})} \log\left\{p(\boldsymbol{y}|\boldsymbol{\Theta},\boldsymbol{w},\boldsymbol{\alpha},\boldsymbol{\Sigma}) \times p(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma})\right\}\right\}}{q(\boldsymbol{\Theta},\boldsymbol{\alpha},\boldsymbol{\Sigma}|\boldsymbol{y})}\right\} - c_1.$$

which needs to be maximized. For notational convenience we rewrite this expression as

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \log\left\{\frac{\widetilde{p}(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta}|\boldsymbol{y})}\right\}, \tag{A.7}$$

without respecting the constant term c_1

Case 1: Proof of eq. (3.11)

Assume $q(\boldsymbol{\vartheta}|\boldsymbol{y})$ is modelled as a dirac, thus $\delta(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}})$ or in other representation:

$$q(\boldsymbol{\vartheta}|\boldsymbol{y}) = \lim_{\sigma^2 \to 0} N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}). \tag{A.8}$$

Plugging in eq. (A.8) into eq. (A.7)

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \log\left\{\frac{\widetilde{p}(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta}|\boldsymbol{y})}\right\} =$$

$$\lim_{\sigma^2 \to 0} \int_{\boldsymbol{\vartheta}} \log \widetilde{p}(\boldsymbol{\vartheta}) \times N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}) \partial \boldsymbol{\vartheta} - \lim_{\sigma^2 \to 0} \int_{\boldsymbol{\vartheta}} \log \left\{ N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}) \right\} \times N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}) \partial \boldsymbol{\vartheta}.$$

Since

$$\int_{\boldsymbol{\vartheta}} \log \left\{ N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}) \right\} \times N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}) \partial \boldsymbol{\vartheta}$$

denoting the negative differential Entropy which can be stated in the Gaussian case as

$$\int_{\boldsymbol{\vartheta}} \log \left\{ N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}) \right\} \times N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}) \partial \boldsymbol{\vartheta} = -\frac{1}{2} \log |2\pi \exp(1) \sigma^2|,$$

we proceed with

$$\begin{aligned} &\mathop{\mathbb{E}}_{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \log \left\{ \frac{\widetilde{p}(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \right\} = \\ &\int_{\boldsymbol{\vartheta}} \log \widetilde{p}(\boldsymbol{\vartheta}) \times \lim_{\sigma^2 \to 0} N_{\sigma^2}(\boldsymbol{\vartheta} - \widehat{\boldsymbol{\vartheta}}) \partial \boldsymbol{\vartheta} + \lim_{\sigma^2 \to 0} \frac{1}{2} \log |2\pi \exp(1) \sigma^2| = \log \widetilde{p}(\widehat{\boldsymbol{\vartheta}}) - \infty = \\ &\log \widetilde{p}(\widehat{\boldsymbol{\vartheta}}) - c_2 \end{aligned} \tag{A.9}$$

Finally, the result in eq. (A.9) implies that maximizing the Variational Lower Bound is equal to maximizing $\log \widetilde{p}(\widehat{\boldsymbol{\vartheta}})$, so

$$\operatorname*{argmax}_{\boldsymbol{\vartheta}} \quad \mathop{\mathbb{E}}_{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \log \left\{ \frac{\widetilde{p}(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \right\} = \operatorname*{argmax}_{\boldsymbol{\vartheta}} \left\{ \widetilde{p}(\widehat{\boldsymbol{\vartheta}}) \right\}$$

Case 2: Proof of eq. (3.12)

Let's first reformulate the Variational Lower Bound in eq. (A.7) as a Kullback Leibler divergence, thus

$$\mathop{\mathbb{E}}_{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \log \left\{ \frac{\widetilde{p}(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \right\} = - \mathop{\mathbb{E}}_{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \log \left\{ \frac{q(\boldsymbol{\vartheta}|\boldsymbol{y})}{\widetilde{p}(\boldsymbol{\vartheta})} \right\} = -\mathrm{D}_{\mathrm{KL}}(q(\boldsymbol{\vartheta}|\boldsymbol{y}) \| \widetilde{p}(\boldsymbol{\vartheta})),$$

and maximizing the Variational Lower Bound is equal to minimizing the Kullback Leibler divergence $\mathrm{D}_{\mathrm{KL}}(q(\boldsymbol{\vartheta}|\boldsymbol{y})\|\widetilde{p}(\boldsymbol{\vartheta}))$. The Kullback-Leibler divergence is always non-negative and zero if $q(\boldsymbol{\vartheta}|\boldsymbol{y}) = \widetilde{p}(\boldsymbol{\vartheta})$.

Now assume $q(\boldsymbol{\vartheta}|\boldsymbol{y})$ and $\widetilde{p}(\boldsymbol{\vartheta})$ to be modelled as $q(\boldsymbol{\vartheta}|\boldsymbol{y}) \sim \mathcal{N}(\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$ and $\widetilde{p}(\boldsymbol{\vartheta}) \sim \mathcal{N}(\boldsymbol{\mu}_{\widetilde{p}}, \boldsymbol{\Sigma}_{\widetilde{p}})$, respectively.

Therefore, $\mathrm{D}_{\mathrm{KL}}(q(\boldsymbol{\vartheta}|\boldsymbol{y})\|\widetilde{p}(\boldsymbol{\vartheta}))$ is minimal (i.e. 0) when $q(\boldsymbol{\vartheta}|\boldsymbol{y}) = \widetilde{p}(\boldsymbol{\vartheta})$, which is if $\boldsymbol{\mu}_q = \boldsymbol{\mu}_{\widetilde{p}}$ and $\boldsymbol{\Sigma}_q = \boldsymbol{\Sigma}_{\widetilde{p}}$:

$$
\begin{aligned}
-\mathrm{D}_{\mathrm{KL}}(q(\boldsymbol{\vartheta}|\boldsymbol{y})\|\widetilde{p}(\boldsymbol{\vartheta})) = 0 \quad &\Leftrightarrow \quad \underset{\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q}{\operatorname{argmax}} \underset{q(\boldsymbol{\vartheta}|\boldsymbol{y})}{\mathbb{E}} \log \left\{ \frac{\widetilde{p}(\boldsymbol{\vartheta})}{q(\boldsymbol{\vartheta}|\boldsymbol{y})} \right\} \\
&\Leftrightarrow \quad \boldsymbol{\mu}_q = \boldsymbol{\mu}_{\widetilde{p}} \text{ and } \boldsymbol{\Sigma}_q = \boldsymbol{\Sigma}_{\widetilde{p}}
\end{aligned}
$$

B Notations and Abbreviations

Notation	Description
$\boldsymbol{a}$	Mean vector of magnitude
a_l	Mean of magnitude of the l-th component
$\boldsymbol{a}_{-l}$	Mean vector of magnitude of all but the l-th component
$\boldsymbol{\alpha}$	Vector of sparsity/precision parameters
α_l	Precision parameter of the l-th component
$\boldsymbol{\alpha}_{-l}$	Vector of precision parameters of all but the l-th component
A	Number of antennas
$\boldsymbol{A}$	Kalman transition matrix
$\boldsymbol{c}(\boldsymbol{\phi})$	Steering vector for azimuth angle of arrival $\boldsymbol{\phi}$
$\boldsymbol{H}$	Kalman measurement matrix
$\boldsymbol{I}$	Identity matrix
j	Complexe number $j = \sqrt{-1}$
$\widehat{\boldsymbol{K}}(t)$	Kalman Gain at time point t
l	Index number of the l-th component
L	Number of assumed components
λ	Wavelength
M	Frequency samples
N	Number of total samples
ω_n	Frequency axis
$\boldsymbol{p}_n$	Antenna positions on the plane
$\widehat{\boldsymbol{P}}^{-}(t)$	Covariance Kalman prediction at time point t
$\widehat{\boldsymbol{P}}(t)$	Covariance Kalman update at time point t
$\boldsymbol{\Phi}$	Covariance matrix of complex amplitude
$\boldsymbol{\phi}$	Mean vector of azimuth angle of arrival
ϕ_l	Mean of azimuth angle of arrival of the l-th component
$\boldsymbol{\phi}_{-l}$	Mean vector of azimuth angle of arrival of all but the l-th component
$\boldsymbol{Q}$	Kalman system noise matrix
$\widehat{\boldsymbol{R}}(t)$	Kalman measurement noise matrix at time point t
s	Index number for Monte Carlo run
$\boldsymbol{s}(\boldsymbol{\theta}_l)$	Nonlinear parametrisized vector by $\boldsymbol{\theta}_l$
$\boldsymbol{S}(\boldsymbol{\Theta})$	Nonlinear parametrisized matrix by $\boldsymbol{\Theta}$
$\boldsymbol{S}(\boldsymbol{\Theta}_{-l})$	Nonlinear parametrisized matrix by $\boldsymbol{\Theta}_{-l}$
σ^2	Noise variance

$\boldsymbol{\Sigma}$	Noise covariance matrix
$\boldsymbol{\Sigma}_l$	Noise covariance matrix of the l-the component
$\boldsymbol{\Sigma}_{-l}$	Noise covariance matrix of all but the l-the component
t	Snapshot time
T	Final snapshot time
$\boldsymbol{\tau}$	Mean vector of time delay
τ_l	Mean of time delay of the l-th component
$\boldsymbol{\tau}_{-l}$	Mean vector of time delay of all but the l-th component
$\boldsymbol{\theta}$	Vector of signal parameters $\{\boldsymbol{\tau}, \boldsymbol{\phi}\}$
$\boldsymbol{\theta}_l$	Vector of signal parameters $\{\boldsymbol{\tau}, \boldsymbol{\phi}\}$ of the l-the component
$\boldsymbol{\Theta}_{-l}$	Matrix of signal parameters $\{\boldsymbol{\tau}, \boldsymbol{\phi}\}$ of all but the l-the component
$\boldsymbol{u}(\boldsymbol{\tau})$	Frequency response with time delay $\boldsymbol{\tau}$
$\boldsymbol{\varphi}$	Mean vector of phase
φ_l	Mean of phase of the l-th component
$\boldsymbol{\varphi}_{-l}$	Mean vector of phase of all but the l-th component
$\boldsymbol{w}$	Mean vector of complex amplitude
w_l	Mean of amplitude of the l-th component
$\boldsymbol{w}_{-l}$	Mean vector of complex amplitude of all but the l-th component
$\boldsymbol{x}_l$	Measurement vector of the l-th component
$\widehat{\boldsymbol{x}}^{-}(t)$	State vector Kalman prediction at time point t
$\widehat{\boldsymbol{x}}(t)$	State vector Kalman update at time point t
$\boldsymbol{\xi}$	Noise zero mean circular complex normal vector
$\boldsymbol{y}$	Measurement vector of size $N \times 1$
$\boldsymbol{\zeta}$	Stacked state vector for magnitude, phase, delay, azimuth angle of arrival, and noise variance with respective differences
$\widetilde{\boldsymbol{\zeta}}$	Stacked state vector for magnitude, phase, delay, azimuth angle of arrival, and noise variance
$\widehat{\boldsymbol{\zeta}}^{-}(t)$	State vector $\boldsymbol{\zeta}$ Kalman prediction at time point t
$\widehat{\boldsymbol{\zeta}}(t)$	State vector $\boldsymbol{\zeta}$ Kalman update at time point t
$\widehat{\boldsymbol{\zeta}}_{\mathrm{VB}}(t)$	New measurements estimated by VB-SAGE at time point t

Abbreviation	Description
BEKS	Bayesian Estimation and Kalman Tracking/Smoothing
BCRLB	Bayesian Cramer-Rao Lower Bound
CRLB	Cramer-Rao Lower Bound
DMC	Dense multipath component
EM	Expectation-Maximization
GPS	Global Positioning System
KEST	Kalman Enhanced Super Resolution Tracking
KL	Kullback-Leibler (divergence)
KP	Kalman Prediction
KU	Kalman Update
LoS	Line-of-Sight
MAP	Maximum a posteriori (estimation)
ML	Maximum likelihood (estimation)
MSE	Mean Squared Error
SAGE	Space-Alternating Generalized Expectation-Maximization
SIMO	Single-Input Multiple-Output
SNR	Signal to noise ratio
SRA	Super resolution algorithm
VB-SAGE	Variational Bayesian SAGE

References

Anderson, B. D. and Moore, J. B. (2005). *Optimal Filtering*, 1 edn, Dover Publications, Inc.

Antreich, F., Nossek, J. and Utschick, W. (2008). Maximum likelihood delay estimation in a navigation receiver for aeronautical applications, *Aerospace Science and Technology* **8**: 256–267.

Beal, M. J. (2003). *Variational Algorithm for Approximate Bayesian Inference*, PhD thesis, University College London.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*, 1 edn, Springer, New York.

Bishop, C. M. and Tipping, M. E. (2000). Variational Relevance Vector Machines, *Uncertainty in Artificial Intelligence* **2**: 46–53.

Brown, R. G. and Hwang, P. Y. (1985). *Introduction to Random Signals and Applied Kalman Filtering*, 3 edn, John Wiley & Sons.

Chung, P. J. and Boehme, J. F. (2001). Recursive EM and SAGE Algorithms, *Statistical Signal Processing. Proceedings of the 11th IEEE Signal Processing Workshop* **36**(4): 540–543.

Feder, M. and Weinstein, E. (1988). Parameter Estimation of Superimposed Signals Using the EM Algorithm, *IEEE Transactions on Acoustic Speech and Signal Processing* **36**(4): 477–489.

Fessler, J. A. and Hero, A. O. (1993). Cramér-Rao Lower Bounds for Biased Image Reconstruction, *Proceedings of the 36th Midwest Symposium on Circuits and Systems* **1**: 253–256.

Fessler, J. A. and Hero, A. O. (1994). Space-Alternating Generalized Expectation-Maximization Algorithm, *IEEE Transactions on Signal Processing* **42**(10): 2664–2677.

Fleury, B. H., Tschudin, M. H., Heddergott, R., Dahlhaus, D. and Pedersen, K. I. (1999). Channel Parameter Estimation in Mobile Radio Environments Using the SAGE Algorithm, *IEEE Journal on Selected Areas in Communications* **17**(3): 434–450.

Hänsler, E. (2001). *Statistische Signale – Grundlagen und Anwendungen*, 3 edn, Springer, Berlin-Heidelberg.

Haykin, S. (2001). *Kalman Filtering and Neural Networks*, 1 edn, John Wiley & Sons.

Jost, T., Wang, W., Fiebig, U.-C. and Pérez-Fontán, F. (2012). Detection and Tracking of Mobile Propagation Channel Paths, *IEEE Transactions on Antennas and Propagation* **60**(10): 4875–4883.

Jost, T., Wang, W., Shutin, D. and Antreich, F. (2012). Using an Autoregressive Model for DMC, *EuCAP 2012* .

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems, *Transactions of the ASME–Journal of Basic Engineering* **82**: 35–45.

Kay, S. M. (1993). *Fundamentals of Statistical Signal Processing: Estimation Theory*, PTP Prentice-Hall, Inc.

Lien, Y.-N. (2011). *RF Signal Propagation*, National Chengchi University, Department of Computer Science.

Mahler, R. P. S. (2007). *Statistical Multisource-Multitarget Information Fusion*, Artech House, Inc., chapter 2.3.1, pp. 34–35.

Mitra, A. (2009). *Free Space Radio Wave Propagation*, 4 edn, Indian Institute of Technology Guwahati.

Pedersen, K. I., Fleury, B. H. and Morgensen, P. (1997). High Resolution of Electromagnetic Waves in Time-Varying Radio Channels, *Personal, Indoor and Mobile Radio Communications. Waves of the Year 2000. The 8th IEEE International Symposium* .

Peebles, P. Z. (2001). *Probability, Random Variables and Random Signal Principles*, 4 edn, McGraw-Hill International Edition, Inc.

Petersen, K. B. and Pedersen, M. S. (2012). *The Matrix Cookbook*, `http://matrixcookbook.com`.

Roweis, S. (1999). *Gaussian identities*, Department of Computer Science – New York University, p. 2.

Safaya, R. (1997). *A Multipath Channel Estimation Algorithm using a Kalman filter*, PhD thesis, Illinois Institute of Technology, Chicago.

Salmi, J. (2012). *Statistical Tools for Measurement Based Modeling and Their Application to Wireless MIMO Channel Characterization*, Aalto University School of Electrical Engineering.

Salmi, J., Richter, A. and Koivunen, V. (2009). Detection and Tracking of MIMO Propagation Path Parameters Using State-Space Approach, *IEEE Transactions on Signal Processing* **57**(2): 1538–1550.

Shutin, D., Buchgraber, T., Kulkarni, S. R. and Poor, H. V. (2011a). Fast Adaptive Variational Sparse Bayesian Learning with Automatic Relevance Determination, *IEEE Transactions on Acoustic Speech and Signal Processing* **49**(7): 2180–2183.

Shutin, D., Buchgraber, T., Kulkarni, S. R. and Poor, H. V. (2011b). Fast Variational Sparse Bayesian Learning With Automatic Relevance Determination for Superimposed Signals, *IEEE Transactions on Acoustic Speech and Signal Processing* **59**(12): 6257–6261.

Shutin, D. and Fleury, B. H. (2011). Sparse Variational Bayesian SAGE Algorithm with Application to the Estimation of Multipath Wireless Channels, *IEEE Transactions on Signal Processing* **59**(8): 3609–3623.

Shutin, D., Kubin, G. and Fleury, B. H. (1997). Application of the Evidence Procedure to the Estimation of Wireless Channels, *Hindawi Publishing Corporation, EURASIP Journal on Advances in Signal Processing* **2**: 650–654.

Shutin, D., Wang, W. and Jost, T. (2013). Incremental Sparse Bayesian Learning for Parameter Estimation of Superimposed Signals, *10th International Conference on Sampling Theory and Applications* .

Storve, S. (2012). *Kalman Smoothing Techniques in Medical Image Segmentation*, PhD thesis, Norwegian University of Science and Technology, Trondheim.

Tichavský, P., Maravchik, C. H. and Nehorai, A. (1998). Posterior Cramér-Rao Bounds for Discrete-Time Nonlinear Filtering, *IEEE Transactions on Signal Processing* **46**(5): 1386–1396.

Tipping, M. E. (2001). Sparse-Variational Bayesian Learning and the Relevance Vector Machine, *Journal of Machine Learning Research* pp. 211–244.

Tschudin, M. H. (1999). *Analyse und Vergleich von hochauflösenden Verfahren zur Funkkanalparameterschätzung*, PhD thesis, Eidgenössische Technische Hochschule Zürich.

Tschudin, M. H., Heddergott, R. and Truffer, P. (1998). Validation of a High Resolution Measurement Technique for Estimating the Parameters of Impinging Waves in Indoor Environments, *Personal, Indoor and Mobile Radio Communications. The 9th IEEE International Symposium* pp. 1411–1416.

Turin, G. L. (1972). A statistical model of urban multipath propagation, *IEEE Transactions on Vehicular Technology* **21**(1): 1–9.

van Trees, H. L. and Bell, K. E. (2007). *Bayesian bounds for Parameter Estimation and Nonlinear Filtering/Tracking*, 1 edn, John Wiley & Sons.

Weickert, J. (2010). Umkehrfunktion und Transformationsregel, *Mathematical Image Analaysis Group* .

Solemn affirmation

Herewith I affirm that I have written this thesis on my own. Cited sources of literature are perceptibly marked and listed at the end of this thesis. The work was not submitted previously in same or similar form to another examination committee and was not yet published.

Munich, January 7, 2014

Matthias Mittermayer