State-space modeling of MEG time series

by

ANTONIO MOLINS JIMÉNEZ

S.M., Massachusetts Institute of Technology (2008)
Telecommunications Engineer, Madrid Polytechnic University (2004)

Submitted to the Department of Health Sciences & Technology
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical and Medical engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2010

© 2010Antonio Molins Jiménez. All rights reserved.

The author hereby grants to MIT permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part in any medium now known or hereafter created.

ARCHIVES

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

OCT 21 2010

LIBRARIES

Author .....
Department of Health Sciences & Technology
September 9, 2010

Certified by .....
Emery N. Brown, M.D., Ph.D.
Professor of Health Sciences & Technology
and Computational Neuroscience
Thesis Supervisor

Accepted by .....
Ram Sasisekharan, Ph.D.
Edward Hood Taplin Professor of Health Sciences & Technology
and Biological Engineering
Director, Harvard-MIT Division of Health Sciences & Technology

State-space modeling of MEG time series

by

Antonio Molins Jiménez

Submitted to the Department of Health Sciences & Technology
on September 9, 2010, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical and Medical engineering

Abstract

Magnetoencephalography (MEG) non-invasively offers information about neural activity in the brain by measuring its magnetic field. Estimating the cerebral sources of neural activity from MEG is an ill-posed inverse problem that presents several challenges. First, this inverse problem is high-dimensional, as the number of possible sources exceeds the number of MEG recording sensors by at least an order of magnitude. Second, even though the neural activity has a strong temporal dynamic and the MEG recordings are made at high-temporal resolution, the temporal dynamic is usually not exploited to enhance the spatial accuracy of the source localization. Third, whereas a dynamic form of the MEG source localization problem can be easily formulated as a state-space model (SSM) problem, the high dimension of the resulting state-space makes this approach computationally impractical.

In this thesis we use a SSM to characterize from MEG recordings the spatiotemporal dynamics of underlying neural activity. We use the Kalman fixed-interval smoother (KS) to obtain maximum a posteriori (MAP) estimates of the hidden states, the expectation-maximization (EM) algorithm to obtain maximum-likelihood (ML) estimates of the parameters defining the SSM, and standard model-selection criteria to choose among competing SSMs. Because of the high dimensionality of the SSM, the computational requirements of these algorithms are high, and preclude the use of current frameworks for MEG analysis. We address these computational problems by developing an accelerated, distributed-memory version of the KS+EM algorithm appropriate for the analysis of high-dimensional data sets. Using the accelerated KS+EM algorithm, we introduce two SSM-based algorithms for MEG data analysis: KronEM (Kronecker Product modeling using KS+EM) and StimEM (Stimulus effect estimation using KS+EM).

KronEM characterizes the spatiotemporal covariance of MEG recordings using an parameterization that efficiently describes the rhythmicity present in resting state neural activity. KronEM describes the data as a sum of components composed of a time-invariant spatial signature and a temporal second-order autorregresive process. In comparison with previous attempts at modeling resting-state activity, the KronEM algorithm estimates the number of such components using the data, and is able to

identify an arbitrary number of them. We illustrate these properties on a simulation study, and then analyze MEG recordings collected from a human subject in resting state. The KronEM algorithm recovered components consistent with well-known physiological rhythmic activity. We then compare the resulting topographic maps of frequency with multi-taper based ones, and show that KronEM-based maps better localize naturally occurring rhythms. These results make the KronEM algorithm a useful single-trial frequency analysis technique.

StimEM estimates neural activity using MEG recordings made in evoked-potential studies, in which the subject is repeatedly presented with a stimulus and only the stimulus effect is of interest. In contrast with other dynamic source-localization techniques, StimEM accepts arbitrary description of neural dynamics, parameterized as a weighted sum of user-defined candidates, and finds the MAP estimate of the weights. Using the estimated dynamics, StimEM generates a time-resolved ML estimate of the evoked-potential activity in the cortex. We illustrate the ability of StimEM to identify dynamics in a simulated data set of realistic dimensions, and show that the estimates improve substantially when dynamics are taken into account. We next analyze experimental MEG data from an auditory evoked-potential study and show that StimEM identifies dynamics consistent with neurophysiology and neuroanatomy and improves the localization of the evoked cortical response.

In summary, we establish the feasibility of non-approximate SSM-based analysis of high-dimensional state-space models using a distributed-memory implementation of an accelerated KS+EM algorithm. We develop two novel algorithms to analyze MEG data in resting-state and evoked potential studies, and show that SSM analysis improves substantially on previous non-SSM based techniques.

Thesis Supervisor: Emery N. Brown, M.D., Ph.D.

Title: Professor of Health Sciences & Technology

and Computational Neuroscience

Acknowledgments

My advisor, **Emery Brown**, took me in his great lab when I most needed it, trusted me for no reason at all, offered and explained an exciting project to me, and granted me access to all the wonderful research resources without which no work would have been accomplished. Emery has been a source of insight, inspiration and support all these years.

Matti Hämäläinen introduced me to the fundamentals of MEG and minimum-norm estimation, provided me with his wonderful software tools, and was always there, even with a whole ocean between us, to solve my multiple questions with a contagious smile. Matti has always been there to help, and provided me with the *Sisu* needed to take this thesis to a satisfactory end, offering support in both high and low times.

Steve Stufflebeam was always there to solve my questions about MEG analysis, guide me through his exceptional clinical data, and connected me to the clinical realities of MEG analysis.

David Cohen took time to share with me his insight in electromagnetism in informal weekly meetings, thus giving me the opportunity to learn the fundamentals of MEG from the very inventor.

All the people at the Neural Statistics Lab were always helpful and supporting: **Iahn**, **Camilo**, **Demba**, **Francisco**, **Shinung**, **Michelle**, **Anna**, **Patrick**, **Giulia**, **Maurizio**, **David**, **Sri**, **Ricardo**, **Sage**, **Julie**... Working is much easier in a lab where a smile is the norm.

The people at the Martinos Center were always available to me. Special thanks go to **Naoro Tanaka**, who greatly helped me with my first attempts at MEG data analysis; **Daniel Wakeman**, who helped me to navigate through the MEG lab at Charlestown; and **Fah-Sua Lin**, who let me make use of his helpful Matlab© scripts.

Outside the lab, lots of beautiful people made my time here much more fun and memorable: **Gonzalito**, **Loida**, **Adam**, **Baldur**, **Andy**, **Miriam**, **Stephanie**, **Sukant**, **Juliette**, **Sachiko**, **Francia**, **Tom**, **Be**, **Eli**, **Angelo**, **Christian**, **Alicia**, **Manuel**, **Ana**, and many, many more helped me to relax and enjoy the weather when possible, and more importantly, when impossible. You guys have made of my stay in Cambridge a growing experience. I am so lucky to be surrounded by such wonderful people.

Finally, my family deserves the greatest thanks of all. They have understood my need to pursue my dreams, wherever these take me. **Papá**, **Mamá**, **Miguel**, **Luis** (you could not be included anywhere else), **Álvaro**, **Piluca**, **Javi**: I miss you every second. You always give me the energy to carry on, the hope when I most need it, the laugh that keeps me sane.

Gracias a todos.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Functional neuroimaging aims to non-invasively characterize the dynamics of the distributed neural processes that mediate brain function in both healthy and diseased subjects. An ideal functional neuroimaging method would provide estimates that are accurately localized both in space and time; in reality, a researcher will have to select from several imaging modalities providing estimates of brain function in different spatial and temporal scales. Functional MRI (fMRI) and positron emission tomography (PET) generate estimates with high spatial resolution (millimeters), but because of their dependence on neurovascular coupling (*cf.* [54]) they suffer from low temporal resolution (seconds to minutes).

Electrical activity of neurons generates electric and magnetic fields detectable on and outside the scalp ([11, 64]). With proper instrumentation, these correlates of brain activity can be recorded on or outside the head surface. Electroencephalography (EEG) and magnetoencephalography (MEG) measure these electric and magnetic fields, respectively. In contrast to other functional neuroimaging modalities, EEG and MEG signals are instantaneously related to neural electric activity. Because of this, both EEG and MEG offer estimates of brain activity with higher temporal resolution: among the available functional imaging techniques, only MEG and EEG imaging have temporal resolutions below 100 ms (see again [51] and the thorough review in [36]).

MEG and EEG imaging have been widely used in research and clinical studies

since the mid-twentieth century. Because of its high temporal resolution, MEG imaging is a powerful tool for studying neural processes in the normal working brain that are of interest for neurophysiologists, psychologists, cognitive scientists, and others interested in human brain function. The two main types of EEG and MEG studies are: (1) resting-state studies, where the subject has no experimental input, and the experimenter tries to identify the dynamics of the free-running brain under some experimental conditions; and (2) event-related studies, also known as evoked-potentials and evoked-fields, where the subject is repeatedly given a set of stimuli, and the experimenter's aim is to find the specific effects of said inputs in brain activity. Clinical applications of EEG and MEG include improved understanding and treatment of serious neurological and neuropsychological disorders. A prominent example is intractable epilepsy, where patients benefit from improved localization of epileptic foci, minimizing the extent of surgery, and non-invasive mapping of cortical areas of functional relevance, improving surgery planning and outcome ([1, 23, 58]).

EEG and MEG imaging use mathematical inversion techniques along with physical models of electric and magnetic field propagation to estimate the distribution of the underlying electrical brain activity. Unfortunately, due to the relatively large distance between the sensors and the sources and a low signal-to-noise ratio, EEG and MEG estimates of brain activity have a low spatial resolution (centimeters, see [60]). Dynamic inversion techniques modeling the underlying brain activity can increase this spatial resolution by exploiting the high temporal resolution of EEG and MEG (see [61]). The dynamic characterization of brain activity makes it possible to analyze the whole acquisition at once, producing better resolved estimates than those produced by non-dynamic techniques. Unfortunately, introducing dynamics in the inversion greatly increases the computational cost of the algorithm; and in order to reduce the cost, existing algorithms introduce non-physiological constraints in the spatiotemporal evolution of neural activity (*e.g.* [68, 28]). In this thesis we will use state-space models (SSMs) to approximate certain spatiotemporal characteristics of the underlying brain activity. Unlike previous approaches, SSMs provide with a flexible description of the dynamics that can model physiological evolution of activity.

More over, since SSMs have been used in statistical and engineering literature for a long time, the inversion can be performed using existing algorithms for estimation ([82]). However, the high-dimensional nature of the SSMs arising in EEG and MEG applications have precluded its application in the field. This thesis will provide a distributed, accelerated SSM framework that will allow us to do inference, parameter estimation and model selection in such high-dimensional state-spaces. Building on this accelerated framework, this thesis will produce SSM-based analysis tools for both resting-state and evoked potential MEG studies. In this thesis, our main emphasis is on MEG analysis but many of the methods apply directly to EEG as well.

The rest of this chapter chapter covers the physiological basis of the MEG signal, briefly reviews the instrumentation used for MEG acquisition and preprocessing, describes the physical model of magnetic field propagation, and introduces the main techniques used for source localization. Chapter 2 will introduce the SSM framework and review the theoretical aspects of dynamical estimation using SSMs. In doing so, Chapter 2 will lay out the mathematical framework that will be at the heart of the two new algorithms introduced in this thesis. KronEM (Kronecker product modeling using KS+EM), described in Chapter 3, will use the SSM framework to better identify rhythmical activity in the MEG readouts on resting-state studies. StimEM (Stimulus effect estimation using KS+EM), described in Chapter 4, will use the SSM framework to provide source activity estimates in MEG evoked potential studies. Both algorithms identify the spatiotemporal correlations existing in brain activity and use them to produce estimates that are more accurate than those provided by algorithms not modeling these dependencies. The characteristics and efficacy of our novel algorithms will be demonstrated both with simulations and analysis of experimental data.

1.1 Physiological basis of MEG

When a neuron receives input from other neurons, postsynaptic potentials (PSPs) are generated at its apical dendritic tree. In an excitatory PSP, the apical dendritic membrane depolarizes transiently, becoming extracellularly electronegative with respect to the cell soma and basal dendrites. This potential difference creates a current flow from the non-excited membrane of the soma and basal dendrites to the apical dendritic tree sustaining the EPSPs ([31]).

Some of the current, called primary current, takes the shortest route between source and the soma by traveling within the dendritic trunk. Conservation of electric charges imposes that the current loop be closed with extra-cellular currents flowing across the entire volume conductor, forming a less compact current known as the secondary (or volume) current.

The spatial arrangement of cells has a crucial role in the generation of detectable magnetic fields outside the head: the postsynaptic current directions have to be consistent across large cell populations. Thus, synchronously activated large pyramidal cortical neurons, pointing perpendicularly to the cortical surface, are believed to be the main MEG generators because of the coherent spatial distribution of their large dendritic trunks oriented in parallel ([67]). In addition, the postsynaptic currents generated in the dendritic trunks last longer than the typical action potentials of cortical neurons, making the temporal integration more effective. Furthermore, the traveling action potential can be modeled with a pair of opposing currents sources whose field is hardly detectable at a large distance.

The number of neurons needed to generate a measurable field outside the head has been estimated in several publications. In [36], the authors used an approximate model of the postsynaptic currents and concluded that approximately a million PSPs might be required to produce a typical current detected in MEG. Later, [63] employed a more accurate model of neuron physiology and were able to trim down the estimate by almost an order of magnitude. Using estimates of macrocellular current density given in [36], a patch of $5\,\mathrm{mm} \times 5\,\mathrm{mm}$ ($40\,\mathrm{mm}^2$ in the worst-case scenario) would create

an extra-cranially detectable dipole of 10 nA·m (for MEG), consistent with empirical observations and invasive studies. EEG would narrow down the needed surface area as reported in [67], but these estimates may be overly optimistic due to the limited resolution of the source estimation methods and the different sensitivity profiles of magnetic and electric sensors (see for example [15]). The detection thresholds gives the physiological resolution limit for both imaging modalities, although other factors related to the estimation process lower the resolution as will be discussed in Section 1.4. Reviews on the electrophysiological process associated with MEG signal generation can be found in [63, 15].

Finally, although MEG signals are believed to originate mainly in the cortex, some authors have reported scalp recordings of deeper cortical structures including the hippocampus, cerebellum and thalamus (see [3] for a list of these findings).

1.2 MEG instrumentation

The magnetic fields generated by neural currents are several orders of magnitude smaller than the background fields in a typical laboratory. Thus elaborate interference blocking mechanisms and extremely sensitive sensors are needed. MEG imaging was made possible by the advent of super-conducting quantum interference devices (SQUID) and shielded rooms ([13, 98]). The first SQUID-based MEG experiment with a human subject was conducted at MIT by David Cohen ([12, 14]), after the same technology was successfully applied to detect the magnetocardiogram in 1969. The first MEG experiments employed a single-sensor system, and multiple acquisitions were needed to map the external magnetic field. Current MEG systems include multiple sensors (150-300) in a helmet-shaped array ([5, 77, 86]). The SQUID technology requires cryogenic temperatures, making MEG setups more expensive and bulkier as compared with EEG instruments. MEG imaging hardware is reviewed, *e.g.*, in [36, 16].

External noise can be dampened by using magnetically shielded rooms, gradiometric coil designs, reference sensor arrays, and software noise cancellation techniques. The MEG data sets used in this thesis were acquired in a magnetically shielded room, using a partly gradiometric coil design (Vectorview, Elekta-Neuromag, Helsinki, Finland). To further reject noise high frequency and low frequency components were filtered out, and signal-space projection technique (SSP, [85]) was applied to the filtered signal. SSP eliminates from the data a noise subspace which the user defines using the data. The subspace is constructed choosing parts of the signal where the artifact is most conspicuous, computing its principal components (PCA, [70]), and selecting among them the ones that most reduce the noise by visual inspection. SSP can be used to reject two of the most usual MEG artifacts: the cardiac artifact, related to the electrical cardiac activity of the subject and correlated with the concurrently acquired electrocardiogram (ECG); and the eye-movement and blink artifacts, related to both the associated muscle activity and the movement of the electrically charged retina relative to the MEG sensors and correlated with the concurrently acquired

electrooculogram (EOG).

1.3 The forward model

In order to recover cortical brain activity from MEG sensor readouts, we need to determine the relationships between the electrical activity in the brain and the recorded extra-cranial magnetic and electric fields, governed by the Maxwell's equations. In MEG and EEG, the quasi-static approximation of the Maxwell's equations applies. This is justified by the low-frequency of the signals, the head size, and the values of the electromagnetic parameters in biological tissue ([30]). The Maxwell's equations and their quasi-static approximation are

$$\begin{cases} \nabla \cdot \vec{\mathrm{E}} = \rho/\epsilon_0 \\ \nabla \times \vec{\mathrm{E}} = -\partial\vec{\mathrm{B}}/\partial t \\ \nabla \cdot \vec{\mathrm{B}} = 0 \\ \nabla \times \vec{\mathrm{B}} = \mu_0(\vec{\mathrm{J}} + \epsilon_0 \partial\vec{\mathrm{E}}/\partial t) \end{cases} \overset{\text{quasi-static}}{\Longrightarrow} \begin{cases} \nabla \cdot \vec{\mathrm{E}} = \rho/\epsilon_0 \\ \nabla \times \vec{\mathrm{E}} = 0 \quad (\Rightarrow \vec{\mathrm{E}} = -\nabla \cdot \mathrm{V}) \\ \nabla \cdot \vec{\mathrm{B}} = 0 \\ \nabla \times \vec{\mathrm{B}} = \mu_0 \vec{\mathrm{J}} \end{cases}, \quad (1.1)$$

where $\vec{\mathrm{E}}$ and $\vec{\mathrm{B}}$ are the electric and magnetic fields, ϵ_0 and μ_0 are the electric permittivity and magnetic permeability in the vacuum, ρ is the free electric charge density in that location, $\vec{\mathrm{J}}$ is the free current density at that location, and V is the electric potential that explains the conservative $\vec{\mathrm{E}}$ field in the quasi-static approximation.

As already discussed in Section 1.1, the current source $\vec{\mathrm{J}}^P$ (primary) generates a volume current $\vec{\mathrm{J}}^{\mathrm{V}}$ so that charge is conserved within the volume conductor. The total current $\vec{\mathrm{J}}$ present in the conductor is then

$$\vec{\mathrm{J}}(\mathbf{r}) = \vec{\mathrm{J}}^P(\mathbf{r}) + \vec{\mathrm{J}}^V(\mathbf{r}) = \vec{\mathrm{J}}^P(\mathbf{r}) + \sigma(\mathbf{r}) \cdot \vec{\mathrm{E}}(\mathbf{r}) = \vec{\mathrm{J}}^P(\mathbf{r}) - \sigma(\mathbf{r}) \cdot \nabla \mathrm{V}(\mathbf{r}). \quad (1.2)$$

Since, according to the fourth Maxwell's equation in Eqs. 1.1, the divergence of the total current vanishes, we obtain a Poisson's equation governing the electric potential

$$\nabla \cdot (\sigma(\mathbf{r}) \nabla \mathrm{V}(\mathbf{r})) = \nabla \cdot \vec{\mathrm{J}}^P(\mathbf{r}), \quad (1.3)$$

which can be used to compute the potential distribution for an arbitrary conductivity distribution $\sigma(\mathbf{r})$ using numerical techniques. Once the total current is known we

can use the Ampère-Laplace law, solution of the quasi-static version of Maxwell's equations, to calculate the magnetic field

$$\vec{\mathrm{B}}(\mathbf{r}) = \frac{\mu_0}{4\pi} \int \vec{\mathrm{J}}(\mathbf{r}) \times \frac{\mathbf{r} - \mathbf{r}'}{\|\mathbf{r} - \mathbf{r}'\|^3} \partial v'. \tag{1.4}$$

This equation can be expressed as (*c.f.* [ ])

$$\vec{\mathrm{B}}(\mathbf{r}) = \frac{\mu_0}{4\pi} \int (\vec{\mathrm{J}}^P(\mathbf{r}) + \mathrm{V}(\mathbf{r}) \nabla' \sigma(\mathbf{r})) \times \frac{\mathbf{r} - \mathbf{r}'}{\|\mathbf{r} - \mathbf{r}'\|^3} \partial v', \tag{1.5}$$

where the differential operator ∇' applies to the primed coordinates.

To solve using these equations, the conductivity distribution $\sigma(\mathbf{r})$ is needed. In practice, the conductivity is often assumed to be piecewise constant. In this case, the electric potential and the magnetic field can be calculated as a solution of integral equations involving potential values on the surfaces separating compartments of different conductivities ([, ,]). The usual practice is to construct a three-compartment model of the head, dividing the cranial volume in scalp, skull and brain. Sometimes a fourth compartment is used for the cerebrospinal fluid (CSF) as well. The integral equations for $\vec{\mathrm{B}}$ and V are discretized using triangular tessellations of the interface surfaces leading to a boundary-element model (BEM) which can be solved using numerical techniques (*e.g.* [ ]). For more details on the BEM method and on forward model calculations, see [ ].

1.4 The MEG inverse problem

Whereas the forward problem of MEG generation (predicting the measurements for a given source configuration) has a unique solution, the inverse problem (estimating the source configuration for a given set of measurements) is ill-posed in the sense of Hadamard ([35]). This is both because of the non-uniqueness of the problem (first reported in [10]), and because of the small number of measurement points ($\sim 10^2$) compared to the number of source locations ($\sim 10^4$). Additionally, small errors in the sensor readouts can produce big changes in the estimates. Because of the ill-posed nature of the inverse problem, source localization algorithms ought to impose additional constraints in the solution to make it unique.

The source localization algorithms proposed in the literature can be classified into two main families: parametric inverse algorithms, where the current source configuration is described by a small number of parameters (current-dipole models), and distributed-source inverse algorithms, where the current source configuration is discretized to a large number of source locations with constrained or unconstrained orientations. In this thesis, we will study distributed-source algorithms that are cortically constrained, *i.e.*, the current sources are restricted to the cortex, which is previously delineated by segmentation of the patient's anatomical MRI.

Distributed-source algorithms can be further classified as static or dynamic, depending on whether temporal correlations in brain activity are included in the formulation of the solver or not. Static distributed-source localization algorithms assume the measurements and brain activity to be temporally white, so that source activity estimates depend on the current measurement only. Thus the additional constraints making the solution unique are only imposed on the spatial configuration of the recovered brain activity. Different spatial constraints have been proposed: minimum energy ([37, 88, 89, 90]), maximum smoothness ([69]), minimum current ([86]), and focality ([32]). The spatial resolution achieved by any given static method depends on the algorithm itself as well as on the subject's anatomy, geometry of the sensor array, and instrumental noise levels. Moreover, this spatial resolution is not uniform

across all source locations. Quantification of the spatial resolution for the general case is thus difficult, but [69] estimated a sub-centimeter spatial resolution for most brain locations using both theoretical considerations and experimental data analysis for the case of the static MNE and dSPM algorithms.

In contrast to static algorithms, dynamic algorithms impose constraints on the time evolution of source activity, thus modeling spatiotemporal correlations across sources. Several studies have shown that by analyzing the complete time series, dynamic algorithms can achieve higher spatial resolution on EEG and MEG imaging ([3, 76, 28, 68]). Simple spatiotemporal dynamics imposing correlations across nearest neighbors improved the quality of the estimates ([28, 30]) and made the inverse problem more observable ([74]). Moreover, both neurophysiology and computational models of the functioning brain strongly suggest that such spatiotemporal correlations must exist. Cortical networks display oscillations that require spatiotemporal interactions (*c.f.* ([10])); intracranial recordings report strong spatial correlations at distances up to 10 mm ([1, 24, 55]); and pyramidal cells spreading laterally at distances up to 6 mm provide an anatomical substrate for such local interactions ([36]). Perhaps more importantly, other functional neuroimaging modalities such PET and fMRI have provided evidence of long-range spatiotemporal correlations in brain activity during resting-state and experimentally administered task periods ([73, 24, 20]).

Existing algorithms favor computationally agile solutions that impose additional spatiotemporal constraints on the neural dynamics in order to efficiently analyze the data. These constraints can be classified in three different strategies: imposing separable spatiotemporal dynamics ([33, 68]), expressing the activity in terms of basis functions ([17]), or providing approximated solutions to the full spatiotemporal model that decouple space and time ([28, 81, 41]). The resulting spatiotemporal dynamics are not based on neurophysiology and can not accommodate for different dynamics than the ones used in the formulation. It would be desirable to develop a framework that would let us include arbitrary spatiotemporal dynamics and, since the real spatiotemporal dynamics are being investigated, and might be subject-dependent, an algorithm that would help us choose among different candidate dynamics based on

the observed data.

1.5 Contributions of this thesis

1.5.1 KS+EM implementation

In this thesis we use a SSM framework to model the dynamics of the MEG time series and its underlying brain activity. This allows us to use the inference, parameter estimation, and model selection tools previously described in the SSM literature. These tools will be described and extended in Chapter 2 of this thesis. Even though previous works, most notably [45, 27, 91], used a SSM framework to model MEG time series, they did so by means of approximations to produce a computationally tractable model. These approximations ignore cross-terms in the spatiotemporal covariance rendering it separable, and reduce the full analysis to a collection of one-dimensional problems. However, the resulting algorithm will not yield maximum likelihood (ML) estimates, and its stability is not guaranteed. The authors justify these drawbacks on the high computational cost of the solution when the full dynamical inversion problem is addressed.

In this thesis we will use a fixed-interval Kalman smoother (KS) to obtain MAP estimates of the activity (see [2] for smoother derivation, and [16] for original work), and the expectation-maximization (EM) algorithm to obtain ML estimates of the SSM parameters, as was first suggested by [52]. EM is guaranteed to converge to a (potentially local) maximum of the full-likelihood, and the smoothed estimates are statistically optimal for the parameters EM estimates from the data. Conveniently, the KS offers as an intermediate result the likelihood of the data given the parameters, so that model selection criteria can be easily computed from the KS+EM algorithm results, making model-selection straightforward. We will avoid approximations, and will obtain statistically optimal estimates of both model parameters and activity estimates.

Unfortunately, the KS computational cost, analyzed in Chapter 2, grows cubically with the state-space dimension, which can be on the order of thousand of sources for the typical MEG inverse problem in distributed solutions. This has been so far the main reason for using approximations or alternative algorithms in MEG data analysis.

To overcome this difficulty, we develop a distributed-memory implementation of the KS that let us spread the computational load and the storage requirements across a set of computers, thus making the KS of high-dimensional state-spaces possible. On top of this, we will explain in Chapter 2 how the steady-state properties of the filter and smoother covariances can be used to reduce both the number of required computations and the memory footprint of our algorithm. This allow us to apply high-dimensional SSMs on data sets with a large number of observations. To further reduce the computational load of the KS+EM approach, we will show how a technique proposed by [15] to speed up the EM algorithm convergency is especially well suited for our case, where the E-step is computed via the KS.

Because the problem of estimating the dynamic model from the data introduces several more unknowns, and this would have a negative effect on the certainty of our estimates, it is of utmost importance to reduce the number of free parameters needed to describe our model, also known as model order. In the two algorithms that will be introduced in the thesis, we strive to achieve a compact representation of the observed data, and use model-selection criteria to reduce the model order to a minimum.

1.5.2 KronEM

KronEM, the first algorithm introduced in this thesis for MEG data analysis, will address the characterization of the process noise in MEG. MEG recordings have been shown to be heteroskedastic (*i.e.* not well described by i.i.d. innovations) in previous works such as [18], making a strong case for a hidden-state (*i.e.* SSM) representation. MEG sensor activity registered with the subject present with no task, known as *background* noise, has been shown to be temporally correlated to some extent, as in [7, 6], where the spatiotemporal covariance matrix of such MEG data was modeled using separate spatial and temporal components. The covariance that was modeled in these works does not translate directly to our SSM framework, where the observed *background* noise would be a function of both the process and measurement noise, the former related to the dynamics and the later to the acquisition device. In developing KronEM, we will show how a simple SSM structure can generalize the spatiotempo-

ral structure used in previous works ([,]) that could explain up to $\sim$ 80% of the noise. KronEM efficiently parameterizes the SSM as a combination of spatial signatures modulating second-order autorregresive processes, each presenting a different resonant frequency. In comparison with previous attempts at modeling resting-state activity, KronEM identifies the required number of resonant frequencies needed to explain the data, and can estimate an arbitrary number of said components, making it a useful addition to traditional single-trial frequency analysis techniques such as multi-taper frequency analysis. In Chapter 3, we set the mathematical foundations of the algorithm building on the KS+EM work of Chapter 2 and demonstrate how KronEM can correctly identify the number of frequency components in a simulated dataset using data alone. We then analyze experimental MEG data from a resting-state study, where KroneEM identifies individual frequency components at frequencies consistent with well-known physiological rhythmic activity. We then compare KronEM estimates with multi-taper based topographic maps of frequency, finding KronEM estimates more spatially localized than those provided by multi-taper spectral estimates generated from the same data.

1.5.3 StimEM

StimEM offers a new approach to evoked response estimation that includes the system dynamics in the inference process, thus better accounting for transient system states that could impair the final estimates if plain event-locked averaging was to be used. To accomplish this, the KS+EM algorithm is modified to better process evoked response studies, where an external stimulus is presented to the subject under study and the desired output of the algorithm is the effect of such stimulus. The usual way of dealing with this kind of studies is event-locked averaging, which assumes that the brain response to the stimulus does not vary across the study except for additive noise. However, both the response and the background noise vary across trials; therefore, the estimates can be expected to become more reliable and accurate when these effects are accounted for. In [ ], a per-trial scaling factor is introduced to deal with habituation and spike inversion effects. In [ ], a more complex factor analysis is introduced that

accounts for variations in background noise. Because no hidden process is employed in these studies, the background activity of the brain is included in the definition of the noise. Our SSM framework does not account naturally for non-linear scaling factors, but the effects of the background activity of the brain can be modeled by using the estimated brain dynamics and an extra stimulus-locked stimulus effect. StimEM includes this stimulus effect term in its generative model, characterizing such effect both temporally and spatially; it also provides with ways of doing ML estimation of the stimulus effect at the same time than the intrinsic brain dynamics.

Chapter 2

The KS+EM algorithm

In this chapter we will introduce the state-space model (SSM) framework that will be used in all following chapters. Because a high computational cost has precluded previous attempts at SSM analysis of MEG data, we review the algorithms used for its inference focusing on the causes for its elevated computational cost. We then propose a set of modifications that reduce said computational complexity, and produce as a result an accelerated SSM framework that will be the foundation of the two algorithms proposed in the following chapters, KronEM and StimEM, making it possible for them to apply the SSM inference algorithms to the high-dimensional state-spaces required for MEG time-series analysis.

A SSM characterizes a set of observations by introducing a collection of hidden states that follow a dynamic defined by the SSM parameters. In this thesis our observations will be the EEG or MEG time series, and the hidden states will depend on the estimation problem at hand and will vary in different chapters. Also, for a given definition of the hidden states, say brain activity, we can have more than one SSM (or model) to choose from. This chapter provides with the tools to tackle the inverse problem of estimation in this scenario, which can be broken down in three pieces: (1) estimating the hidden states, (2) estimating the model parameters, and (3) choosing between different possible models.

We formally define a SSM in Section 2.1, discuss some of its characteristics and briefly introduce different hidden-state choices that will be used in later chapters.

The EM algorithm is introduced in Section 2.2. The EM algorithm deals with parts (1) and (2) of the complete estimation problem; it provides with both estimates for the SSM parameters and the hidden states. EM is an iterative algorithm that brokes the estimation problem in two steps: the E-step estimates the hidden states for the given estimate of the SSM parameters, while the M-step estimates the parameters of the SSM for the hidden state estimates computed in the E-step. This two steps iterate sequentially in a set of initial estimates until a given convergency criterion is met. The E-step and M-step are covered in Sections 2.3 and 2.4 respectively.

The E-step is the most computationally intensive part of the EM algorithm. Due to the SSM structure, the E-step can be accomplished quite efficiently using the Kalman smoother (KS). As this is the most computationally intensive part of the algorithm, we will refer to the method as KS+EM. Due to the dimensionality of the state-spaces we will be analyzing in the thesis, the application of the KS+EM algorithm is challenging, because of its high computational cost and the amount of memory it requires. To alleviate this, the algorithm was implemented in a distributed memory system, the KS was modified to reduce the number of computations, and the EM algorithm was modified in order to reduce the number of required iterations. All these are described in Section 2.5.

To solve the remaining part (3) of the estimation problem, we will need to choose among different competing SSMs to explain our data. This can be done using information criteria that will be described in Section 2.6. This criteria rely on approximations will allow us to choose the least complex model that describe our data well, a characteristic known as parsimony.

2.1 State-space model

State-space model formulations are a data-augmentation strategy to model the dynamics of a time series $\mathbf{y}(t)$ by means of a hidden, unobservable process $\mathbf{x}(t)$, which is instantaneously related to the original time-series under consideration ([17], 9.1). In this thesis we consider discrete-time invariant SSMs, where the dynamics of the hidden process are described by the state transition equation

$$\mathbf{x}(t) = \mathrm{A}\mathbf{x}(t-1) + \mathrm{B}\mathbf{u}(t) + \mathbf{v}(t), \qquad t = 1, \ldots, n, \tag{2.1}$$

where the state-transition matrix A defines the spatiotemporal dynamics, the matrix B defines the effect on the hidden process of the inputs to the system $\mathbf{u}(t)$, and all other variations of the hidden process are described by the temporally white and normally distributed innovations $\mathbf{v}(t)$, also known as process noise. The instantaneous relationship of $\mathbf{x}(t)$ and $\mathbf{y}(t)$ is described by the observation or measurement equation

$$\mathbf{y}(t) = \mathrm{C}\mathbf{x}(t) + \mathbf{w}(t), \qquad t = 1, \ldots, n, \tag{2.2}$$

where the observation matrix C defines the observation model, and the temporally white and normally distributed $\mathbf{w}(t)$, known as measurement noise, captures all sources of variation independent of $\mathbf{x}(t)$.We will assume that the observation noise $\mathbf{w}(t)$ and the process noise $\mathbf{v}(t)$ are independent and zero-mean, but should this condition not hold all what follows could be modified to accommodate for such correlations. The correlation structure that defines the SSM can be compactly described by

$$E\left(\begin{bmatrix} \mathbf{x}(0) \\ \mathbf{v}(m) \\ \mathbf{w}(n) \end{bmatrix}\begin{bmatrix} \mathbf{x}(0) \\ \mathbf{v}(m) \\ \mathbf{w}(n) \end{bmatrix}^{\mathsf{T}}\right) = \begin{bmatrix} \Lambda_0 & 0 & 0 \\ 0 & \delta(m-n)\mathrm{Q} & 0 \\ 0 & 0 & \delta(m-n)\mathrm{R} \end{bmatrix}. \tag{2.3}$$

In this thesis we will apply the state-space formulation to two different scenarios, and the corresponding models will be fully described in the following chapters. In

order to motivate the reader, we will briefly explain the two scenarios here:

- **Spatiotemporal characterization of baseline activity**: In this scenario, the measurements y are obtained when no stimulus is present, so that $\mathbf{u}(t) \equiv \mathbf{0}$. Since rhythmic activity is of physiological importance, and it is known to describe the majority of temporal correlations in resting-state activity, and since autorregresive process (AR) describe rhythmic activity efficiently, we will model the data as a linear combination of several AR processes. The hidden process of the SSM, $\mathbf{x}$, will consist of the present and past values of such AR process, the transition matrix A will define the coefficients of the AR processes, the process noise covariance matrix $\mathbf{Q}$ will define the power of each AR process, the observation matrix C will define the spatial signature of each process, and the measurement noise covariance matrix R represents the amount of variance in the measurements not captured by the model.

- **Evoked potential studies**: In this scenario, we obtain the measurements $\mathbf{y}$ resulting from applying to the subject a stimulus $\mathbf{u} = \{\mathbf{u}(t), t = 1, \ldots, n\}$ that is known, and we want to estimate the input effect matrix B. In this case, $\mathbf{x}(t)$ represents the brain activity at time point t, A describes the spatial spreading of activity in the brain from one time-point to the next, Q gives an idea of the increase in brain activity among two consecutive samples, and B describes the effect of a given input in any of the source locations where we are estimating brain activity. The observation model of Eq.2.2 is known in advance, with C given by the BEM forward model, and R estimated from empty-room measurements. We will have to estimate both A, B, and Q. We will also introduce some physiologically-inspired constraints in A to reduce the dimensionality of the model to make estimation possible given the size of the data set.

State-space models are well-studied in the literature, and offer very efficient algorithms for inference and estimation. We briefly review now some of the characteristics that justify their popularity in modeling.

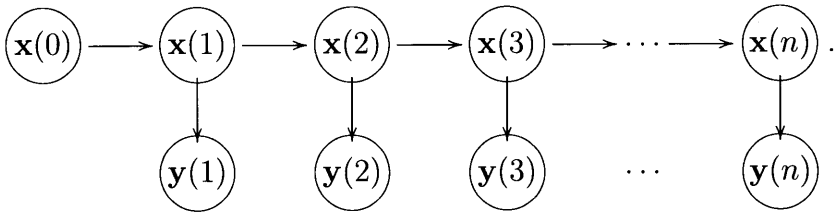
Because of the structure of the model, the hidden states $\mathbf{x}(t)$ are a Markov process,

$$p(\mathbf{x}(t)|\mathbf{x}(t-1), \mathbf{x}(t-2), \ldots, \mathbf{x}(0)) = p(\mathbf{x}(t)|\mathbf{x}(t-1)), \tag{2.4}$$

and the observations $\mathbf{y}_i$ only depend on the hidden state at that time point, so that the full-data probability can be expressed as

$$p(\mathbf{x}(0), \mathbf{x}(1), \ldots, \mathbf{x}(n), \mathbf{y}(1), \ldots, \mathbf{y}(n)) = p(\mathbf{x}(0)) \prod_{t=1}^{n} \left(p(\mathbf{x}(t)|\mathbf{x}(t-1))p(\mathbf{y}(t)|\mathbf{x}(t))\right). \tag{2.5}$$

This factor graph structure corresponds to the graphical model



(2.6)

only components of the graphical model that are joined by a line have non-zero elements of the precision matrix that are non-zero.

It is because of this sparse precision matrix that we can do estimation in the SSM very efficiently. The algorithm first described by [18] exploits this covariance structure to estimate the values of the hidden parameters $\mathbf{x} = \{\mathbf{x}(t), t = 0, \ldots, n\}$ from the observed measurements $\mathbf{y} = \{\mathbf{y}(t), t = 1, \ldots, n\}$: it works its way from the first hidden state value $\mathbf{x}(0)$ to the last value $\mathbf{x}(n)$ and then goes back to propagate the information.

2.2 The Expectation-Maximization algorithm

The expectation-maximization algorithm (see [ ]) provides with estimates of the SSM parameters θ using the measured data $\mathbf{y}$, without knowing the values of the hidden parameters $\mathbf{x}$. It does so by maximizing the likelihood of θ,

$$ll(\theta) = log\left(p(\mathbf{y}|\theta\right), \tag{2.7}$$

in a recursive way. Starting from an initial guess θ_0 we proceed iteratively: In iteration $n+1$ we have the current estimate θ_n and want a new estimate θ such that

$$ll(\theta) > L\left(\theta_n\right) . \tag{2.8}$$

We can lower-bound the increase in likelihood obtained replacing θ_n by θ is

$$\begin{aligned}
&ll(\theta) - ll(\theta_n) = \\
&= log\left(p(\mathbf{y}|\theta)\right) - log\left(p(\mathbf{y}|\theta_n)\right) \\
&= log\left(\int p(\mathbf{y}|\mathbf{x},\theta)p(\mathbf{x}|\theta)\partial\mathbf{x}\right) - log\left(p(\mathbf{y}|\theta_n)\right) && (2.9) \\
&= log\left(\int p(\mathbf{x}|\mathbf{y},\theta_n)\frac{p(\mathbf{y}|\mathbf{x},\theta)p(\mathbf{x}|\theta)}{p(\mathbf{x}|\mathbf{y},\theta_n)}\partial\mathbf{x}\right) - log\left(p(\mathbf{y}|\theta_n)\right) \\
&\geq \int p(\mathbf{x}|\mathbf{y},\theta_n) log\left(\frac{p(\mathbf{y}|\mathbf{x},\theta)p(\mathbf{x}|\theta)}{p(\mathbf{x}|\mathbf{y},\theta_n)}\right)\partial\mathbf{x} - log\left(p(\mathbf{y}|\theta_n)\right) && (2.10) \\
&= \int p(\mathbf{x}|\mathbf{y},\theta_n) log\left(\frac{p(\mathbf{y}|\mathbf{x},\theta)p(\mathbf{x}|\theta)}{p(\mathbf{x}|\mathbf{y},\theta_n)p(\mathbf{y}|\theta_n)}\right)\partial\mathbf{x} \\
&\triangleq \Delta\left(\theta|\theta_n\right) , && (2.11)
\end{aligned}$$

where in 2.9 we use the law of total probability, and in 2.10 we use the Jensen inequality.

We can prove that the bound is tight in

$$\begin{aligned}
&\Delta\left(\theta_n|\theta_n\right) = \\
&= \int p(\mathbf{x}|\mathbf{y},\theta_n) log\left(\frac{p(\mathbf{y}|\mathbf{x},\theta_n)p(\mathbf{x}|\theta_n)}{p(\mathbf{x}|\mathbf{y},\theta_n)p(\mathbf{y}|\theta_n)}\right)\partial\mathbf{x} \\
&= \int p(\mathbf{x}|\mathbf{y},\theta_n) log\left(\frac{p(\mathbf{y},\mathbf{x}|\theta_n)}{p(\mathbf{x},\mathbf{y}|\theta_n)}\right)\partial\mathbf{x} \\
&= 0\,. \qquad (2.12)
\end{aligned}$$

If we now choose θ_{n+1} so that $\Delta\left(\theta_{n+1}|\theta_n\right)$ is maximized, Eqs. 2.11 and 2.12 imply that we will either move to a new estimate producing higher likelihood or we will stay in the current estimate if it locally maximizes the likelihood. This updating strategy is very clearly explained in a graphical manner in [ ],and lies at the core of the EM algorithm. Formally,

$$\begin{aligned}
\theta_{n+1} &= \arg\max_{\theta}\left\{\Delta\left(\theta|\theta_n\right)\right\} \\
&= \arg\max_{\theta}\left\{\int p(\mathbf{x}|\mathbf{y},\theta_n) log\left(\frac{p(\mathbf{y}|\mathbf{x},\theta)p(\mathbf{x}|\theta)}{p(\mathbf{x}|\mathbf{y},\theta_n)p(\mathbf{y}|\theta_n)}\right)\partial\mathbf{x})\right\} \\
&= \arg\max_{\theta}\left\{\int p(\mathbf{x}|\mathbf{y},\theta_n) log\left(p(\mathbf{y}|\mathbf{x},\theta)p(\mathbf{x}|\theta)\right)\partial\mathbf{x}\right\} \\
&= \arg\max_{\theta}\left\{\int p(\mathbf{x}|\mathbf{y},\theta_n) log\left(p(\mathbf{y},\mathbf{x}|\theta)\right)\partial\mathbf{x}\right\} \\
&= \arg\max_{\theta}\left\{E_{\mathbf{x}|\mathbf{y},\theta_n}\{log(p(\mathbf{y},\mathbf{x}|\theta\}\right\}\,, \qquad (2.13)
\end{aligned}$$

where we define the expected log-likelihood as

$$G(\theta) \triangleq E_{\mathbf{x}|\mathbf{y},\theta_n}\{log(p(\mathbf{y},\mathbf{x}|\theta\}. \qquad (2.14)$$

Equation 2.13 shows the recursive nature of the EM algorithm. Starting in a initial candidate θ_0, we iterate on the

- E step: Determine the form of the expected log-likelihood $G(\theta)$ for the current estimates.
- M step: Maximize $G(\theta)$ with respect to θ to obtain the new estimate.

2.3 E step

In this section we will describe the E step for our SSM. In Section 2.3.1 we obtain an expression for the expected log-likelihood of the SSM given our parameters, that is $G(\theta)$. The expression obtained depends on the smoothed estimates for the hidden states. This estimates are efficiently computed using the Kalman smoother (KS), which is described in Section 2.3.2. Because this is by far the most computationally demanding part of the KS+EM algorithm, we provide some observations about the structure of the KS in Section 2.3.5. Finally, in Section 2.3.6 we rewrite $G(\theta)$ using sufficient statistics computed from the KS results.

2.3.1 Expected log-likelihood for the SSM

For the SSM in Eq. 2.1-2.2, the full likelihood of the observed data $\mathbf{y}$ and hidden states $\mathbf{x}$ given the parameters $\theta = \{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{Q}, \mathrm{R}\}$ (Eq. 2.7) can be written using the innovations form as

$$\begin{aligned} ll(\theta) &= -\frac{1}{2}log|\Lambda_0| - \frac{1}{2}(\mathbf{x}_0 - \mu_0)^\mathsf{T}\Lambda_0^{-1}(\mathbf{x}_0 - \mu_0) \\ &\quad -\frac{n}{2}log|\mathrm{Q}| - \frac{1}{2}\sum_{t=1}^{n}(\mathbf{x}_t - \mathrm{A}\mathbf{x}_{t-1} - \mathrm{B}\mathbf{u}_t)^\mathsf{T}\mathrm{Q}^{-1}(\mathbf{x}_t - \mathrm{A}\mathbf{x}_{t-1} - \mathrm{B}\mathbf{u}_t) \\ &\quad -\frac{n}{2}log|\mathrm{R}| - \frac{1}{2}\sum_{t=1}^{n}(\mathbf{y}_t - \mathrm{C}\mathbf{x}_t)^\mathsf{T}\mathrm{R}^{-1}(\mathbf{y}_t - \mathrm{C}\mathbf{y}_t)\,. \end{aligned} \tag{2.15}$$

Then, using $\mathbf{a}\mathbf{b}^\mathsf{T} = tr(\mathbf{b}^\mathsf{T}\mathbf{a})$,

$$\begin{aligned} ll(\theta) &= -\frac{1}{2}log|\Lambda_0| - \frac{1}{2}tr(\Lambda_0^{-1}(\mathbf{x}_0 - \mu_0)(\mathbf{x}_0 - \mu_0)^\mathsf{T}) \\ &\quad -\frac{n}{2}log|\mathrm{Q}| - \frac{1}{2}tr(\sum_{t=1}^{n}\mathrm{Q}^{-1}(\mathbf{x}_t - \mathrm{A}\mathbf{x}_{t-1} - \mathrm{B}\mathbf{u}_t)(\mathbf{x}_t - \mathrm{A}\mathbf{x}_{t-1} - \mathrm{B}\mathbf{u}_t)^\mathsf{T}) \\ &\quad -\frac{n}{2}log|\mathrm{R}| - \frac{1}{2}tr(\sum_{t=1}^{n}\mathrm{R}^{-1}(\mathbf{y}_t - \mathrm{C}\mathbf{x}_t)(\mathbf{y}_t - \mathrm{C}\mathbf{x}_t)^\mathsf{T})\,; \end{aligned} \tag{2.16}$$

and by taking expectations we obtain the expected log-likelihood

$$
\begin{aligned}
G(\theta) = & \\
& -\frac{1}{2}log|\Lambda_0| - \frac{1}{2}tr(\Lambda_0^{-1}E\{(\mathbf{x}_0 - \mu_0)(\mathbf{x}_0 - \mu_0)^\mathsf{T}|\mathbf{y}\}) \\
& -\frac{n}{2}log|\mathrm{Q}| - \frac{1}{2}tr(\sum_{t=1}^{n}\mathrm{Q}^{-1}E\{(\mathbf{x}_t - \mathrm{A}\mathbf{x}_{t-1} - \mathrm{B}\mathbf{u}_t)(\mathbf{x}_t - \mathrm{A}\mathbf{x}_{t-1} - \mathrm{B}\mathbf{u}_t)^\mathsf{T}|\mathbf{y}\}) \\
& -\frac{n}{2}log|\mathrm{R}| - \frac{1}{2}tr(\sum_{t=1}^{n}\mathrm{R}^{-1}E\{(\mathbf{y}_t - \mathrm{C}\mathbf{x}_t)(\mathbf{y}_t - \mathrm{C}\mathbf{x}_t)^\mathsf{T}|\mathbf{y}\}) \,. \qquad (2.17)
\end{aligned}
$$

To compute the conditional expectations in eq. 2.17 we will need the smoothed estimates for $\mathbf{x}$,

$$
\begin{aligned}
\mathbf{x}_{i|n} &\triangleq E(\mathbf{x}_i|\mathbf{y}_1, \ldots, \mathbf{y}_n), i = 1, \ldots, n & (2.18) \\
\Lambda_{i|n} &\triangleq E((\mathbf{x}_i - \mathbf{x}_{i|n})(\mathbf{x}_i - \mathbf{x}_{i|n})^\mathsf{T}|\mathbf{y}_1, \ldots, \mathbf{y}_n), i = 1, \ldots, n. & (2.19)
\end{aligned}
$$

which will be efficiently computed with the help of the Kalman smoother.

2.3.2 The Kalman smoother

In the E step, we want to obtain maximum likelihood estimates for the non-observed values $\mathbf{x}$ given the observed values $\mathbf{y}$. On the SSM described by Eq. 2.1-2.2 this can be done quite efficiently using the Kalman smoother first proposed in [18]. The KS is a two pass algorithm: the forward pass computes the filtered estimates, *i.e.* an estimate that uses all present and past data relative to the value to be estimated; and the backward pass computes the smoothed estimates, *i.e.* an estimate that uses all present, past and future data relative to the value to be estimated. For the forward pass the reader can consult the original work by Kalman. For the smoother, a particularly elegant derivation can be found in [2], although the results were first obtained for the continuous case in [14].

2.3.3 Forward pass: filtering

In the first pass we compute

$$\mathbf{x}_{i|i} \triangleq E(\mathbf{x}_i|\mathbf{y}_1,\ldots,\mathbf{y}_i), i=1,\ldots,n \tag{2.20}$$

$$\Lambda_{i|i} \triangleq E((\mathbf{x}_i-\mathbf{x}_{i|i})(\mathbf{x}_i-\mathbf{x}_{i|i})^{\mathsf{T}}|\mathbf{y}_1,\ldots,\mathbf{y}_i), i=1,\ldots,n. \tag{2.21}$$

First we obtain $\Lambda_{1|1}$ using the initial conditions

$$\Lambda_{1|0} = \Lambda_0, \tag{2.22}$$

$$\mathrm{K}_1 = \Lambda_{1|0}\mathrm{C}^{\mathsf{T}}(\mathrm{C}\Lambda_{1|0}\mathrm{C}^{\mathsf{T}}+\mathrm{R})^{-1}, \tag{2.23}$$

$$\Lambda_{1|1} = \Lambda_{1|0}-\mathrm{K}\mathrm{C}\Lambda_{1|0}; \tag{2.24}$$

and then compute $\Lambda_{i|i}$ for the remaining steps using the recursion $\Lambda_{i-1|i-1} \rightarrow \Lambda_{i|i}$ given by

$$\Lambda_{i|i-1} = \mathrm{A}\Lambda_{i-1|i-1}\mathrm{A}^{\mathsf{T}}+\mathrm{Q} \tag{2.25}$$

$$\mathrm{K}_i = \Lambda_{i|i-1}\mathrm{C}^{\mathsf{T}}(\mathrm{C}\Lambda_{\mathrm{i|i-1}}\mathrm{C}^{\mathsf{T}}+\mathrm{R})^{-1}, \tag{2.26}$$

$$\Lambda_{i|i} = \Lambda_{i|i-1}-\mathrm{K}\mathrm{C}\Lambda_{i|i-1}. \tag{2.27}$$

Once we have obtained the values for the Kalman gains K_i, we can compute $\mathbf{x}_{1|1}$ using the initial conditions

$$\mathbf{x}_{1|0} = \mu_0, \tag{2.28}$$

$$\mathbf{x}_{1|1} = \mathbf{x}_{1|0}+\mathrm{K}_1(\mathbf{y}_1-\mathrm{C}\mathbf{x}_{1|0}); \tag{2.29}$$

and then compute $\mathbf{x}_{i|i}$ for the remaining steps using the recursion $\mathbf{x}_{i-1|i-1} \rightarrow \mathbf{x}_{i|i}$ given by

$$\mathbf{x}_{i|i-1} = \mathrm{A}\mathbf{x}_{i-1|i-1}, \tag{2.30}$$

$$\mathbf{x}_{i|i} = \mathbf{x}_{i|i-1}+\mathrm{K}_i(\mathbf{y}_i-\mathrm{C}\mathbf{x}_{i|i-1}). \tag{2.31}$$

2.3.4 Backward pass: smoothing

In the second pass we compute the smoothed estimates

$$\mathbf{x}_{i|n} \triangleq E(\mathbf{x}_i|\mathbf{y}_1,\ldots,\mathbf{y}_i,\ldots,\mathbf{y}_n), i=1,\ldots,n \tag{2.32}$$

$$\Lambda_{i|n} \triangleq E((\mathbf{x}_i-\mathbf{x}_{i|n})(\mathbf{x}_i-\mathbf{x}_{i|n})^{\mathsf{T}}|\mathbf{y}_1,\ldots,\mathbf{y}_i,\ldots,\mathbf{y}_n), i=1,\ldots,n, \tag{2.33}$$

where the notation stresses the fact that the smoothed estimate $\mathbf{x}_{i|n}$ incorporates future measurements $\{\mathbf{y}_i, i>n\}$.

As was the case in the forward case, we can first compute the smoothed error covariances (Eq. 2.19) and then proceed to compute the filtered estimates (Eq. 2.18). The initial conditions are trivial for the smoother, since for time step $i=n$ the filtered and smoothed estimate and error covariance are the same.

We obtain the smoothed error covariances using the recursion $\Lambda_{i|n} \rightarrow \Lambda_{i-1|n}$ given by

$$\mathrm{S}_i = \Lambda_{i|i}\mathrm{A}^{\mathsf{T}}\Lambda_{i+1|i}^{-1}, \tag{2.34}$$

$$\Lambda_{i|n} = \Lambda i|i - \mathrm{S}_i(\Lambda_{i+1|i}-\Lambda_{i+1|n})\mathrm{S}_i{}^{\mathsf{T}}; \tag{2.35}$$

and then compute the smoothed estimates for the previous steps using the values of the backward Kalman gains $\{\mathrm{S}_i\}$ and the recursion $\mathbf{x}_{i|n} \rightarrow \mathbf{x}_{i-1|n}$ given by

$$\mathbf{x}_{i|n} = \mathbf{x}_{i|i} + \mathrm{S}_i(\mathbf{x}_{i+1|n}-\mathbf{x}_{i+1|i}) \tag{2.36}$$

2.3.5 Remarks

For reasons that will be discussed later in Section 2.5.3, it is important to note that:

- The computation of the filtered and smoothed values (Eq. 2.20 and 2.18) can be done independently from that of the error covariances (Eq. 2.21 and 2.19), by first computing the error covariances and then using the resulting $\{\mathrm{K}_i, \mathrm{S}_i\}$ to compute the filtered and smoothed values.

- Both the filtered and smoothed error covariances depend solely on the parameters of the model $\{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{Q}, \mathrm{R}, \mu_0, \Lambda_0\}$.

2.3.6 Sufficient statistics

Once we run the KS on the data and obtain the smoothed estimates for $\mathbf{x}$ and its corresponding smoothed covariances, Eq. 2.17 can be rewritten by expanding the associated conditional expectations

$$E\{(\mathbf{x}_0 - \mu_0)(\mathbf{x}_0 - \mu_0)^{\mathsf{T}}|\mathbf{y}\} = \Lambda_{0|n} + (\mathbf{x}_{0|n} - \mu_0)(\mathbf{x}_{0|n} - \mu_0)^{\mathsf{T}}, \tag{2.37}$$

$$\begin{aligned} E\{((\mathbf{x}_t - \mathrm{A}\mathbf{x}_{t-1} - \mathrm{B}\mathbf{u}_t)(\mathbf{x}_t - \mathrm{A}\mathbf{x}_{t-1} - \mathrm{B}\mathbf{u}_t)^{\mathsf{T}}|\mathbf{y}\} = \\ (\mathbf{x}_{t|n} - \mathrm{A}\mathbf{x}_{t-1|n} - \mathrm{B}\mathbf{u}_t)(\mathbf{x}_{t|n} - \mathrm{A}\mathbf{x}_{t-1|n} - \mathrm{B}\mathbf{u}_t)^{\mathsf{T}} \\ +\Lambda_{t|n} + \mathrm{A}\Lambda_{t-1|n}\mathrm{A}^{\mathsf{T}} - \Lambda_{t,t-1|n}\mathrm{A}^{\mathsf{T}} - \mathrm{A}\Lambda_{t-1,t|n}{}^{\mathsf{T}} \\ = \Lambda_{t|n} + \mathbf{x}_{t|n}\mathbf{x}_{t|n}{}^{\mathsf{T}} + \mathrm{A}(\Lambda_{t-1|n} + \mathbf{x}_{t-1|n}\mathbf{x}_{t-1|n}{}^{\mathsf{T}})\mathrm{A}^{\mathsf{T}} - (\Lambda_{t,t-1|n} + \mathbf{x}_{t|n}\mathbf{x}_{t-1|n}{}^{\mathsf{T}})\mathrm{A}^{\mathsf{T}} \\ - \mathrm{A}(\Lambda_{t-1,t|n} + \mathbf{x}_{t-1|n}\mathbf{x}_{t|n}{}^{\mathsf{T}})^{\mathsf{T}} + \mathrm{B}\mathbf{u}_t\mathbf{u}_t{}^{\mathsf{T}}\mathrm{B}^{\mathsf{T}} - \mathrm{B}\mathbf{u}_t\mathbf{x}_{t|n}{}^{\mathsf{T}} + \mathrm{B}\mathbf{u}_t\mathbf{x}_{t-1|n}{}^{\mathsf{T}}\mathrm{A}^{\mathsf{T}} \\ - \mathbf{x}_{t|n}\mathbf{u}_t{}^{\mathsf{T}}\mathrm{B}^{\mathsf{T}} + \mathrm{A}\mathbf{x}_{t-1|n}\mathbf{u}_t{}^{\mathsf{T}}\mathrm{B}^{\mathsf{T}}, \end{aligned} \tag{2.38}$$

$$E\{(\mathbf{y}_t - \mathrm{C}\mathbf{x}_t)(\mathbf{y}_t - \mathrm{C}\mathbf{y}_t)^{\mathsf{T}}|\mathbf{y}\} = (\mathbf{y}_t - \mathrm{C}\mathbf{x}_{t|n})(\mathbf{y}_t - \mathrm{C}\mathbf{x}_{t|n})^{\mathsf{T}} + \mathrm{C}\Lambda_{t|n}\mathrm{C}^{\mathsf{T}} \tag{2.39}$$

where

$$\Lambda_{i,j|n} \triangleq E\{\mathbf{x}_i\mathbf{x}_j{}^{\mathsf{T}}|\mathbf{y}_0, \ldots, \mathbf{y}_n\}. \tag{2.40}$$

For clarity purposes, this expressions can be simplified using

$$\mathrm{N}_1^t \triangleq \Lambda_{t|n} + \mathbf{x}_{t|n}\mathbf{x}_{t|n}^{\mathsf{T}}, \tag{2.41}$$

$$\mathrm{N}_2^t \triangleq \mathrm{N}_1^{t-1}, \tag{2.42}$$

$$\mathrm{N}_3^t \triangleq \Lambda_{t,t-1|n} + \mathbf{x}_{t|n}\mathbf{x}_{t-1|n}^{\mathsf{T}}, \tag{2.43}$$

$$\mathrm{N}_4^t \triangleq \mathbf{x}_{t|n}\mathbf{u}_t^{\mathsf{T}}, \tag{2.44}$$

$$\mathrm{N}_5^t \triangleq \mathbf{x}_{t-1|n}\mathbf{u}_t^{\mathsf{T}}, \tag{2.45}$$

$$\mathrm{N}_6^t \triangleq \mathbf{x}_{t|n}\mathbf{y}_t^{\mathsf{T}}, \text{and} \tag{2.46}$$

$$\mathrm{N}_7^t \triangleq \mathbf{y}_t\mathbf{y}_t^{\mathsf{T}} \tag{2.47}$$

to rewrite 2.17 as

$$\begin{aligned} G(\theta) = & \\ & -\frac{1}{2}log|\Lambda_0| - \frac{1}{2}tr(\Lambda_0^{-1}(\Lambda_{0|n} + (\mathbf{x}_{0|n} - \mu_0)(\mathbf{x}_{0|n} - \mu_0)^{\mathsf{T}})) - \frac{n}{2}log|\mathrm{Q}| \\ & -\frac{1}{2}tr(\sum_{t=1}^{n}\mathrm{Q}^{-1}(\mathrm{N}_1^t + \mathrm{A}\mathrm{N}_2^t\mathrm{A}^{\mathsf{T}} - \mathrm{N}_3^t\mathrm{A}^{\mathsf{T}} - \mathrm{A}\mathrm{N}_3^{t\mathsf{T}} + \mathrm{B}\mathbf{u}_t\mathbf{u}_t^{\mathsf{T}}\mathrm{B}^{\mathsf{T}} - \mathrm{B}\mathrm{N}_4^{t\mathsf{T}} - \mathrm{N}_4^t\mathrm{B}^{\mathsf{T}} \\ & \qquad +\mathrm{A}\mathrm{N}_5^t\mathrm{B}^{\mathsf{T}} + \mathrm{B}\mathrm{N}_5^{t\mathsf{T}}\mathrm{A}^{\mathsf{T}})) \\ & -\frac{n}{2}log|\mathrm{R}| - \frac{1}{2}tr(\sum_{t=1}^{n}\mathrm{R}^{-1}(\mathrm{N}_7^t - \mathrm{C}\mathrm{N}_6^t - (\mathrm{C}\mathrm{N}_6^t)^{\mathsf{T}} + \mathrm{C}\mathrm{N}_1^t\mathrm{C}^{\mathsf{T}})). \end{aligned} \tag{2.48}$$

Finally, by defining the sufficient statistics

$$\mathrm{N}_i \triangleq \sum_{t=1}^{n}\mathrm{N}_i^t \quad \text{for } i = 1, \ldots, 7, \text{and} \tag{2.49}$$

$$\mathrm{U} \triangleq \sum_{t=1}^{n}\mathbf{u}_t\mathbf{u}_t^{\mathsf{T}}, \tag{2.50}$$

we can rewrite 2.48 as

$$
\begin{aligned}
G(\theta) = & \\
& -\frac{1}{2}log|\Lambda_0| - \frac{1}{2}tr(\Lambda_0^{-1}(\Lambda_{0|n} + (\mathbf{x}_{0|n} - \mu_0)(\mathbf{x}_{0|n} - \mu_0)^{\mathsf{T}})) \\
& -\frac{n}{2}log|\mathrm{Q}| - \frac{1}{2}tr(\mathrm{Q}^{-1}(\mathrm{N}_1 + \mathrm{A}\mathrm{N}_2\mathrm{A}^{\mathsf{T}} - \mathrm{N}_3\mathrm{A}^{\mathsf{T}} - \mathrm{A}\mathrm{N}_3{}^{\mathsf{T}} + \mathrm{B}\mathrm{U}\mathrm{B}^{\mathsf{T}} \\
& \qquad\qquad -\mathrm{B}\mathrm{N}_4{}^{\mathsf{T}} - \mathrm{N}_4\mathrm{B}^{\mathsf{T}} + \mathrm{A}\mathrm{N}_5\mathrm{B}^{\mathsf{T}} + \mathrm{B}\mathrm{N}_5{}^{\mathsf{T}}\mathrm{A}^{\mathsf{T}})) \\
& -\frac{n}{2}log|\mathrm{R}| - \frac{1}{2}tr(\mathrm{R}^{-1}(\mathrm{N}_7 - \mathrm{C}\mathrm{N}_6 - (\mathrm{C}\mathrm{N}_6)^{\mathsf{T}} + \mathrm{C}\mathrm{N}_1\mathrm{C}^{\mathsf{T}}))\,. \qquad (2.51)
\end{aligned}
$$

2.4 M step

In this section we compute the M updates for all the parameters of the SSM, θ. Using the expression for $G(\theta)$ obtained in the last section, we will differentiate with respect to the parameters and find the new values that produce zero derivative, thus maximizing the expected log-likelihood as required by the EM algorithm.

In Section 2.4.1 we assume all the parameters θ are fully parameterized. Although this is a valid strategy, it is important to note that the number of parameters grows quadratically with the dimensions of the system, most notably so with A and Q which depend on the dimensions of the hidden states. A fully parameterized A update will have too many degrees of freedom in the case the hidden states dimensions are high, which will be the case the models we will discuss later on. For this reason, in Section 2.4.2 we propose an alternative M step in which the matrix A is constrained to be in the subspace spanned by a set of candidate matrices, $\{\mathrm{A}_i\}$, greatly reducing the number of parameters to estimate. This formulation will be exploited in both KronEM and StimEM. We will also contemplate the case in which part of the A matrix is fixed, which will be the case in the KronEM algorithm, where the hidden state describes AR processes and need to incorporate previous values in the transition matrix. The new update resulting from this is derived in Section 2.4.3.

2.4.1 Fully parameterized θ

As discussed in Section 2.2, in the M step we set to maximize $G(\theta)$, and we do so finding the values $\{\hat{\mathrm{A}}, \hat{\mathrm{B}}, \hat{\mathrm{C}}, \hat{\mathrm{Q}}, \hat{\mathrm{R}}\}$ that give a root on the partial derivatives:

$$\begin{aligned}
\frac{\partial}{\partial \mathrm{A}} G(\theta) = & \\
& -\frac{1}{2}\frac{\partial}{\partial \mathrm{A}} tr(\mathrm{Q}^{-1}(\mathrm{N}_1 + \mathrm{A}\mathrm{N}_2\mathrm{A}^{\mathsf{T}} - \mathrm{N}_3\mathrm{A}^{\mathsf{T}} - \mathrm{A}\mathrm{N}_3^{\mathsf{T}} + \mathrm{B}\mathrm{U}\mathrm{B}^{\mathsf{T}} \\
& \qquad -\mathrm{B}\mathrm{N}_4^{\mathsf{T}} - \mathrm{N}_4\mathrm{B}^{\mathsf{T}} + \mathrm{A}\mathrm{N}_5\mathrm{B}^{\mathsf{T}} + \mathrm{B}\mathrm{N}_5^{\mathsf{T}}\mathrm{A}^{\mathsf{T}})) \\
= & -\frac{1}{2}\frac{\partial}{\partial \mathrm{A}} tr(\mathrm{Q}^{-1}(\mathrm{N}_1 + \mathrm{B}\mathrm{U}\mathrm{B}^{\mathsf{T}} - \mathrm{B}\mathrm{N}_4^{\mathsf{T}} - \mathrm{N}_4\mathrm{B}^{\mathsf{T}} + \mathrm{A}\mathrm{N}_2\mathrm{A}^{\mathsf{T}} \\
& \qquad +\mathrm{A}(\mathrm{N}_5\mathrm{B}^{\mathsf{T}} - \mathrm{N}_3^{\mathsf{T}}) + (\mathrm{B}\mathrm{N}_5^{\mathsf{T}} - \mathrm{N}_3)\mathrm{A}^{\mathsf{T}})) \\
= & -\frac{1}{2}\frac{\partial}{\partial \mathrm{A}} tr(\mathrm{Q}^{-1}(\mathrm{A}\mathrm{N}_2\mathrm{A}^{\mathsf{T}} + \mathrm{A}(\mathrm{N}_5\mathrm{B}^{\mathsf{T}} - \mathrm{N}_3^{\mathsf{T}}) + (\mathrm{B}\mathrm{N}_5^{\mathsf{T}} - \mathrm{N}_3)\mathrm{A}^{\mathsf{T}})) \qquad (2.52) \\
= & -\frac{1}{2}(\mathrm{Q}^{-1\mathsf{T}}\mathrm{A}\mathrm{N}_2^{\mathsf{T}} + \mathrm{Q}^{-1}\mathrm{A}\mathrm{N}_2 + 2\mathrm{Q}^{-1}(\mathrm{B}\mathrm{N}_5^{\mathsf{T}} - \mathrm{N}_3)) \\
= & \mathrm{Q}^{-1}(\mathrm{A}\mathrm{N}_2 + \mathrm{B}\mathrm{N}_5^{\mathsf{T}} - \mathrm{N}_3)\,, \qquad (2.53) \\
0 = & \hat{\mathrm{Q}}^{-1}(\hat{\mathrm{A}}\mathrm{N}_2 + \hat{\mathrm{B}}\mathrm{N}_5^{\mathsf{T}} - \mathrm{N}_3)\,, \\
\hat{\mathrm{A}} = & (\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5^{\mathsf{T}})\mathrm{N}_2{}^{-1}\,. \qquad (2.54)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial}{\partial \mathrm{B}} G(\theta) = & \\
& -\frac{1}{2}\frac{\partial}{\partial \mathrm{B}} tr(\mathrm{Q}^{-1}(\mathrm{N}_1 + \mathrm{A}\mathrm{N}_2\mathrm{A}^{\mathsf{T}} - \mathrm{N}_3\mathrm{A}^{\mathsf{T}} - \mathrm{A}\mathrm{N}_3^{\mathsf{T}} + \mathrm{B}\mathrm{U}\mathrm{B}^{\mathsf{T}} \\
& \qquad -\mathrm{B}\mathrm{N}_4^{\mathsf{T}} - \mathrm{N}_4\mathrm{B}^{\mathsf{T}} + \mathrm{A}\mathrm{N}_5\mathrm{B}^{\mathsf{T}} + \mathrm{B}\mathrm{N}_5^{\mathsf{T}}\mathrm{A}^{\mathsf{T}})) \\
= & -\frac{1}{2}\frac{\partial}{\partial \mathrm{B}} tr(\mathrm{Q}^{-1}(\mathrm{B}\mathrm{U}\mathrm{B}^{\mathsf{T}} + \mathrm{B}(\mathrm{N}_5^{\mathsf{T}}\mathrm{A}^{\mathsf{T}} - \mathrm{N}_4^{\mathsf{T}}) + (\mathrm{A}\mathrm{N}_5 - \mathrm{N}_4)\mathrm{B}^{\mathsf{T}}) \\
= & \mathrm{Q}^{-1}(\mathrm{B}\mathrm{U} + \mathrm{A}\mathrm{N}_5 - \mathrm{N}_4)\,, \qquad (2.55) \\
0 = & \hat{\mathrm{Q}}^{-1}(\hat{\mathrm{B}}\mathrm{U} + \hat{\mathrm{A}}\mathrm{N}_5 - \mathrm{N}_4)\,, \\
\hat{\mathrm{B}} = & (\mathrm{N}_4 - \hat{\mathrm{A}}\mathrm{N}_5)\mathrm{U}^{-1}\,. \qquad (2.56)
\end{aligned}$$

$$
\begin{aligned}
&\frac{\partial}{\partial \mathrm{C}} G(\theta) = \\
&\qquad -\frac{1}{2}\frac{\partial}{\partial \mathrm{C}} tr(\mathrm{R}^{-1}(\mathrm{C}\mathrm{N}_1\mathrm{C}^{\mathsf{T}} - \mathrm{C}\mathrm{N}_6 - (\mathrm{C}\mathrm{N}_6)^{\mathsf{T}}) \\
&= -\frac{1}{2}\mathrm{R}^{-1}(\mathrm{C}\mathrm{N}_1{}^{\mathsf{T}} + \mathrm{C}\mathrm{N}_1 - 2\mathrm{N}_6{}^{\mathsf{T}}), \qquad (2.57) \\
0 &= -\frac{1}{2}\hat{\mathrm{R}}^{-1}(\hat{\mathrm{C}}\mathrm{N}_1{}^{\mathsf{T}} + \hat{\mathrm{C}}\mathrm{N}_1 - 2\mathrm{N}_6{}^{\mathsf{T}}), \\
\hat{\mathrm{C}} &= \mathrm{N}_6{}^{\mathsf{T}}\mathrm{N}_1^{-1}. \qquad (2.58)
\end{aligned}
$$

$$
\begin{aligned}
&\frac{\partial}{\partial \mathrm{Q}} G(\theta) = \\
&\qquad -\frac{n}{2}\mathrm{Q}^{-1} - \frac{1}{2}\frac{\partial}{\partial \mathrm{Q}} tr(\mathrm{Q}^{-1}(\mathrm{N}_1 + \mathrm{A}\mathrm{N}_2\mathrm{A}^{\mathsf{T}} - \mathrm{N}_3\mathrm{A}^{\mathsf{T}} - \mathrm{A}\mathrm{N}_3{}^{\mathsf{T}} + \mathrm{B}\mathrm{U}\mathrm{B}^{\mathsf{T}} \\
&\qquad\qquad -\mathrm{B}\mathrm{N}_4{}^{\mathsf{T}} - \mathrm{N}_4\mathrm{B}^{\mathsf{T}} + \mathrm{A}\mathrm{N}_5\mathrm{B}^{\mathsf{T}} + \mathrm{B}\mathrm{N}_5{}^{\mathsf{T}}\mathrm{A}^{\mathsf{T}})) \\
&= -\frac{n}{2}\mathrm{Q}^{-1} + \frac{1}{2}\mathrm{Q}^{-1}(\mathrm{N}_1 + \mathrm{A}\mathrm{N}_2\mathrm{A}^{\mathsf{T}} - (\mathrm{N}_3 - \mathrm{B}\mathrm{N}_5{}^{\mathsf{T}})\mathrm{A}^{\mathsf{T}} - \mathrm{A}(\mathrm{N}_3 - \mathrm{B}\mathrm{N}_5{}^{\mathsf{T}})^{\mathsf{T}} \\
&\qquad\qquad +\mathrm{B}\mathrm{U}\mathrm{B}^{\mathsf{T}} - \mathrm{B}\mathrm{N}_4{}^{\mathsf{T}} - \mathrm{N}_4\mathrm{B}^{\mathsf{T}})\mathrm{Q}^{-1}, \qquad (2.59) \\
0 &= -\frac{n}{2}\hat{\mathrm{Q}}^{-1} + \frac{1}{2}\hat{\mathrm{Q}}^{-1}(\mathrm{N}_1 + \hat{\mathrm{A}}\mathrm{N}_2\hat{\mathrm{A}}^{\mathsf{T}} - (\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})\hat{\mathrm{A}}^{\mathsf{T}} - \hat{\mathrm{A}}(\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})^{\mathsf{T}} + \hat{\mathrm{B}}\mathrm{U}\hat{\mathrm{B}}^{\mathsf{T}} \\
&\qquad\qquad -\hat{\mathrm{B}}\mathrm{N}_4{}^{\mathsf{T}} - \mathrm{N}_4\hat{\mathrm{B}}^{\mathsf{T}})\hat{\mathrm{Q}^{-1}}, \\
\hat{\mathrm{Q}} &= \frac{1}{n}(\mathrm{N}_1 + \hat{\mathrm{A}}\mathrm{N}_2\hat{\mathrm{A}}^{\mathsf{T}} - (\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})\hat{\mathrm{A}}^{\mathsf{T}} - \hat{\mathrm{A}}(\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})^{\mathsf{T}} \\
&\qquad\qquad +\hat{\mathrm{B}}\mathrm{U}\hat{\mathrm{B}}^{\mathsf{T}} - \hat{\mathrm{B}}\mathrm{N}_4{}^{\mathsf{T}} - \mathrm{N}_4\hat{\mathrm{B}}^{\mathsf{T}}). \qquad (2.60)
\end{aligned}
$$

$$
\begin{aligned}
&\frac{\partial}{\partial \mathrm{R}} G(\theta) = \\
&= -\frac{n}{2}\mathrm{R}^{-1} - \frac{1}{2}\frac{\partial}{\partial \mathrm{R}} tr(\mathrm{R}^{-1}(\mathrm{N}_7 - \mathrm{C}\mathrm{N}_6 - (\mathrm{C}\mathrm{N}_6)^{\mathsf{T}} + \mathrm{C}\mathrm{N}_1\mathrm{C}^{\mathsf{T}})) \\
&= -\frac{n}{2}\mathrm{R}^{-1} + \frac{1}{2}\mathrm{R}^{-1}(\mathrm{N}_7 - \mathrm{C}\mathrm{N}_6 - (\mathrm{C}\mathrm{N}_6)^{\mathsf{T}} + \mathrm{C}\mathrm{N}_1\mathrm{C}^{\mathsf{T}})\mathrm{R}^{-1}, \qquad (2.61) \\
0 &= -\frac{n}{2}\hat{\mathrm{R}}^{-1} + \frac{1}{2}\hat{\mathrm{R}}^{-1}(\mathrm{N}_7 - \hat{\mathrm{C}}\mathrm{N}_6 - (\hat{\mathrm{C}}\mathrm{N}_6)^{\mathsf{T}} + \hat{\mathrm{C}}\mathrm{N}_1\hat{\mathrm{C}}^{\mathsf{T}})\hat{\mathrm{R}}^{-1}, \\
\hat{\mathrm{R}} &= \frac{1}{n}(\mathrm{N}_7 - \hat{\mathrm{C}}\mathrm{N}_6 - (\hat{\mathrm{C}}\mathrm{N}_6)^{\mathsf{T}} + \hat{\mathrm{C}}\mathrm{N}_1\hat{\mathrm{C}}^{\mathsf{T}}). \qquad (2.62)
\end{aligned}
$$

In solving for the above we used the matrix derivatives (from [71])

$$\frac{\partial}{\partial \mathrm{X}} tr(\mathrm{AXBX}^{\mathsf{T}}) = \mathrm{A}^{\mathsf{T}}\mathrm{XB}^{\mathsf{T}} + \mathrm{AXB}, \tag{2.63}$$

$$\frac{\partial}{\partial \mathrm{X}} tr(\mathrm{AXB}) = \frac{\partial}{\partial \mathrm{X}} tr(\mathrm{B}^{\mathsf{T}}\mathrm{X}^{\mathsf{T}}\mathrm{A}^{\mathsf{T}}) = \mathrm{A}^{\mathsf{T}}\mathrm{B}^{\mathsf{T}}, \tag{2.64}$$

$$\frac{\partial}{\partial \mathrm{X}} tr(\mathrm{X}^{-1}\mathrm{B}) = -(\mathrm{X}^{-1}\mathrm{BX}^{-1})^{\mathsf{T}}, \tag{2.65}$$

$$\frac{\partial}{\partial \mathrm{X}} log|\mathrm{X}| = (\mathrm{X}^{-1})^{\mathsf{T}}, \tag{2.66}$$

We can see that in addition to the sufficient statistics N_i, $\hat{\mathrm{Q}}$ depends on $\hat{\mathrm{A}}$ and $\hat{\mathrm{B}}$, $\hat{\mathrm{B}}$ depends on $\hat{\mathrm{A}}$, and $\hat{\mathrm{A}}$ depends on $\hat{\mathrm{B}}$. So when solving simultaneously for $\hat{\mathrm{A}}$ and $\hat{\mathrm{B}}$:

$$\hat{\mathrm{A}} = (\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})\mathrm{N}_2{}^{-1} \tag{2.67}$$

$$= (\mathrm{N}_3 - (\mathrm{N}_4 - \hat{\mathrm{A}}\mathrm{N}_5)\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}})\mathrm{N}_2{}^{-1}, \tag{2.68}$$

$$\hat{\mathrm{A}}\mathrm{N}_2 = \mathrm{N}_3 - \mathrm{N}_4\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}} + \hat{\mathrm{A}}\mathrm{N}_5\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}}, \tag{2.69}$$

$$\hat{\mathrm{A}}(\mathrm{N}_2 - \mathrm{N}_5\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}}) = \mathrm{N}_3 - \mathrm{N}_4\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}}, \tag{2.70}$$

$$\hat{\mathrm{A}} = (\mathrm{N}_3 - \mathrm{N}_4\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}})(\mathrm{N}_2 - \mathrm{N}_5\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}})^{-1} \tag{2.71}$$

Piecing all together, the M step given the sufficient statistics computed from the former E step is

$$\hat{\mathrm{A}} = (\mathrm{N}_3 - \mathrm{N}_4\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}})(\mathrm{N}_2 - \mathrm{N}_5\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}})^{-1}, \tag{2.72}$$

$$\hat{\mathrm{B}} = (\mathrm{N}_4 - \hat{\mathrm{A}}\mathrm{N}_5)\mathrm{U}^{-1}, \tag{2.73}$$

$$\hat{\mathrm{C}} = \mathrm{N}_6{}^{\mathsf{T}}\mathrm{N}_1^{-1}, \tag{2.74}$$

$$\hat{\mathrm{Q}} = \frac{1}{n}(\mathrm{N}_1 + \hat{\mathrm{A}}\mathrm{N}_2\hat{\mathrm{A}}^{\mathsf{T}} - (\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})\hat{\mathrm{A}}^{\mathsf{T}} - \hat{\mathrm{A}}(\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})^{\mathsf{T}} + \hat{\mathrm{B}}\mathrm{U}\hat{\mathrm{B}}^{\mathsf{T}} - \hat{\mathrm{B}}\mathrm{N}_4{}^{\mathsf{T}} - \mathrm{N}_4\hat{\mathrm{B}}^{\mathsf{T}}), \text{and} \tag{2.75}$$

$$\hat{\mathrm{R}} = \frac{1}{n}(\mathrm{N}_7 - \hat{\mathrm{C}}\mathrm{N}_6 - (\hat{\mathrm{C}}\mathrm{N}_6)^{\mathsf{T}} + \hat{\mathrm{C}}\mathrm{N}_1\hat{\mathrm{C}}^{\mathsf{T}}). \tag{2.76}$$

2.4.2 Constrained A matrix

The number of free elements in A grows quadratically with the dimensionality of the hidden-state $\mathbf{x}$, and for a given number of observations, increasing the number of free parameters will produce unreliable estimates. In order to reduce the degrees of freedom introduced by A, we can add the constraint

$$\mathrm{A} = \sum_{i=1}^{N_a} \alpha_i \mathrm{A}_i \,, \tag{2.77}$$

so that the constrained state transition matrix $\hat{\mathrm{A}}$ can be described by the basis $\{\mathrm{A}_i\}$ and a set of weights $\{\hat{\alpha}_i\}$. The elements of the basis $\{\mathrm{A}_i\}$ can be chosen so that they are physiologically plausible. For example, when the hidden states $\mathbf{x}$ correspond to brain activity on the cortex, each A_i can correspond to an exponentially decaying function, as shown in figure 2-1.

In this scenario, the equation 2.52 turns into

$$\begin{aligned}
\frac{\partial}{\partial \alpha_{\mathrm{i}}} G(\theta) = & \\
& -\frac{1}{2}\frac{\partial}{\partial \alpha_{\mathrm{i}}} tr(\mathrm{Q}^{-1}(\mathrm{A}\mathrm{N}_2\mathrm{A}^{\mathsf{T}} + \mathrm{A}(\mathrm{N}_5\mathrm{B}^{\mathsf{T}} - \mathrm{N}_3{}^{\mathsf{T}}) + (\mathrm{B}\mathrm{N}_5{}^{\mathsf{T}} - \mathrm{N}_3)\mathrm{A}^{\mathsf{T}})) \\
= & -\frac{1}{2}\frac{\partial}{\partial \alpha_{\mathrm{i}}} tr(\mathrm{Q}^{-1}(\sum_{i=1}^{N_a}\sum_{i=j}^{N_a} \alpha_i\alpha_j \mathrm{A}_{\mathrm{i}}\mathrm{N}_2\mathrm{A}_{\mathrm{j}}{}^{\mathsf{T}})) - \frac{1}{2} tr(\mathrm{Q}^{-1}(\mathrm{A}_i(\mathrm{N}_5\mathrm{B}^{\mathsf{T}} - \mathrm{N}_3{}^{\mathsf{T}}) \\
& \qquad\qquad +(\mathrm{B}\mathrm{N}_5{}^{\mathsf{T}} - \mathrm{N}_3)\mathrm{A}_i{}^{\mathsf{T}})) \\
= & -\frac{1}{2}\sum_{i=1}^{N_a}\sum_{j=1}^{N_a} \frac{\partial}{\partial \alpha_{\mathrm{i}}} (\alpha_i\alpha_j tr(\mathrm{Q}^{-1}\mathrm{A}_{\mathrm{i}}\mathrm{N}_2\mathrm{A}_{\mathrm{j}}{}^{\mathsf{T}})) + \frac{1}{2}c_i + \frac{1}{2}\psi_i(\mathrm{B}) \\
= & -\alpha_i tr(\mathrm{Q}^{-1}\mathrm{A}_{\mathrm{i}}\mathrm{N}_2\mathrm{A}_{\mathrm{i}}{}^{\mathsf{T}}) - \frac{1}{2}\sum_{j\neq i} \alpha_j tr(\mathrm{Q}^{-1}(\mathrm{A}_{\mathrm{i}}\mathrm{N}_2\mathrm{A}_{\mathrm{j}}{}^{\mathsf{T}} + \mathrm{A}_{\mathrm{j}}\mathrm{N}_2\mathrm{A}_{\mathrm{i}}{}^{\mathsf{T}})) + \frac{1}{2}c_i + \frac{1}{2}\psi_i(\mathrm{B}) \\
= & -\frac{1}{2}\sum_{j=1}^{N_a} \alpha_j \mathrm{T}_{i,j} + \frac{1}{2}c_i + \frac{1}{2}\psi_i(\mathrm{B}) \,,
\end{aligned} \tag{2.78}$$

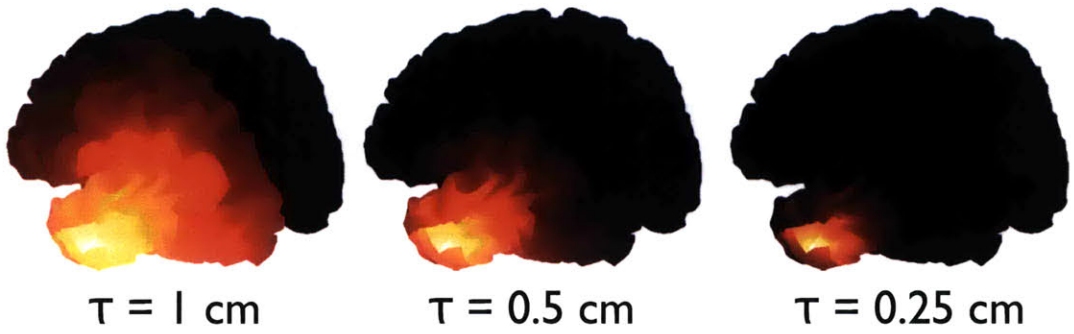


Figure 2-1: Illustration of a possible base $\{A_i\}$ with decaying functions of distance. We show topographic maps of a given row of A_i for different values of the spatial decay τ in the cortical surface.

where we defined

$$c_i \triangleq tr(\mathrm{Q}^{-1}(\mathrm{A}_i\mathrm{N}_3{}^{\mathsf{T}} + \mathrm{N}_3\mathrm{A}_i{}^{\mathsf{T}})) \tag{2.79}$$

$$\psi_i(\mathrm{B}) \triangleq -tr(\mathrm{Q}^{-1}(\mathrm{A}_i\mathrm{N}_5\mathrm{B}^{\mathsf{T}} + \mathrm{B}\mathrm{N}_5{}^{\mathsf{T}}\mathrm{A}_i{}^{\mathsf{T}})) \tag{2.80}$$

$$\mathrm{T}_{i,j} \triangleq tr(\mathrm{Q}^{-1}(\mathrm{A_i}\mathrm{N}_2\mathrm{A_j}{}^{\mathsf{T}} + \mathrm{A_j}\mathrm{N}_2\mathrm{A_i}{}^{\mathsf{T}}))\,. \tag{2.81}$$

To find the update for $\{\alpha_i\}$ we set

$$0 = \frac{\partial}{\partial\alpha_{\mathrm{i}}} G(\hat{\mathrm{A}}, \hat{\mathrm{B}}, \hat{\mathrm{Q}}, \hat{\mathrm{R}}), \tag{2.82}$$

$$0 = -\frac{1}{2}\sum_{j=1}^{N_a} \alpha_j \mathrm{T}_{i,j} + \frac{1}{2}c_i + \frac{1}{2}\psi_i(\hat{\mathrm{B}})\,. \tag{2.83}$$

Because this is true for all $i = 1 \ldots N_a$, we can express these set of equations as

$$\mathrm{T}\hat{\alpha} = \mathbf{c} + \psi(\hat{\mathrm{B}})\,. \tag{2.84}$$

Now, plugging 2.77 into 2.56, we get

$$\hat{\mathrm{B}} = (\mathrm{N}_4 - \sum_{i=1}^{N_a} \hat{\alpha}_i \mathrm{A}_i\mathrm{N}_5)\mathrm{U}^{-1}\,. \tag{2.85}$$

so that the M update for $\hat{\alpha}$ and $\hat{\mathrm{B}}$ have to be computed simultaneously for both.

In the case of $\mathrm{Q} = \lambda\mathrm{I}$, is easy to show that

$$c_i = \lambda^{-1}tr(\mathrm{A}_i\mathrm{N}_3{}^{\mathsf{T}} + \mathrm{N}_3\mathrm{A}_i{}^{\mathsf{T}}) = 2\lambda^{-1}tr(\mathrm{A}_i\mathrm{N}_3{}^{\mathsf{T}}) \tag{2.86}$$

$$\psi_i(\mathrm{B}) = -\lambda^{-1}tr(\mathrm{A}_i\mathrm{N}_5\mathrm{B}^{\mathsf{T}} + \mathrm{B}\mathrm{N}_5{}^{\mathsf{T}}\mathrm{A}_i{}^{\mathsf{T}}) = -2\lambda^{-1}tr(\mathrm{A}_i\mathrm{N}_5\mathrm{B}^{\mathsf{T}}) \tag{2.87}$$

$$\mathrm{T}_{i,j} = \lambda^{-1}tr(\mathrm{A_i}\mathrm{N}_2\mathrm{A_j}{}^{\mathsf{T}} + \mathrm{A_j}\mathrm{N}_2\mathrm{A_i}{}^{\mathsf{T}}) = 2\lambda^{-1}tr(\mathrm{A_i}\mathrm{N}_2\mathrm{A_j}{}^{\mathsf{T}})\,, \tag{2.88}$$

and because all λ terms cancel in 2.84, we can redefine them as

$$
\begin{aligned}
c_i^{us} &= 2tr(\mathrm{A}_i\mathrm{N}_3{}^{\mathsf{T}}) && (2.89)\\
\psi_i^{us}(\hat{\mathrm{B}}) &= -2tr(\mathrm{A}_i\mathrm{N}_5\hat{\mathrm{B}}^{\mathsf{T}}) && (2.90)\\
\mathrm{T}_{i,j}^{us} &= 2tr(\mathrm{A_i}\mathrm{N}_2\mathrm{A_j}{}^{\mathsf{T}}) && (2.91)\\
\mathrm{T}^{us}\hat{\alpha} &= \mathbf{c}^{us} + \psi^{us}(\hat{\mathrm{B}})\,. && (2.92)
\end{aligned}
$$

To solve for $\hat{\mathrm{B}}$ and ψ^{us}, we are going to use the $vec()$ operator, that converts a matrix in a column vector that has stacked in order all the columns of the original matrix. This way we can rewrite 2.90 as

$$
\psi_i^{us}(\mathrm{B}) = -2vec(\mathrm{A}_i\mathrm{N}_5)^{\mathsf{T}}vec(\hat{\mathrm{B}})\,, \tag{2.93}
$$

and defining

$$
\begin{aligned}
\mathrm{M} &\triangleq \begin{bmatrix} 2vec(\mathrm{A_1N_5})^{\mathsf{T}} \\ \vdots \\ 2vec(\mathrm{A}_{N_a}\mathrm{N}_5)^{\mathsf{T}} \end{bmatrix} && (2.94)\\
\mathrm{K} &\triangleq [vec(\mathrm{A_1N_5U^{-1}}, \ldots, vec(\mathrm{A}_{N_a}\mathrm{N_5U^{-1}})] && (2.95)
\end{aligned}
$$

we can rewrite 2.92 and 2.85 as

$$
\begin{aligned}
vec(\hat{\mathrm{B}}) + \mathrm{K}\hat{\alpha} &= vec(\mathrm{N_4U^{-1}}) && (2.96)\\
\mathrm{M}vec(\hat{\mathrm{B}}) + \mathrm{T}^{us}\hat{\alpha} &= \mathbf{c}^{us}\,. && (2.97)
\end{aligned}
$$

This way, we can solve the M step for both $\hat{\mathrm{B}}$ and $\hat{\alpha}$ by solving the system

$$
\begin{bmatrix} \mathrm{I}_{N_s^2N_u^2} & \mathrm{K} \\ \mathrm{M} & \mathrm{T}^{us} \end{bmatrix} \begin{bmatrix} vec(\hat{\mathrm{B}}) \\ \hat{\alpha} \end{bmatrix} = \begin{bmatrix} vec(\mathrm{N_4U^{-1}}) \\ \mathbf{c}^{us} \end{bmatrix}\,. \tag{2.98}
$$

The system is very sparse, because the dimensions of the square matrix of the left hand side of the equation are $N_s^2N_u^2 + N_a$, and out of those $N_s^2N_u^2$ are occupied by the

identity matrix. This sparsity can be exploited by using the block matrix inversion formula:

$$\begin{bmatrix} A & C \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} S_D^{-1} & -ABS_A^{-1} \\ -D^{-1}CS_D^{-1} & S_A^{-1} \end{bmatrix}, \tag{2.99}$$

where A and D are square non-singular matrices and

$$S_A = D - CA^{-1}B \tag{2.100}$$

$$S_D = A - BD^{-1}C \tag{2.101}$$

are the Schur complements of A and D, respectively. In our particular case, A = I, so that Eq. 2.99-2.101 turns into

$$\begin{bmatrix} I & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} S_D^{-1} & -BS_A^{-1} \\ -D^{-1}CS_D^{-1} & S_A^{-1} \end{bmatrix}, \tag{2.102}$$

$$S_A = D - CB\,, \text{and} \tag{2.103}$$

$$S_D = I - BD^{-1}C \tag{2.104}$$

If we apply this to Eq. 2.98 as it is we would still have to invert S_D, which is undesirable, as it is a square matrix $N_s^2 N_u^2$ columns wide. To avoid this, we make use of the Woodbury matrix identity,

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}\,, \tag{2.105}$$

which applied to Eq. 2.104 gets

$$S_D^{-1} = I + B(D - CB)^{-1}C = I + BS_A^{-1}C \tag{2.106}$$

which only needs the inversion of a N_a sided square matrix.

Piecing all together, we get

$$\begin{bmatrix} vec(\hat{\mathrm{B}}) \\ \hat{\alpha} \end{bmatrix} = \begin{bmatrix} \mathrm{I}_{N_s^2 N_u^2} & \mathrm{K} \\ \mathrm{M} & \mathrm{T}^{us} \end{bmatrix}^{-1} \begin{bmatrix} vec(\mathrm{N}_4\mathrm{U}^{-1}) \\ \mathbf{c}^{us} \end{bmatrix} \tag{2.107}$$

$$= \begin{bmatrix} \mathrm{I}_{N_s^2 N_u^2} + \mathrm{KZM} & -\mathrm{KZ} \\ -\mathrm{T}^{us-1}\mathrm{M}(\mathrm{I}_{N_s^2 N_u^2} + \mathrm{KZM}) & \mathrm{Z} \end{bmatrix} \begin{bmatrix} vec(\mathrm{N}_4\mathrm{U}^{-1}) \\ \mathbf{c}^{us} \end{bmatrix} \tag{2.108}$$

where

$$\mathrm{Z} = (\mathrm{T}^{us} - \mathrm{MK})^{-1} \tag{2.109}$$

and finally, the M update for this case is

$$\hat{\alpha} = -\mathrm{T}^{us-1}\mathrm{M}(\mathrm{I}_{N_s^2 N_u^2} + \mathrm{KZM})vec(\mathrm{N}_4\mathrm{U}^{-1}) + \mathrm{Z}\mathbf{c}^{us}, \tag{2.110}$$

$$\hat{\mathrm{A}} = \sum_{i=1}^{N_a} \hat{\alpha}_i \mathrm{A}_i, \tag{2.111}$$

$$vec(\hat{\mathrm{B}}) = (\mathrm{I}_{N_s^2 N_u^2} + \mathrm{KZM})vec(\mathrm{N}_4\mathrm{U}^{-1}) - \mathrm{KZ}\mathbf{c}^{us}. \tag{2.112}$$

2.4.3 Adding a constant term to the A matrix

If we want part of the matrix A fixed, *i.e.*

$$\mathrm{A} = \mathrm{A}_0 + \mathrm{A}_{var}, \tag{2.113}$$

where A_0 is fixed *a priori* and A_{var} is estimated from the data. Because the derivatives of G with respect to A and A_{var} are the same, we can rewrite the update equations 2.54 and 2.56 as

$$\hat{\mathrm{A}}_{var} = (\widetilde{\mathrm{N}}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})\mathrm{N}_2{}^{-1}, \text{ and} \tag{2.114}$$

$$\hat{\mathrm{B}} = (\widetilde{\mathrm{N}}_4 - \hat{\mathrm{A}}_{var}\mathrm{N}_5)\mathrm{U}^{-1}. \tag{2.115}$$

where we define

$$\widetilde{\mathrm{N}}_3 \triangleq \mathrm{N}_3 - \mathrm{A}_0\mathrm{N}_2\,, \text{and} \tag{2.116}$$

$$\widetilde{\mathrm{N}}_4 \triangleq \mathrm{N}_4 - \mathrm{A}_0\mathrm{N}_5\,. \tag{2.117}$$

With this reformulation, the update equations 2.72, 2.73,2.74, 2.75, and 2.76 turn into

$$\hat{\mathrm{A}}_{var} = (\widetilde{\mathrm{N}}_3 - \widetilde{\mathrm{N}}_4\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}})(\mathrm{N}_2 - \mathrm{N}_5\mathrm{U}^{-1}\mathrm{N}_5{}^{\mathsf{T}})^{-1}\,, \tag{2.118}$$

$$\hat{\mathrm{A}} = \mathrm{A}_0 + \hat{\mathrm{A}}_{var}\,, \tag{2.119}$$

$$\hat{\mathrm{B}} = (\widetilde{\mathrm{N}}_4 - \hat{\mathrm{A}}_{var}\mathrm{N}_5)\mathrm{U}^{-1}\,, \tag{2.120}$$

$$\hat{\mathrm{C}} = \mathrm{N}_6{}^{\mathsf{T}}\mathrm{N}_1^{-1}\,, \tag{2.121}$$

$$\begin{aligned}\hat{\mathrm{Q}} = \frac{1}{n}(&\mathrm{N}_1 + \hat{\mathrm{A}}\mathrm{N}_2\hat{\mathrm{A}}^{\mathsf{T}} - (\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})\hat{\mathrm{A}}^{\mathsf{T}} - \hat{\mathrm{A}}(\mathrm{N}_3 - \hat{\mathrm{B}}\mathrm{N}_5{}^{\mathsf{T}})^{\mathsf{T}} \\ &+\hat{\mathrm{B}}\mathrm{U}\hat{\mathrm{B}}^{\mathsf{T}} - \hat{\mathrm{B}}\widetilde{\mathrm{N}}_4^{\mathsf{T}} - \widetilde{\mathrm{N}}_4\hat{\mathrm{B}}^{\mathsf{T}})\,, \text{and}\end{aligned} \tag{2.122}$$

$$\hat{\mathrm{R}} = \frac{1}{n}(\mathrm{N}_7 - \hat{\mathrm{C}}\mathrm{N}_6 - (\hat{\mathrm{C}}\mathrm{N}_6)^{\mathsf{T}} + \hat{\mathrm{C}}\mathrm{N}_1\hat{\mathrm{C}}^{\mathsf{T}})\,. \tag{2.123}$$

If we want to include the fixed term for A in the constrained model, we rewrite 2.77 as

$$\mathrm{A} = \mathrm{A}_0 + \sum_{i=1}^{N_a} \alpha_i \mathrm{A}_i \tag{2.124}$$

we can interpret this as in 2.77 where $\alpha_{N_a+1} = 1$, $\mathrm{A}_{N_a+1} = \mathrm{A}_0$, and we don't update α_{N_a+1} in the M step. This way we wan rewrite all that has been shown in the last section up to equation 2.78 where we substitute c_i for $\tilde{c}_i$,

$$\tilde{c}_i \triangleq tr(\mathrm{Q}^{-1}(\mathrm{A}_i\widetilde{\mathrm{N}}_3^{\mathsf{T}} + \widetilde{\mathrm{N}}_3\mathrm{A}_i{}^{\mathsf{T}}))\,, \tag{2.125}$$

in equation 2.89 we substitute c_i^{us} for $\tilde{c}_i^{us}$,

$$\tilde{c}_i^{us} = 2tr(\mathrm{A}_i\widetilde{\mathrm{N}}_3^{\mathsf{T}})\,, \tag{2.126}$$

and equation 2.96 turns into

$$vec(\hat{\mathrm{B}}) + \mathrm{K}\hat{\alpha} = vec(\widetilde{\mathrm{N}}_4\mathrm{U}^{-1})\,, \tag{2.127}$$

and the final update for this case is

$$\begin{aligned}
\hat{\alpha} &= -\mathrm{T}^{us-1}\mathrm{M}(\mathrm{I}_{N_s^2N_u^2} + \mathrm{KZM})vec(\widetilde{\mathrm{N}}_4\mathrm{U}^{-1}) + \mathrm{Z}\tilde{\mathbf{c}}^{us}\,, && (2.128)\\
\hat{\mathrm{A}} &= \mathrm{A}_0 + \sum_{i=1}^{N_a}\hat{\alpha}_i\mathrm{A}_i\,, && (2.129)\\
vec(\hat{\mathrm{B}}) &= (\mathrm{I}_{N_s^2N_u^2} + \mathrm{KZM})vec(\widetilde{\mathrm{N}}_4\mathrm{U}^{-1}) - \mathrm{KZ}\tilde{\mathbf{c}}^{us}\,. && (2.130)
\end{aligned}$$

2.5 Accelerating the KS+EM algorithm

2.5.1 Distributed implementation

To reduce the time devoted to computations, we produced a distributed implementation of all the KS+EM routines. Even though this does not reduce the computational cost of the algorithm, it reduces the running time by distributing computations among different computing nodes, and it reduces the amount of memory needed per processor, making it possible to store all the required data structures in random access memory and hence reducing the running time.

The KS+EM algorithm described in this chapter was first implemented using Matlab (© The MathWorks, Inc.). All routines were then ported to C++ using the distributed memory linear algebra libraries ScaLAPACK (included in the Intel Math Kernel Libraries) for computations and the MPICH implementation of the message passing interface (MPI) for inter-node communication. ScaLAPACK uses the block-cyclic distribution to store dense matrices. While this makes it very efficient balancing the computational load of matrix multiplication, factorization and such across nodes, it also calls for data redistribution routines. All this book-keeping was encapsulated in a distributed matrix (pMatrix) class, that communicates with MPI to rearrange the data as needed by the current number of processors and their configuration. Then a class pKSmoother uses pMatrix objects to implement the KS+EM algorithm. To reduce the memory footprint of the algorithm, a pMatCollection class was generated to handle collections of distributed matrices that were backed up on disc and accessed only when needed. This was used to store the predicted, filtered and smoothed covariances the KS calls for. Two different versions of this class were generated, one using the low-level routine memmap to speed up disk access, and one using standard fopen/fread/fseek system calls. Even though the memory mapped version of the class did improve performance, the standard reading routines were favored at the end because memory mapping would exhaust the virtual memory of the computing nodes rather quickly. To deal with distributed reading/storing of matrices, we decided to use the local hard drives of each node and parallel reads in each, effectively increasing

the perceived bandwidth of the system as more computing nodes were recruited for the computations.

To find the optimal number of processors that should be used for a given SSM, one has to look at the speed-limiting operation. In KS+EM, matrix multiplication and matrix inversion are the most prevalent operations, and among them, the most costly are inversions and multiplications of square matrices whose size is the dimension of the hidden state. For different hidden state dimensions, the distributed routines will show different efficiency behavior when the number of computing nodes is increased, depending on whether the routine is communication-intensive or computation intensive. For computation-intensive routines such as matrix inversion or matrix multiplication, the speedup will be maximum when all computation nodes are working all the time; this means that small hidden state dimensions can show sub-unit speedup due to communication overhead, whereas large hidden state dimensions will make the speed up peak once all nodes are fully working. In communication-intensive routines, such as the *vec*() operator where all there is to do is data rearrangement, speedup will be typically sub-unit when we increase the number of processors, and will keep reducing when the number of nodes is increased because all the nodes will be doing is exchanging data through the communication network. Because all these behaviors are very dependent on the underlying processors in each node, their memory architecture, the libraries used for computations, and the communication routines and hardware, is difficult to predict what will be the optimal number of computing nodes for a given matrix size. We did some simulation studies where a given routine throughput was averaged across 1000 trials for matrix multiplication, matrix inversion and vectorization. The results can be seen in figure 2-2. For a matrix of size 1000, no more than 16 nodes are optimal, whereas for matrices of 2500 and 5000, increasing the number of nodes to 25 is recommended. We found the KS+EM algorithm to scale with efficiency above 90% for up to 64 computing nodes.

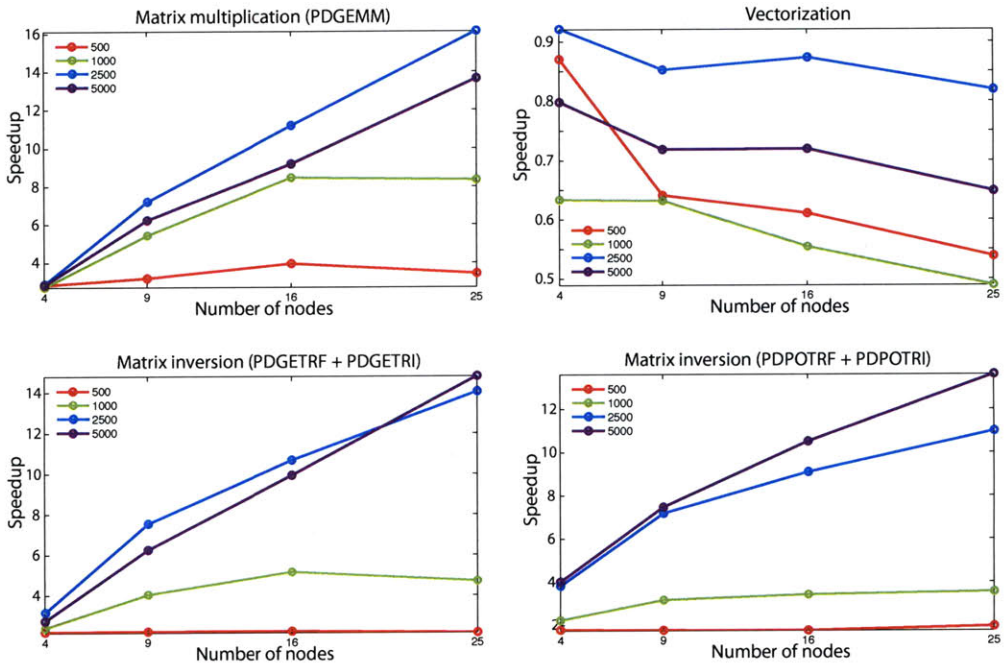


Figure 2-2: Speedup as a function of number of computing nodes for the distributed C++ implementation of matrix multiplication (upper left), matrix vectorization (upper right), matrix inversion (lower left), and positive definite matrix inversion (lower right). All but the vectorization are computation-intensive functions, and accordingly they show increasing speedups with increasing number of computing nodes. For matrix sized 1000 and under, the speedup stops growing after 16 nodes, but keeps growing with bigger matrices as discussed in the text.

2.5.2 Computational cost of the E step

In order to analyze the computational cost of the equations in Sections 2.3.3 and 2.3.4, we use the following costs per operation (in floating point operations, or flops) as reported in the documentation for the corresponding functions of the ScaLAPACK implementation in Intel Math Kernel Libraries:

- A matrix multiplication AB $\rightarrow$ C where A $\in \mathbb{R}^{m\times n}$, B $\in \mathbb{R}^{n\times p}$, and C $\in \mathbb{R}^{m\times p}$, has a cost of $\sim mnp$ multiplications. This could be lowered to $\sim 4n^{2.8}$ for $m = n = p$ when using the Schönhage-Strassen algorithm ([78]) which is unfortunately not implemented in ScaLAPACK PDGEMM function.

- A matrix inversion $A^{-1} \rightarrow$ B where A $\in \mathbb{R}^{n\times n}$ is a non-singular matrix has a cost of $\sim 2/3n^3$ flops for the factorization (MKL ScaLAPACK PDGETRF function) and a cost of $\sim 4/3n^3$ for the inversion of the factorized form (MKL ScaLAPACK PDGETRI function), for a total of $\sim 2n^3$ flops.

- A matrix inversion $A^{-1} \rightarrow$ B where A $\in \mathbb{R}^{n\times n}$ is a symmetric, positive-definite matrix has a cost of $\sim 1/3n^3$ flops for the triangulation (MKL ScaLAPACK PDPOTRF function) and a cost of $\sim 2/3n^3$ flops for the inversion of the triangulated form (ScaLAPACK PDPOTRI function), for a total of $\sim n^3$ flops.

With this in mind, and assuming that all the covariances involved in the formulae are non-singular (and hence, being covariance matrices, are positive definite), we can approximate the cost of each equation involved in the E step as:

- The computation of the filtered error covariance matrix takes

$$\begin{aligned}(N_s^2 + 2N_s^3) + (2N_s^2N_d + 2N_d^2N_s + N_d^2 + 2N_d^3) + (2N_dN_s^2 + N_s^2) = \\ 2N_s^3 + (2 + 4N_d)N_s^2 + 2N_d^2N_s + N_d^2 + N_d^3\end{aligned} \tag{2.131}$$

- The computation of the filtered state takes

$$(N_s^2) + (2N_dN_s + N_d + N_s) = N_s^2 + (2N_d + 1)N_s + N_d \tag{2.132}$$

- The computation of the smoothed error covariance matrix takes

$$(4N_s^3) + (2N_s^3 + 2N_s^2) = 6N_s^3 + 2N_s^2 \tag{2.133}$$

- The computation of the smoothed state takes

$$N_s^2 + 2N_s \tag{2.134}$$

We arranged the cost of each equation following the powers of N_s, the dimensionality of our hidden state, since that is the biggest number. From this we make two observations:

- The smoother is roughly three times as costly as the filter is, $2O(N_s^3)$ versus $6O(N_s^3)$.
- The computation of the covariances is $O(N_s^3)$, much more costly than the computation of the estimates which is $O(N_s^2)$.

2.5.3 Steady-state Kalman Smoother: accelerating the E step

Because the most computationally intensive part of the E step is the computation of the covariance matrices, we would want to skip that step whenever possible. Under certain conditions, it can be proven that a given state-space model will converge to a steady-state configuration in which the error covariances $\Lambda_{i|i}, \Lambda_{i|n}$ do not depend on i anymore, as explained in [36].

Determining whether a steady-state Kalman filter and smoother exists involves checking the stochastic observability and controllability of the SSM. The SSM will be observable if the hidden states can be completely recovered in absence of noise, and it will be controllable if we can drive the hidden states to a desired state from any arbitrary starting point using the stochastic control variables, which in our formulation correspond to the process noise. As [26], we will use the check for stochastic controllability and observability described in [39], which makes use of the eigenmodes of the

transition matrix A, its transpose A^T, the observation matrix C, and the stochastic control matrix which in our system correspond to $S = Q^{1/2}$. The system will be observable if and only if there is no eigenvector of A that is a null space vector of C, *i.e.*, if $A\mathbf{x}_{mode} = \lambda\mathbf{x}_{mode}, \lambda \neq 0$ and $C\mathbf{x}_{mode} = 0$, then the system is observable if only $\mathbf{x}_{mode} = 0$. For stochastic controllability, if $A^T\mathbf{x}_{mode} = \lambda\mathbf{x}_{mode}, \lambda \neq 0$ and $S^T\mathbf{x}_{mode} = 0$, then the system is controllable if only $\mathbf{x}_{mode} = 0$. For a time-invariant system as the ones described in this thesis, the Kalman filter and smoother will reach a steady-state covariance matrix if the system is detectable and controllable, as it is explained in [26].

After each run of the M-step in the KS+EM algorithm, we check for the convergency of the Kalman filter and smoother gains, that is the steady-state filtered, predicted and smoothed covariances, and stop updating them once they converge to reduce the computational load. Because in C++ we need to know how many covariances we will need to store, we use the doubling algorithm of [26] that provides the covariance at step $2t$ starting from that at step t, effectively cutting from linear to logarithmic the number of iterations to compute a given step covariance. This way we check for convergency first, allocate enough memory to store the results until convergency, and then proceed to compute the required covariances step by step.

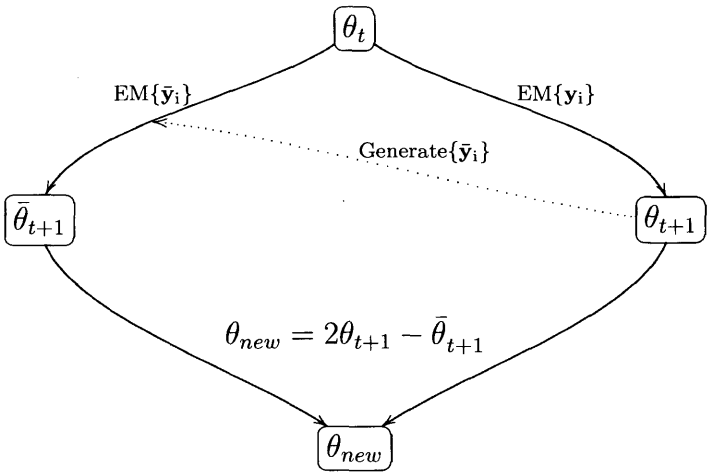
2.5.4 Ikeda acceleration: accelerating EM convergence

One of the main drawbacks of the EM algorithm is its slow convergency rate, as discussed in [50]. Alternative methods to maximize the likelihood as Newton-Rhapson methods guarantee quadratic convergency once the results are near the real values of the parameters to be estimated. This methods need to compute the observed information matrix to work and the Jacobian of the M step. In the KS+EM algorithm this information matrix becomes computationally costly to compute, since its number of variables grow with the square of the number of parameters of the model. The supplemented EM algorithm ([50], Chapter 4.5) compute the Jacobian matrix of the M step component-wise, using a sequence of "forced EMs".

Other acceleration strategies have been proposed in the literature. Louis' method

([ ]) works by updating the result of the EM iteration by using the Jacobian matrix, that has to be computed for each iteration. Again, this computation is costly and hence the method is also computationally expensive.

Ikeda proposes a method in [ ] that reuses the EM code to obtain an update strategy that is an approximation of the scoring method of Fisher, hence guaranteeing best than linear convergency. The method computes the EM updated parameters θ_{t+1} from the set of parameters θ_t and the data $\{\mathbf{y}_i\}$. Then generates a new dataset $\{\bar{\mathbf{y}}_i\}$ with the new parameters θ_{t+1}, and runs EM in this new dataset using θ_t as the initial parameters, obtaining an alternative estimate $\bar{\theta}_{t+1}$. The accelerated update is then a combination of the original update and the alternative update,



(2.135)

Although Ikeda's claims of threefold acceleration in convergency could not be reproduced in tests with the SSM KS+EM model, we were able to obtain two-fold speedups in almost every situation. This would not be particularly advantageous in the standard case, since the Ikeda scheme requires of two EM cycles per one new estimate. But in the case of the KS all the computational burden goes to the computation of the covariance matrices (see sec. 2.5.2), and these computations only depend in the parameters of the model θ and not in the data (see sec. 2.3.5). Now, since the two EM iterations that Ikeda requires are started with the same parameter values, we can reuse the covariances computed in the first EM iteration for the second one, effectively saving almost all the computation time in the second EM iteration so that the

two-fold convergency rate of the accelerated scheme cuts the computations required in almost half.

2.5.5 Numerical issues

Scaling

Because of rounding errors and the recursive nature of the KS+EM algorithm, matrices can become singular or non-positive definite easily after several iterations. To ameliorate the situation, it is often useful to work with a rescaled system. To do so, we will exploit the fact that the state-space formulation in 2.1-2.2 is non-unique. If we introduce the variable change

$$\mathbf{x}_i^* = \mathrm{M}\mathbf{x}_i\,, \tag{2.136}$$

$$\mathbf{y}_i^* = \mathrm{N}\mathbf{y}_i\,, \tag{2.137}$$

then an equivalent formulation of 2.1-2.2 is

$$\begin{aligned} \mathbf{x}_i^* &= \mathrm{A}^*\mathbf{x}_{i-1}^* + \mathrm{B}^*\mathbf{u}_i + \mathbf{v}_i^*\,, \\ \mathbf{y}_i^* &= \mathrm{C}^*\mathbf{x}_i^* + \mathbf{w}_i^*\,, \end{aligned} \tag{2.138}$$

where the new parameters relate to the old ones as

$$\mathrm{A}^* = \mathrm{MAM}^{-1}\,, \tag{2.139}$$

$$\mathrm{B}^* = \mathrm{MB}\,, \tag{2.140}$$

$$\mathrm{C}^* = \mathrm{NCM}^{-1}\,, \tag{2.141}$$

$$\mathbf{v}_i^* = \mathrm{M}\mathbf{v}_i\,, \text{and} \tag{2.142}$$

$$\mathbf{w}_i^* = \mathrm{N}\mathbf{w}_i\,. \tag{2.143}$$

$$\tag{2.144}$$

By choosing the scaling factors M and N carefully, we can whiten the data so that the covariance computations are always done on a scale as close as possible to the unity, if the condition number of the matrices involved is not too high. The scaling

factors used are

$$\mathrm{M} = \mathrm{Q}^{-1/2}\,, \text{ and} \tag{2.145}$$

$$\mathrm{N} = \mathrm{R}^{-1/2}\,. \tag{2.146}$$

Covariance inverses and determinant computation

Because both Q and R are big matrices, and the condition number of both can be quite big, the computation of the determinant of both can be quite challenging. We make use of the fact that both matrices are defined positive (because both are covariances) to compute the inverse and the determinant using the Cholesky decomposition,

$$\mathrm{X} = \mathrm{L}\mathrm{L}^{T}\,. \tag{2.147}$$

The inverse of the upper-triangular factor is computed, and from there is easy to compute the full inverse,

$$\mathrm{A}^{-1} = \mathrm{L}^{-T}\mathrm{L}^{-1}\,. \tag{2.148}$$

To compute the determinant, we use

$$log(|\mathrm{A}|) = 2log(|\mathrm{L}|) = 2\sum_{i} log(\mathrm{L}_{ii})\,. \tag{2.149}$$

Computing the logarithm directly increases the numerical stability for badly conditioned matrices.

2.6 Model selection

As discussed in this chapter's introduction, in order to apply KS+EM we will have to choose among different SSMs to describe our data. We need the chosen SSM to closely approximate the underlying distribution for the estimates of Θ to be reliable. Each of the candidate SSMs will be described by a set of hyper-parameters Γ, and the model selection problem is to select Γ, *i.e.*, choose the model that induces an underlying distribution closest to that which generated our data. Unfortunately, we do not know this underlying distribution. We can go around this by using an information criterion (IC), which gives an approximate bound on the Kullback-Leibler distance (KL distance, see [ ]) between the originating distribution and the estimated one. An IC employs the fact that the maximized log-likelihood of the data given the model, $ll(\Gamma)$, is a biased estimate of the desired KL distance. Then it computes a bias-correcting term that depends on the model structure and is asymptotically independent of the hidden states $\mathbf{x}$. Using the IC, we can select the hyper-parameters which explain the data without over-fitting. We run the KS+EM algorithm for different values of Γ, obtaining $ll(\Gamma)$, then approximate the KL distance for each value using the IC, and finally choose the value of Γ producing the lowest KL distance.

There are different ICs proposed in the literature, and none of them provides with a closed form for the SSM structures used in this thesis. All the information criteria tend to over-estimate the KL distance when the degrees of freedom of the model, k, get close to the number of data points available, n_t. In several cases in this thesis, we will be working in situations where $n_t \simeq 2k$.Finding an optimal IC for this scenario is still an open problem, although it has been proposed that the appropriate bias-correction term can be found with expensive Monte-Carlo simulations ([ ]). Instead of striving for the optimal solution we will will base our decision on the examination of several closed-form ICs:

- The Akaike information criterion (AIC, see [ ]),

$$AIC(\Gamma) = 2k - 2ll(\Gamma)\,. \tag{2.150}$$

AIC deviates considerably from the real KL distance when $k \simeq n_t$.

- The corrected Akaike information criterion (AICc, see [11]),

$$AICc(\Gamma) = AIC(\Gamma) + 2\frac{(k+1)(k+2)}{n_t - k - 2}, \tag{2.151}$$

which provides less biased KL distance estimates for $k \simeq n_t$ than AIC.

- The Bayesian information criterion (BIC),

$$BIC(\Gamma) = log(n_t)k - 2ll(\Gamma). \tag{2.152}$$

which provides with a consistent estimate, (AIC and AICc are efficient, see [7] for a description of BIC and discussion).

2.7 Summary

In this chapter we introduced the state-space model (SSM) framework that will be used in KronEM and StimEM to analyze MEG time-series. Sections 2.2, 2.3 and 2.4 reviewed the Kalman Smoother and the EM algorithm, which together solve the dynamic inverse problem in its SSM formulation. Because KS+EM have been previously avoided in the literature due to its high computational cost, in section 2.5 we provided a computational analysis of the algorithms, and building on this we developed an accelerated framework that reduces both the number of required computations and the time spent per computation by using Ikeda acceleration, steady-state covariance analysis, and a distributed-memory implementation. With this accelerated framework it will possible to analyze the high-dimensional state-spaces required for MEG analysis, as following chapters will show.

Finally, section 2.6 reviewed information criteria that let us decide among competing SSMs using the observed data. Together with KS+EM, information criteria provide us with all the tools needed to solve the estimation and model selection problems arising in SSM analysis of MEG time-series.

Chapter 3

KronEM: ML spatiotemporal covariance matrix estimation on resting-state MEG studies

Stochastic characterization of EEG/MEG spontaneous (or resting-state) brain activity is a challenging task that that can provide with insights in brain functioning. Here we introduce the KronEM algorithm, which provides maximum-likelihood estimates of the EEG/MEG spatiotemporal covariance matrix. KronEM parameterizes the spatiotemporal covariance as a sum of Kronecker products (KP) and models the data generation with a state-space model. KronEM automatically decides on the number of KPs needed and estimates their corresponding spatial and temporal characteristics using only the data. KronEM uses a Kalman smoother coupled with the expectation-maximization (EM) algorithm to estimate the model parameters, and an information criterion to select among candidate models. We apply KronEM to both synthetic data, single-channel experimental MEG data, and multichannel experimental MEG data. In synthetic data, KronEM correctly recovered the underlying spatiotemporal covariance structure, including the number of KPs; on single-channel MEG data, KronEM correctly identified physiological rhythms; on multichannel data, KronEM produced topographic frequency maps better resolved than the corresponding multitaper frequency spectrum estimates.

3.1 Introduction

Background noise is present in all EEG and MEG measurements. This background noise is due to instrumental noise in the sensors and environmental disturbances (measurement noise), and spontaneous brain activity unrelated to the given experiment (brain noise). It is of interest to properly characterize background noise for two reasons: employing information about measurement and brain noise leads to source localization algorithms with smaller variances of the estimated parameters (*c.f.* [62, 80, 36, 29]), and characterization of brain noise provides insights to brain physiology (*c.f.* [7, 12, 11, 53]).

Background noise characterization is challenging due to the dimensionality of the problem: for a data set with n_t time samples and n_c channels, the spatiotemporal covariance to be estimated has $(n_t n_c)(n_t n_c + 1)/2$ free parameters, clearly too high for the data available in a typical EEG/MEG experiment. Furthermore, one cannot guarantee across-epoch stationarity when the experiment becomes long. As a result, a reduced number of samples can be used for estimation, and the estimates may be unreliable. On top of that, the computational and storage needs of the estimation process soon would become prohibitive even in modern computers.

To overcome this dimensionality and computational problems, [69] assumed a Kronecker product (KP) structure for the background noise covariance matrix, which expresses the covariance as the KP of a temporal and a spatial term. This reformulation cuts down the number of free parameters to $\left(n_t(n_t + 1) + n_c(n_c + 1)\right)/2$, much lower than that of the full problem, in addition to be less computationally demanding due to the properties of the Kronecker product algebra (see [56] for a review). Further work by [12] assumed the temporal component of the covariance to be stationary, cutting down the number of parameters to $n_t + n_c(n_c + 1)/2$, and proposed a maximum-likelihood (ML) approach to estimate the parameters. However, the single KP structure could not capture the brain noise structure when spatially segregated brain areas present with rhythms at different frequency bands, a plausible scenario due to brain physiology. In this case, each rhythm should be accounted for by a

different KP structure. Such situation was accounted for in the work of [7], where the background noise covariance was modeled as a sum of KPs (KP-sum), each KP can accounting for a different rhythm. The approach showed promising results for 2 KPs, with the first KP resembling α-rhythm activity and the second one being temporally white and spatially broad, characteristics expected of measurement noise. However, the estimation method proposed was mathematically complicated; only results for two KPs were presented. In order to overcome the difficulties of the KP-sum model, [72] proposed a multi-pair KP approximation using either orthogonal or independent spatial bases, and provide a computationally feasible way to estimate the model components for an arbitrary number of KPs. Furthermore, [7] nor [72] provided a principled way for estimating the number of KP components.

In this chapter we introduce KronEM, an algorithm that provides a robust, self-contained method for background noise modeling and estimation. KronEM models background noise generation using a state-space model (SSM) that naturally imposes a KP structure on its spatiotemporal covariance. This SSM approach reduces the number of parameters to be estimated and allows us to apply the estimation techniques already developed for the SSM framework. We use a Kalman smoother (KS, [18, 2]) coupled with the expectation-maximization (EM) algorithm (as in [52]) to recover the maximum *a posteriori* (MAP) estimates for the SSM parameters, which fully characterize the covariance matrix. In contrast with previous methods ([62, 12, 7, 72]), the KronEM approach warrants convergency, allows for arbitrary number of KP components, and offers tools for model selection to decide how many KP components are needed for a parsimonious explanation of the data. Because the computational cost of the algorithm is higher than in single-KP or multi-pair-KP approximations, our implementation reduces the computation time exploiting the KS steady-state regime to reduce the required number computations per KS run, and using the acceleration scheme described in [15] to reduce the number of iterations EM takes to converge.

The rest of the chapter is organized as follows: the Methods section introduces the SSM at the heart of the KronEM algorithm, shows its relationship with previous KP-based approaches, reviews the KS+EM algorithm used to estimate the model

parameters along with the modifications decreasing its running time, explains how KronEM chooses a given model, including the number of KPs needed to explain the data, and concludes describing the experimental data used in the example analysis. We then describe the results of applying KronEM to synthetic data, to single-channel experimental MEG data, and to multi-channel MEG data. We conclude with a discussion and describe possible future research directions.

3.2 Methods

3.2.1 Spatiotemporal covariance structure

Ideally we would want to isolate brain noise and measurement noise to characterize them separately. Unfortunately, since brain noise is to be observed through the sensors, it will always be accompanied by measurement noise. KronEM models background noise using an additive model,

$$\mathbf{y}(t) = \mathbf{y}^b(t) + \mathbf{n}^m(t)\,, \tag{3.1}$$

where $\mathbf{y}(t) \in \mathbb{R}^{n_s \times 1}$ contains the EEG and/or MEG measurements at time t, $\mathbf{y}^b(t)$ is the component of $\mathbf{y}(t)$ due to spontaneous brain activity (brain noise), and $\mathbf{n}^m(t)$ is the measurement noise. As in former KP work ([62, 12, 7, 72]), we assume that the covariance of $\mathbf{y}^b(t)$ changes with time, *i.e.* the process $\mathbf{y}^b(t)$ is heteroskedastic, and model the spatiotemporal covariance among sensors i and j at times m and n as the product of a temporal and a spatial contribution,

$$\mathrm{E}\{\mathbf{y}^b_i(m)\mathbf{y}^b_j(n)\} = \mathrm{T}_{m,n}\mathrm{X}_{i,j}\,, \tag{3.2}$$

where $\mathrm{X} \in \mathbb{R}^{n_s \times n_s}$ is the spatial term of the covariance, and $\mathrm{T} \in \mathbb{R}^{n_t \times n_t}$ is the temporal term. The spatiotemporal covariance of the brain noise can then be expressed as a Kronecker product,

$$\mathbf{y}^b = \left(\mathbf{y}^b(1)^\mathsf{T}, \ldots, \mathbf{y}^b(n_t)^\mathsf{T}\right)^\mathsf{T} \tag{3.3}$$

$$\mathrm{E}\{\mathbf{y}^b\mathbf{y}^{b\mathsf{T}}\} = \begin{pmatrix} \mathrm{T}_{1,1}\mathrm{X}, & \cdots & \mathrm{T}_{1,n_t}\mathrm{X} \\ \vdots & \ddots & \vdots \\ \mathrm{T}_{N_\beta,1}\mathrm{X}, & \cdots & \mathrm{T}_{n_t,n_t}\mathrm{X} \end{pmatrix} = \mathrm{T} \otimes \mathrm{X}\,. \tag{3.4}$$

We will assume measurement noise $\mathbf{n}^m(t)$ to be temporally white. MEG measurement noise can be readily recorded in the magnetically shielded room void of a subject ("empty-room" noise), and it can be shown to be temporally white with

proper artifact rejection. In EEG, the dependency of noise in the skin-electrode interface precludes its direct observation, but KronEM will assume it temporally white as well.

3.2.2 KronEM model for the single KP case

Here we introduce the KronEM algorithm, which models the background noise data as a linear transformation of a n_β-th order multivariate autorregresive (MVAR) model. For conceptual clarity, we will focus in the single KP case first, and will extend it to multiple KPs in the next section.

In the single-KP model, the brain noise $\mathbf{y}^b(t)$ in Eq. 3.1 is generated as the product of a constant spatial component $\mathrm{L} \in \mathbb{R}^{n_c \times n_s}$ and a temporal component $\mathbf{h}(t) \in \mathbb{R}^{n_s \times 1}$, modeled as a n_β-th order MVAR,

$$\mathbf{y}^b(t) = \mathrm{L}\mathbf{h}(t) \text{ and} \tag{3.5}$$

$$\mathbf{h}(t) = \sum_{k=1}^{n_\beta} \beta_k \mathbf{h}(t-k) + \mathbf{z}(t)\,, \tag{3.6}$$

where $\mathbf{z}(t) \sim \mathcal{N}(\mathbf{0}, \mathrm{I}_{n_s})$ and white, where $\mathrm{I}_{n_s} \in \mathbb{R}^{n_s \times n_s}$ is the identity matrix. The MVAR order, n_β, and its dimension, n_s, are hyper-parameters defining our model and should be selected to fit the data as discussed in Section 3.2.5..

If we now compute the spatiotemporal covariance of the full measurement data $\mathbf{y}$, we get the desired KP structure of Eq. 3.4, where

$$\mathrm{X} = \mathrm{L}\mathrm{L}^T \tag{3.7}$$

with $rank(\mathrm{X}) \leq min(n_s, n_c)$, and

$$\mathrm{T}_{m,n} = \sigma^2_{m,n} = \mathrm{E}\{h_1(m)h_1(n)\}. \tag{3.8}$$

Because the hidden process $\mathbf{h}(t)$ is MVAR, the matrix T is Toeplitz and symmetric,

$$\mathrm{T}_{m,n} = g(|m-n|)\,, \tag{3.9}$$

so that the spatiotemporal covariance structure is identical to that reported in [6], where it explained $\sim$ 80% of the background noise in MEG measurements. The function $g(t)$ can be computed applying the $\mathcal{Z}$-transform to Eq. 3.6 to obtain

$$g(t) = \mathcal{Z}^{-1}\left\{\frac{1}{1-\sum_{k=1}^{n_\beta}\beta_k z^{-k}}\right\}, \tag{3.10}$$

where $\mathcal{Z}^{-1}$ is the inverse unilateral $\mathcal{Z}$-transform. The frequency response of $g(t)$ can be computed using the Fourier transform, $\mathcal{F}\{g(t)\}$.

In the case $n_\beta = n_t$, Eq. 3.10 does not impose any constraint in the shape of $g(t)$ besides $g(0) = 1$. This is not a problem, since $g(0)$ is a covariance and has to be positive, and any further scaling could be accounted for in L. When $n_\beta < n_t$, we get a more compact description of T but we constrain the functions $g(t)$ that can be described this way. Finding an appropriate n_β is part of the model selection problem discussed in Section 3.2.5.

The key observation for KronEM is that Eqs. 3.1, 3.5, and 3.6 are equivalent to the state-space model

$$\mathbf{x}(t+1) = \mathrm{A}\mathbf{x}(t) + \mathbf{w}(t)\,, \tag{3.11}$$

$$\mathbf{y}(t) = \mathrm{L}\mathbf{x}(t) + \mathbf{n}^m(t); \tag{3.12}$$

where

$$\mathbf{x}(t)^\mathsf{T} = \left(\mathbf{h}(t)^\mathsf{T}, \mathbf{h}(t-1)^\mathsf{T}, \cdots, \mathbf{h}(t-n_\beta+1)^\mathsf{T}\right) \tag{3.13}$$

describes the hidden state,

$$\mathrm{A} = \begin{pmatrix} \beta_1, & \beta_2, & \cdots & \beta_{n_\beta} \\ 1, & 0, & \cdots & 0 \\ 0, & 1, & \cdots & 0 \\ & & \ddots & \end{pmatrix} \otimes \mathrm{I}_{n_s} \tag{3.14}$$

incorporates past values of $\mathbf{h}(t)$ in $\mathbf{h}(t+1)$ and propagates past values of $\mathbf{h}(t)$,

$$\mathbf{w}(t)^\mathsf{T} = \left(\mathbf{z}(t+1)^\mathsf{T}, {\mathbf{0}_{n_s}}^\mathsf{T}, {\mathbf{0}_{n_s}}^\mathsf{T}, \cdots\right) \tag{3.15}$$

incorporates the noise $\mathbf{n}(t+1)$ in the current value of $\tau(t+1)$, $\mathbf{0}_{n_s} \in \mathbb{R}^{n_s \times 1}$ is a zero vector, and $\mathbf{n}^m(t) \in \mathbb{R}^{n_c \times 1}$ is a white, Gaussian process with covariance R that models the measurement noise contribution to $\mathbf{y}(t)$.

3.2.3 KronEM model for the KP-sum case

As discussed before, [7, 79] extended the single-KP model to include different spatiotemporal components that can account for spatially segregated rhythms present in the brain noise. In this case, Eq. 3.4 turns into

$$\mathrm{E}\{\mathbf{y}^b {\mathbf{y}^b}^\mathsf{T}\} = \sum_{1=1}^{n_{KP}} \mathrm{T}^{(i)} \otimes \mathrm{X}^{(i)}\,. \tag{3.16}$$

The extension of the KronEM algorithm to such cases is trivial using the SSM formulation of Eq. 3.11 and Eq. 3.12 with several hidden processes,

$$\mathbf{x}^{(i)}(t+1) = \mathrm{A}^{(i)}\mathbf{x}^{(i)}(t) + \mathbf{w}^{(i)}(t)\,, i = 1, \ldots, n_{KP} \tag{3.17}$$

$$\mathbf{y}(t) = \sum_{i=1}^{n_{KP}} \mathrm{L}^{(i)}\mathbf{x}^{(i)}(t) + \mathbf{n}^m(t); \tag{3.18}$$

these equations can be combined in a single SSM as

$$\mathbf{x}_\Sigma(t+1) = \mathrm{A}_\Sigma \mathbf{x}_\Sigma(t) + \mathbf{w}_\Sigma(t)\,, \tag{3.19}$$

$$\mathbf{y}(t) = \mathrm{L}_\Sigma \mathbf{x}_\Sigma(t) + \mathbf{n}^m(t); \tag{3.20}$$

where the parameters for the combined SSM can be obtained from those of the single-KP KronEM models for the pairs $\{\mathrm{X}_i, \mathrm{T_i}\}$ as

$$\mathbf{x}_\Sigma(t)^\mathsf{T} = \left(\mathbf{x}^{(1)}(t)^\mathsf{T}, \mathbf{x}^{(2)}(t)^\mathsf{T}, \cdots, \mathbf{x}^{(n_{KP})^\mathsf{T}}(t)\right), \tag{3.21}$$

$$\mathrm{A}_\Sigma = \begin{pmatrix} \mathrm{A}^{(1)}, & 0_{n_s}, & \cdots & 0_{n_s} \\ 0_{n_s}, & \mathrm{A}^{(2)}, & 0_{n_s}, & \cdots \quad 0_{n_s} \\ & & \ddots & \\ 0_{n_s}, & & \cdots & 0_{n_s}, \quad \mathrm{A}^{(n_{KP})} \end{pmatrix}, \tag{3.22}$$

$$\mathbf{w}_\Sigma(t)^\mathsf{T} = \left(\mathbf{w}^{(1)}(t)^\mathsf{T}, \mathbf{w}^{(2)}(t)^\mathsf{T}, \cdots, \mathbf{w}^{(n_{KP})}(t)^\mathsf{T}\right), \tag{3.23}$$

$$\mathrm{L}_\Sigma = \left(\mathrm{L}^{(1)}, \mathrm{L}^{(2)}, \cdots, \mathrm{L}^{(n_{KP})}\right). \tag{3.24}$$

$$\tag{3.25}$$

The generality and power of the SSM framework is now evident: the estimation procedure for the parameters of the covariance to be presented below will be identical for the single-KP or KP-sum case, because the structure of the SSM described by Eq. 3.19 and Eq. 3.20 remains unchanged after incorporating more than one KP term in the covariance.

3.2.4 Parameter estimation

As evidenced by Eqs. 3.4, 3.7 - 3.10, the estimation of the spatiotemporal covariance in the KronEM algorithm is equivalent to recovering the temporal parameters $\beta^{(i)}$ and the spatial parameters $\mathrm{L}^{(i)}$ for $i = 1, \ldots, n_{KP}$. The measurement noise covariance, R, is also estimated from the data. The parameters are then $\Theta = \{\mathrm{R}, \beta^{(1)}, \ldots, \beta^{(n_{KP})}, \mathrm{L}^{(1)}, \ldots, \mathrm{L}^{(n_{KP})}\}$. This assumes that we know the num-

ber of KPs to be estimated, n_{KP}, the dimensionality of the hidden processes, n_s, and the order of the AR model, n_β, which is not the case. The estimation of the vector of hyper-parameters $\Gamma = \{n_s, n_\beta, n_{KP}\}$ is discussed in Section 3.2.5.

We use the expectation-maximization (EM) algorithm to recover the parameters. Because of the structure of our SSM, the E step uses a Kalman smoother (KS). The filter was originally described in [18] while [2] presented a particularly elegant derivation of the smoother. The combined application of EM and KS was first introduced and thoroughly described in the classic work of [32].

In the implementation of KronEM we introduced a couple of modifications to both the E and M steps to reduce computation time and guarantee stability. In each run of the KS (the E step) we check for observability and stochastic controllability using the methods described by [30], to ensure that the KS will behave properly. We then employ the doubling algorithm as explained in [26] to obtain the steady-state filter and smoother covariance. If the steady-state filter and smoother covariance exist, we use the steady-state values for both once steady-state is reached, reducing the time spent in computations. The M-step is similar to that described in [32], but the $\beta^{(i)}$ update is an original contribution of this thesis. Because the algorithm converges slowly, we make use of an acceleration scheme proposed by [13] that uses a second M update and typically cuts the number of EM iterations required for convergence to half..

3.2.5 Model selection

As discussed above, the hyper-parameters $\Gamma = \{n_s, n_\beta, n_{KP}\}$ define our model, and have to be selected outside the KS+EM algorithm. These parameters define our model, and we want a model that closely approximates the underlying distribution, so that the estimates for Θ we recover with KS+EM are close to its real values. Unfortunately, we do not know this underlying distribution. We can go around this by using an information criterion (IC), which gives an approximate bound on the Kullback-Leibler distance (KL distance, see [19]) between the originating distribution and the estimated one. An IC employs the fact that the maximized log-likelihood

of the data given the model, $ll(\Gamma)$, is a biased estimate of the desired KL distance. Then it computes a bias-correcting term that depends on the model structure and is asymptotically independent of the hidden states $\mathbf{x}$. Using the IC, we can select the hyper-parameters which explain the data without over-fitting. We run the KS+EM algorithm for different values of Γ, obtaining $ll(\Gamma)$, then approximate the KL distance for each value using the IC, and finally choose the value of Γ producing the lowest KL distance.

There are different ICs proposed in the literature, and none of them provides with a closed form for SSM structures as the one KronEM uses. All the information criteria tend to over-estimate the KL distance when the degress of freedom of the model, k, get close to the number of data points available, n_t. For KronEM,

$$k = n_{KP}(n_\beta + n_c n_s) + \frac{1}{2} n_c(n_c + 1)\,, \tag{3.26}$$

and we will be working in situations where $n_t \simeq 2k$.Finding an optimal IC for this scenario is still an open problem, although it has been proposed that the appropriate bias-correction term can be found with expensive Monte-Carlo simulations ([ ]). Instead of striving for the optimal solution we will will base our decision on the examination of several closed-form ICs:

- The Akaike information criterion (AIC, see [ ]),

$$AIC(\Gamma) = 2k - 2ll(\Gamma)\,. \tag{3.27}$$

  AIC deviates considerably from the real KL distance when $k \simeq n_t$.

- The corrected Akaike information criterion (AICc, see [ ]),

$$AICc(\Gamma) = AIC(\Gamma) + 2\frac{(k+1)(k+2)}{n_t - k - 2}\,, \tag{3.28}$$

  which provides less biased KL distance estimates for $k \simeq n_t$ than AIC.

- The Bayesian information criterion (BIC),

$$BIC(\Gamma) = log(n_t)k - 2ll(\Gamma)\,. \tag{3.29}$$

  which provides with a consistent estimate, (AIC and AICc are efficient, see [79] for a description of BIC and discussion).

For the sake of simplicity, we will fix $n_\beta = 2$ for the application examples, so that the corresponding AR processes will have a unimodal power spectral density. This allows us to compare KronEM with multi-taper spectrum estimates. We also fix $n_s = 1$, so that each $\mathrm{X}^{(i)}$ is rank-one (see Eq. 3.7). Because of these choices, the model selection part of KronEM reduces to searching for the value of n_{KP} minimizing the IC, $\hat{n}_{KP}$.

3.2.6 Data acquisition and preprocessing

The MEG data were acquired with a 306-channel MEG system (Vectorview, Elekta-Neuromag, Helsinki, Finland). The coils of the MEG channels are arranged in a hemispherical mosaic with 102 locations. At each location, a magnetometer measures the normal component of the magnetic field, while two planar gradiometers measure the two perpendicular off-diagonal gradients of the same component (see [86]). Bandwidth was set at 0.1 to 334 Hz, and data were digitized at 1798 samples/sec. The relative position of the head and MEG array was determined at the beginning of each acquisition by feeding currents to head-position indicator coils and by locating them on the basis of the magnetic fields measured with the MEG sensors ([87]). Subjects were instructed to close their eyes and recordings were acquired for several minutes. Continuous data segments where rhythmic activity was prominent were selected for analysis.

Signal-space projectors (SSP, [85]) were applied to the data to suppress the ECG and eye-movement related artifacts. These SSP used the main PCA directions of the data only at the times when a given artifact was maximal, and the source of the artifact was identified using the concurrently acquired EKG and EOG of the

subject. SSP removes a n_{SSP}-dimensional noise subspace from the data, and therefore is a rank-reducing operation which makes the covariance singular. This would cause trouble in the KS+EM algorithm. To ensure a full-rank covariance we excluded from subsequent analysis the n_{SSP} channels whose exclusion least reduced the power in the remaining signal. Data were then band-pass filtered from 1 to 40 Hz, and downsampled 11 times so that the Nyquist frequency (81.72 Hz) was twice that of the maximum frequency component in the signal.

3.3 Results

3.3.1 Simulations

To validate KronEM, we generated 5 data sets using Eqs. 3.19 and 3.20 with $n_c = 10$, $n_s = 1$, $n_t = 1000$, $n_\beta = 2$, and $n_{KP} = 1, 2, \ldots, 5$. Under this conditions, the equations 3.19 and 3.20 can be written as

$$x^{(i)}(t) = \beta_1^{(i)} x^{(i)}(t-1) + \beta_2^{(i)} x^{(i)}(t-2) + w^{(i)}(t), i = 1, \ldots, n_{KP} \quad (3.30)$$

$$\mathbf{y}(t) = \sum_{i=1}^{n_{KP}} \mathbf{l}^{(i)} x^{(i)}(t) + \mathbf{n}^m(t) \quad (3.31)$$

where $\mathbf{l}^{(i)} \in \mathbb{R}^{n_c \times 1}$ is the spatial component that process $x^{(i)}(t)$ modulates, $w^{(i)}(t) \sim \mathcal{N}(0,1)$ and white, and $\mathbf{n}^m(t) \sim \mathcal{N}(\mathbf{0}, \mathrm{R})$ and white. The $\mathbf{l}^{(i)}$ were sampled from a $\mathcal{N}(0,1)$ distribution, the pairs $(\beta_1^{(i)}, \beta_2^{(i)})$ generate a sinusoid of discrete frequency $f^{(i)} \in [0.1\pi, 0.4\pi]$, and R is a positive definite random matrix that is scaled for $SNR = 1$, with

$$SNR = \frac{\frac{1}{n_t} trace(\sum_{t=1}^{n_t} (\mathbf{y}(t) - \mathbf{v}(t))(\mathbf{y}(t) - \mathbf{v}(t))^{\mathsf{T}})}{trace(\mathrm{R})} \quad (3.32)$$

In each data set we run the estimation part of KronEM using different candidate values for the number of hidden processes, $\dot{n}_{KP} = 1, 2, \ldots, 6$, obtaining a maximized log-likelihood and values for each IC. Figure 3-1 show plots for AIC, AICc and BIC for the data sets correponding to $n_{KP} = 2, 3, 4, 5$ versus model order k (see Eq. 3.26) for each candidate value $\dot{n}_{KP}$. The discontinuous grey line indicates the true value of k. BIC is the most reliable estimator of model order: In all cases, using BIC leads to successful model selection, $\hat{n}_{KP} = n_{KP}$, and the minimum reached by BIC is much sharper. Using AIC or AICc would lead to $\hat{n}_{KP} > n_{KP}$ for $n_{KP} = 5$.

The results for the data set with $n_{KP} = 3$ are then analyzed to illustrative the estimation part of KronEM. Figure 3-2 shows the progression of the log-likelihood with EM iterations, with and without Ikeda acceleration. Figure 3-2 also includes the

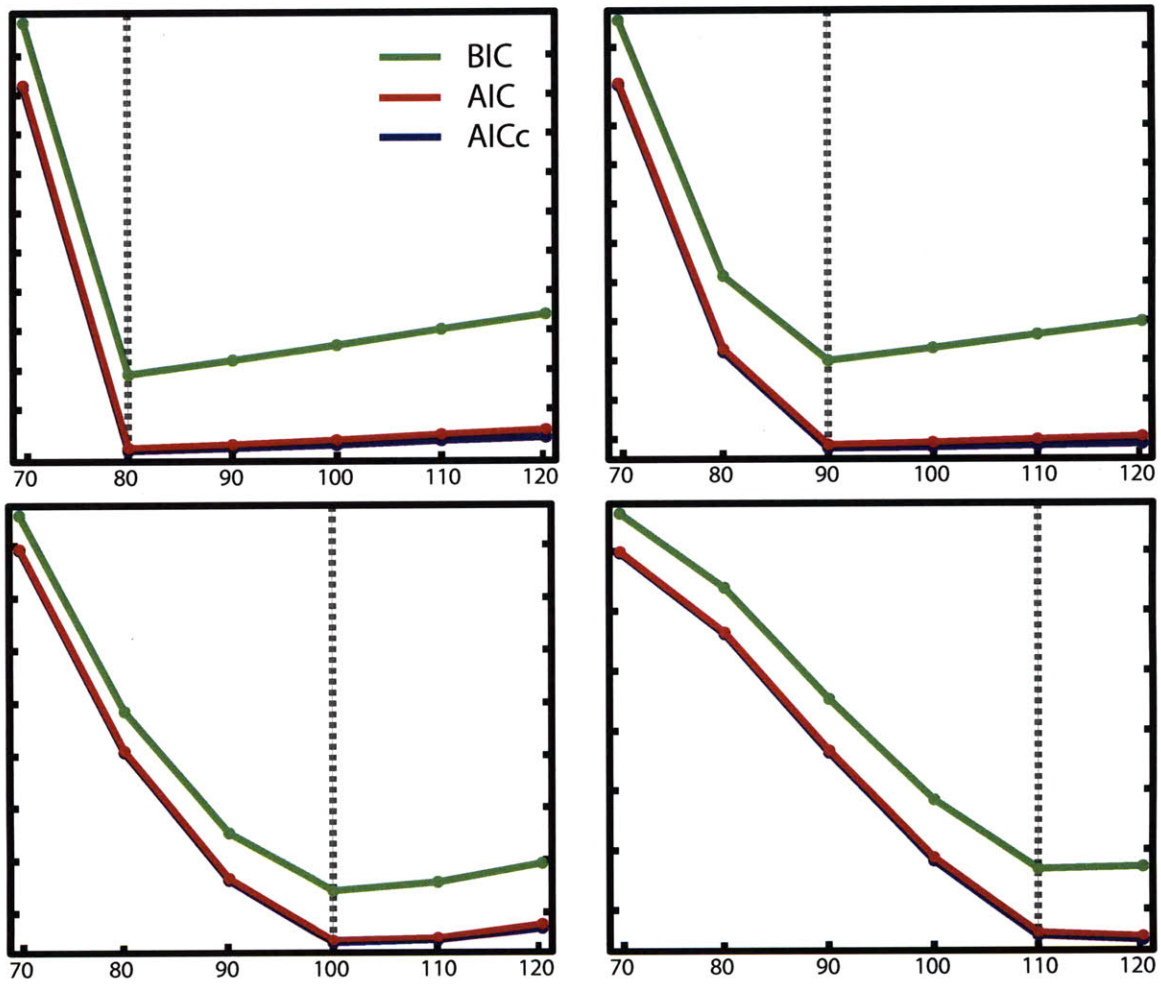


Figure 3-1: AIC, AICc and BIC values for data sets with $n_{KP} = 2, 3, 4, 5$, plotted versus model order k corresponding to candidate $\dot{n}_{KP} = 1, 2, \ldots, 6$. The correct model order is marked by the discontinuous grey line In all cases, BIC reaches the minimum in the correct model order, and produces a sharper minimum. Both AIC and AICc overestimate model order for $n_{KP} = 5$ (bottom left inset).

progression of the mode of the frequency response, plotted along with its full-width half-maximum as an area plot for both the accelerated and non-accelerated scheme. It can be observed that Ikeda's method reduced the number of iterations needed to reach a maximum for the log-likelihood, and that the accelerated version produced the same frequency distribution. The frequencies became stable many iterations before the log-likelihood did. Figure 3-3 shows the recovered $\mathbf{l}^{(i)}$ and its corresponding frequency responses, the Fourier transforms of $g^{(i)}(t)$ (see Eq. 3.10). The model order was correctly obtained minimizing the BIC. Both the spatial and temporal components were well recovered. Because each AR process $x_i(t)$ is order 2, the corresponding frequency responses are unimodal, and the real and KronEM-estimated peak locations showed good agreement.

3.3.2 Experimental data

Single-channel data

In this experiment, one left occipital magnetometer showing α-rhythm activity was selected from the data set and studied using KronEM to illustrate the frequency decomposition in experimental data for $n_\beta = 2$. A total of 35.5 seconds of the trace were analyzed for a total of 5787 samples. Both AIC, AICc and BIC suggested $\hat{n}_{KP} = 14$, and the KS+EM part of KronEM converged in 100 iterations. The resulting hidden process contributions were analyzed using multitaper spectrum estimation to illustrate the way KronEM decomposed the signal in different components ($l^{(i)}x^{(i)}(t)$, see Eq. 3.18). The top inset of Figure 3-4 shows the nearly identical spectra of both the estimated and original signal. The α-rhythm could be seen as the narrow-band activity around 10Hz. On the same plot we show the multitaper spectra corresponding to each individual component: the first component explained all activity below α, and the α-band activity was split in two components, showing good agreement with the original signal spectra. When plotted in the time domain (bottom inset in Figure 3-4) the oscillatory nature of each component is seen clearly.

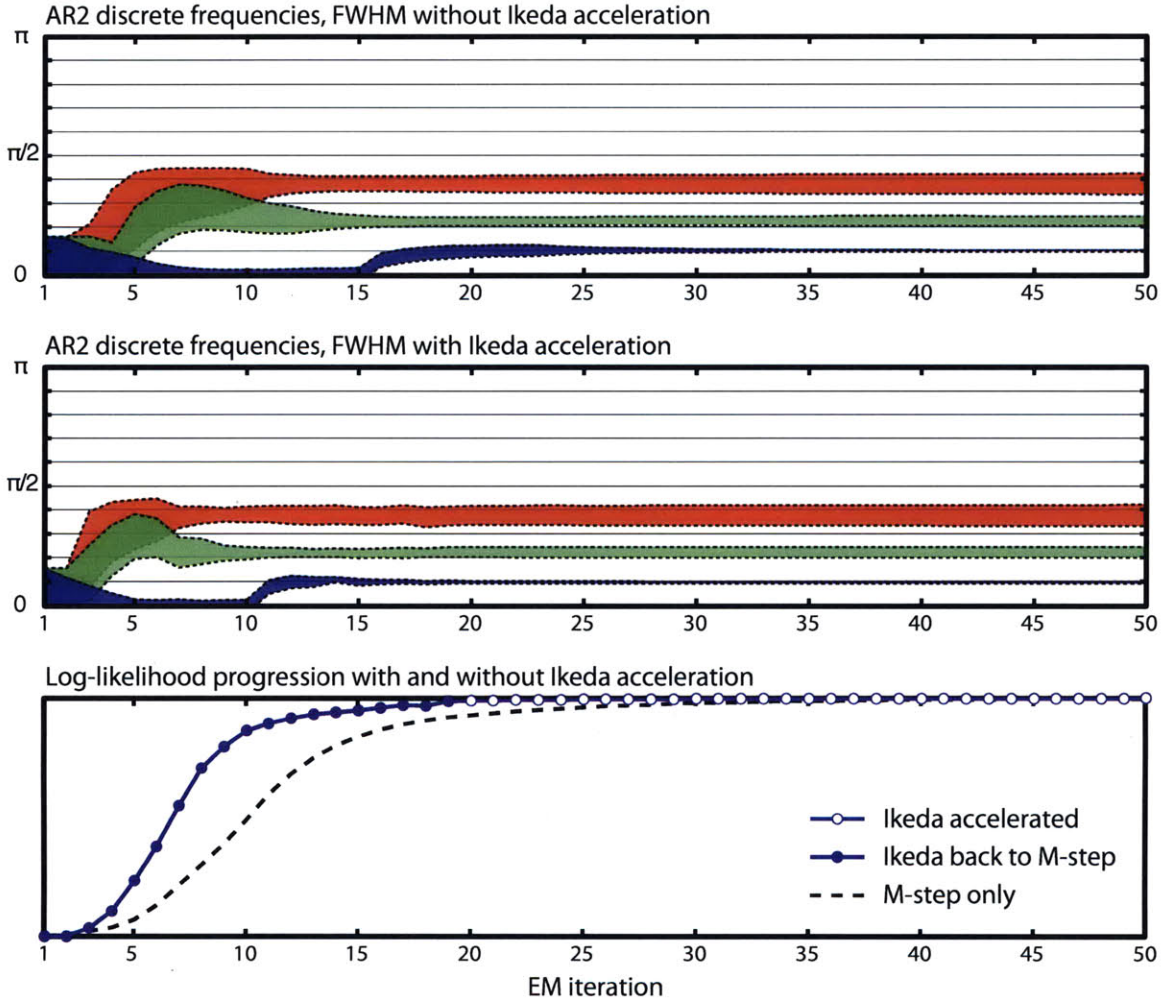


Figure 3-2: Evolution of AR2 coefficients and likelihood for both accelerated and non-accelerated EM, with $n_{KP} = 3$. In top two panels frequency evolution is tracked with center and FWHM of the frequency response induced by the AR coefficients at each iteration. Note that both standard EM (top inset) and accelerated EM (center inset) converge to the same frequencies. The log-likelihood (bottom inset) increases faster for the accelerated version, and frequencies converge before log-likelihood does.

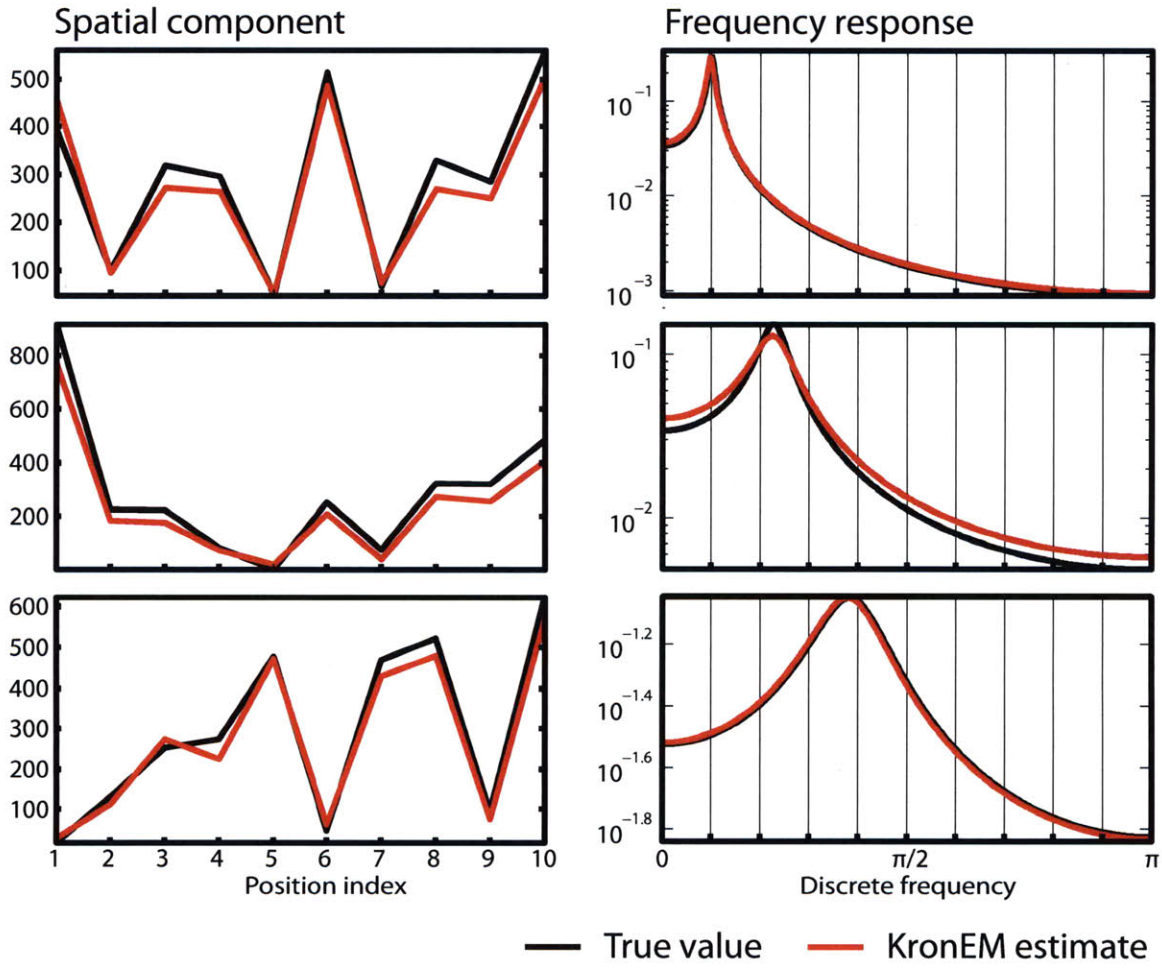


Figure 3-3: KronEM estimates for the data set $n_{KP} = \hat{n}_{KP} = 3$. Plots show real (black) and estimated (red) values for $\mathbf{b}^{(i)}$ (left) and frequency response $\mathcal{F}\{g^{(i)}(t)\}$ for the three KP terms (top to bottom).

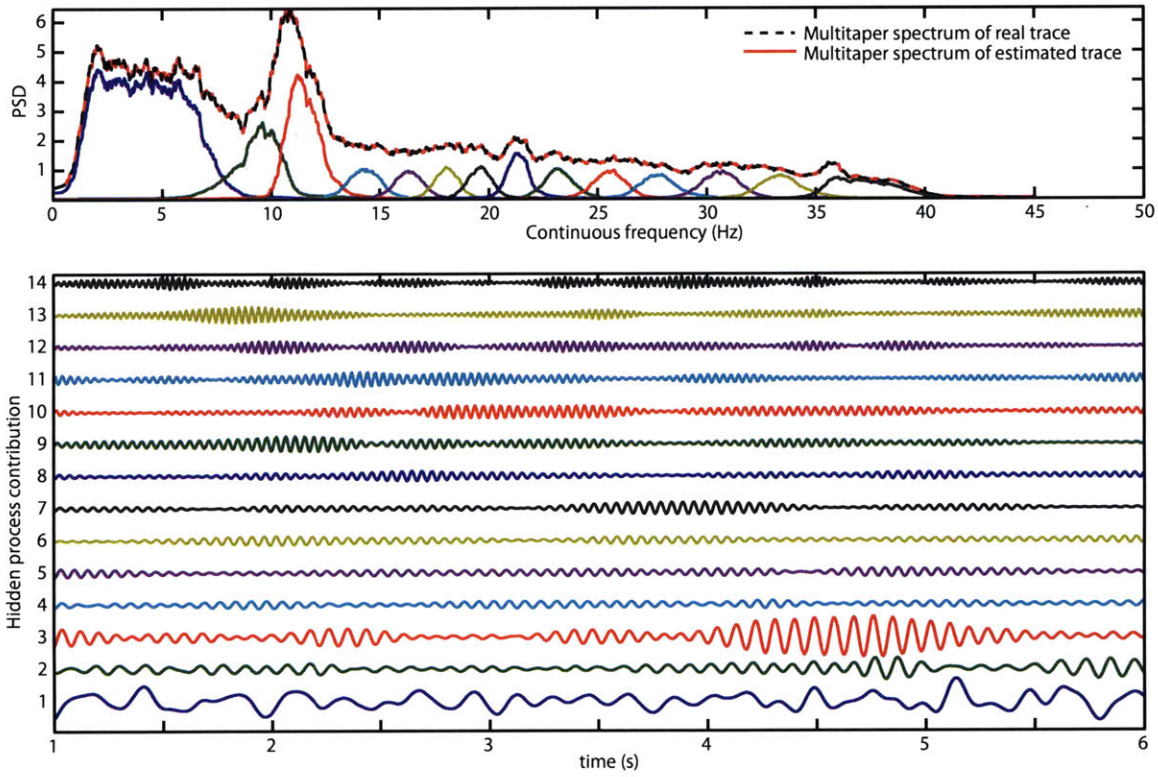


Figure 3-4: KronEM estimates for a single MEG channel with α-rhythm activity. Top inset shows the multitaper frequency plots for the estimated signal, $\sum_i^{n_{KP}} l^{(i)} x^{(i)}(t)$ (continuous red), the original signal (dashed black), and each of the hidden process contributions, $l^{(i)} x^{(i)}(t)$ (different colors). The prominent α-rhythm activity at $\simeq$ 10.5Hz is captured by the second and third hidden processes (red and green), while the first one explains all activity below the α band. The spectrum of the estimated and original signal are nearly identical. The bottom inset shows 6s of each hidden process contribution, with same color coding as top inset.

Multi-channel data

In this experiment we analyzed all the MEG sensors for $n_t = 2500$ time points, and splitting the data in magnetometers ($n_c = 94$) and gradiometers ($n_c = 195$). Again we set $n_s = 1$, $n_\beta = 2$ to obtain sinusoidal contributions modulating corresponding spatial maps. We tried $\dot{n}_{KP} = 10, 25, 50, 100, 150, 200$, and got a minimum of BIC (but not of AIC or AICc) at $\hat{n}_{KP} = 100$ for magnetometers, and at $\hat{n}_{KP} = 200$ for gradiometers. We then compared the results with the multitaper analysis of the temporal series analyzed. To this end we grouped the hidden processes by the mode of its frequency response, using frequency bands 0.5Hz wide, and computed the per-band estimate of the signal standard deviation for each sensor location. Figure 3-5 shows topographic plots of the multitaper and KronEM standard deviation estimates corresponding to the frequency bands surrounding α activity ($8.5 - 12.5$Hz), where KronEM provides a more compact localization in space and frequency of the two components of the α-rhythm when compared to multi-taper.

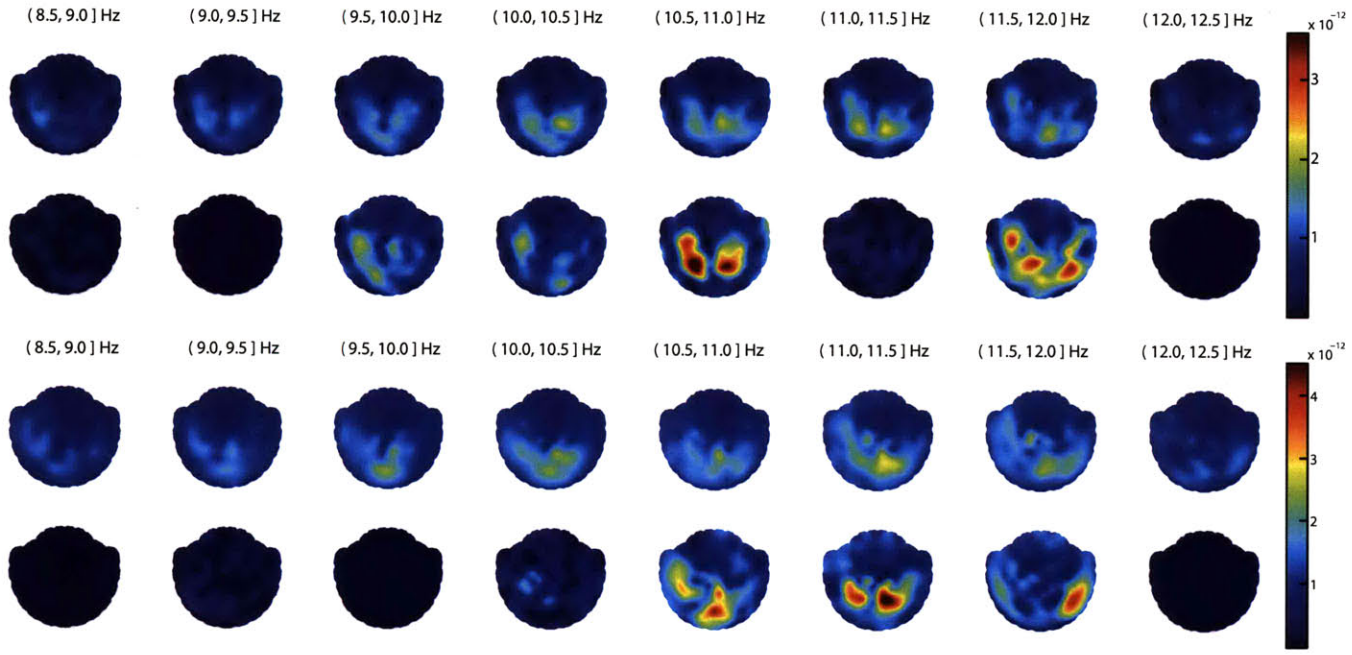


Figure 3-5: KronEM and multitaper estimates for the magnetometer activity (upper two rows) and the gradiometers (lower two rows) for $8.5 - 12.5$ Hz; upper row shows multitaper estimate, lower row KronEM estimate. KronEM and multitaper show similar patterns, but KronEM ones are more localized both in space and in frequency. In the magnetometers the α activity gets separated in two peaks as was seen in the single-channel data set, (see figure 3-4). Both magnetometers and gradiometers show occipitally localized α activity.

3.4 Discussion

In this chapter, we presented KronEM, an algorithm based in the accelerated SS framework introduced in Chapter 2 that estimates the spatiotemporal covariance matrix of MEG measurements acquired from a subject in steady-state. The KP structure reduces the number of parameters of the covariance matrix to be estimated, effectively decoupling the covariance elements in the product of a spatial and temporal term, and has been shown to effectively explain the power in the spatiotemporal signal in former works. In contrast with previously proposed methods for spatiotemporal covariance estimation using KPs, KronEM does not impose a limit in the number of KP structures that can be estimated, does not use approximations in its formulation, and uses efficient statistical metrics to decide on the number of KP structures needed to explain the data.

KronEM imposes a sum-KP structure on the resulting covariance by reformulating the problem in a state-space framework. In its estimation step, KronEM uses a Kalman smoother to recover MAP estimates of the hidden process, and uses the EM algorithm to estimate the parameters defining the spatiotemporal covariance that generalizes to any number of KP components. In its model selection step, KronEM chooses the most correct number of KPs over a set of candidate values by minimizing an information criterion (IC). We illustrate the use of the algorithm in synthetic and experimental data.

We first applied KronEM results to synthetic data sets whose covariance structure corresponded with different number of KP products. In the model selection step, KronEM recovered the correct number of KPs for all data sets. This contrast with alternative methods that do not provide with an informed way of estimating model order ([7, 72]). Out of the three ICs used for model selection (AIC, AICc, and BIC), BIC proved the most reliable, showing sharper minima on the right model order, while AIC and AICc failed to recover the model order for $n_{KP} = 5$. The different behaviors could be attributed to the heavier penalty BIC imposes on the addition of more explaining variables. Even though all ICs have been reported to be unreliable on

small data sets where $k \simeq n_t$, all ICs behaved well in our tests ($k \simeq 100, n_t = 1000$).

KronEM estimates of the spatial and temporal components of each KP were close to the real values for every data set. Of note, KronEM correctly recovered the spatiotemporal covariance when more than two KPs were needed (Figure 3-3), a situation previous methods found either mathematically challenging ([7]) or solved through approximations ([12]). Because the KS+EM algorithm used in the estimation may converge slowly, we implemented the acceleration method of [15], which reduced in half the required number of EM iterations while producing identical results (Figure 3-2).

We then applied KronEM to experimental MEG data. In the model selection step, BIC reached a minimum for all data sets, while AIC and AICc kept decreasing with higher model orders. This is consistent with KronEM behavior on the synthetic data sets. Consistent with previous results in [6], the number of KPs needed to explain the multichannel data BIC suggested was on the order of the number of sensors: KronEM explained the activity with a full-rank covariance ($n_c \simeq n_{KP}, n_s = 1$) and a full-rank white measurement noise, while [7] suggests 2 full-rank KPs (one of them with no temporal structure), and no extra measurement noise term.

Estimation using KronEM with the suggested number of KPs did also show results that were consistent with multitaper frequency estimates. In the single-channel data set, KronEM partitioned the data in physiologically relevant frequency bands, assigning the α-rhythm activity to 2 of the 14 hidden processes resulting from the analysis. Multitaper analysis of the KP contributions suggested that there were indeed two components of the α-rhythm (see Figure 3-4). In the multichannel data, KronEM estimates for both magnetometers and gradiometers showed good agreement with multitaper ones (see Figure 3-5), while producing frequency content estimates that are more compact both in space and frequency.

3.5 Summary

This chapter introduced KronEM, an algorithm providing a principled way to model the spatiotemporal covariance of MEG and EEG sensors by a sum of Kronecker products. KronEM formulates the estimation problem as a SSM, and uses the accelerated framework presented in Chapter 2 to retrieve the Kronecker product description. Compared to other available methods of spatiotemporal covariance estimation, KronEM automatically estimates the number of KP structures needed to explain the data efficiently, and then estimates the spatial and temporal components of such KPs with no approximation. When applied to experimental data, KronEM can provide with topographic, frequency resolved power estimates that show better spatial localization and automatically choose the number frequency components needed to explain the data parsimoniously.

Chapter 4

StimEM: ML input effect estimation on evoked-potential MEG studies

This chapter introduces StimEM, an algorithm for spatiotemporal estimation of brain activity using MEG measurements that uses a state-space formulation of the MEG generation. State-space formulations have already been proposed by various authors ([28, 20, 17, 52, 50]). Inversion of state-space models can be efficiently accomplished using the smoother introduced by [18], providing with full spatiotemporal inversion with arbitrary – albeit linear – dynamics. The state-space model StimEM uses accounts for evoked activity and background dynamics separately. The versatile state-space formulation allows for arbitrary spatiotemporal dynamics specification, and it uses the data to estimate the relative contribution of each candidate using the expectation-maximization algorithm (EM) as pioneered by [19]. Furthermore, StimEM uses model selection criteria ([1, 79]) to ascertain the existence of the dynamics in the data.

In spite of its versatility and power, state-space model inversion as performed by StimEM has not been previously reported on full MEG data sets due to its high computational cost. [28] neglected spatiotemporal correlations, decoupling source trajectories and making the inversion computationally tractable at the expense of

optimality and stability; [ ] applied the smoother to the event-triggered average, greatly reducing the number of time points considered. StimEM uses an accelerated version of the Kalman smoother for inference and the Expectation-Maximization algorithm for parameter estimation. Because of the lack of approximations, and in contrast with previous algorithms, StimEM provides with ML estimates of the parameters defining the state-space model, including the contribution of each candidate matrix and the evoked activity associated with each input.

In summary, this chapter presents StimEM, an algorithm that (1) accommodate for evoked potential studies by introducing an input effect term, (2) provide with a flexible way of incorporating, adding and comparing different candidate spatiotemporal dynamics, (3) prove the feasibility of the KS+EM framework with no approximations for the full-sized dynamic inverse problem. We show StimEM estimates on both simulated and experimental data sets.

4.1 Introduction

In this chapter we will focus on evoked potential studies. In these studies experimentally-administered stimuli are presented repeatedly to the subject in order to discern the brain activity elicited by each stimuli. Application of the state-space framework as used in [28, 52, 50] to evoked potential studies is problematic, since brain activity dynamics during stimulus presentation differ significantly from that of stimulus-free, or baseline, activity. In this section we provide a simple example that shows how if this duality is not explicitly modeled, the state-space estimated using the experimental data will reflect only one of the two dynamics, and the estimates will be severely biased.

In our simple model, brain activity x_t is a simple 1D process conforming a first order AR process that is affected by the inputs u_t as

$$x_{t+1} = ax_t + \sum_{\tau=1}^{n_\tau} \mathbf{b}_\tau u_{t-\tau+1} + v_t, \tag{4.1}$$

where the innovations $v_t \sim \mathcal{N}(0, \sigma_v^2)$ and white, $x_0 \sim \mathcal{N}(0, \sigma_0^2)$ is independent of v_t, a is the AR coefficient, and $\mathbf{b} \in \mathbb{R}^{n_\tau \times 1}$ models the input effect. The input $u_t \in \{0, 1\}$ is an indicator function taking the value 1 when stimulus is presented.

We are asked to recover x_t, and for that we are given the input signal u_t and a signal y_t that is a scaled version of x_t corrupted by white noise,

$$y_t = cx_t + w_t, \tag{4.2}$$

where $w_t \sim \mathcal{N}(0, \sigma_w^2)$.

We know the observation model parameters $\{c, \sigma_w^2\}$, as well as the measurements y_t and the input u_t. In order to estimate x_t we need to estimate the parameters defining the dynamic model, $\{a, \mathbf{b}, \sigma_v^2\}$.

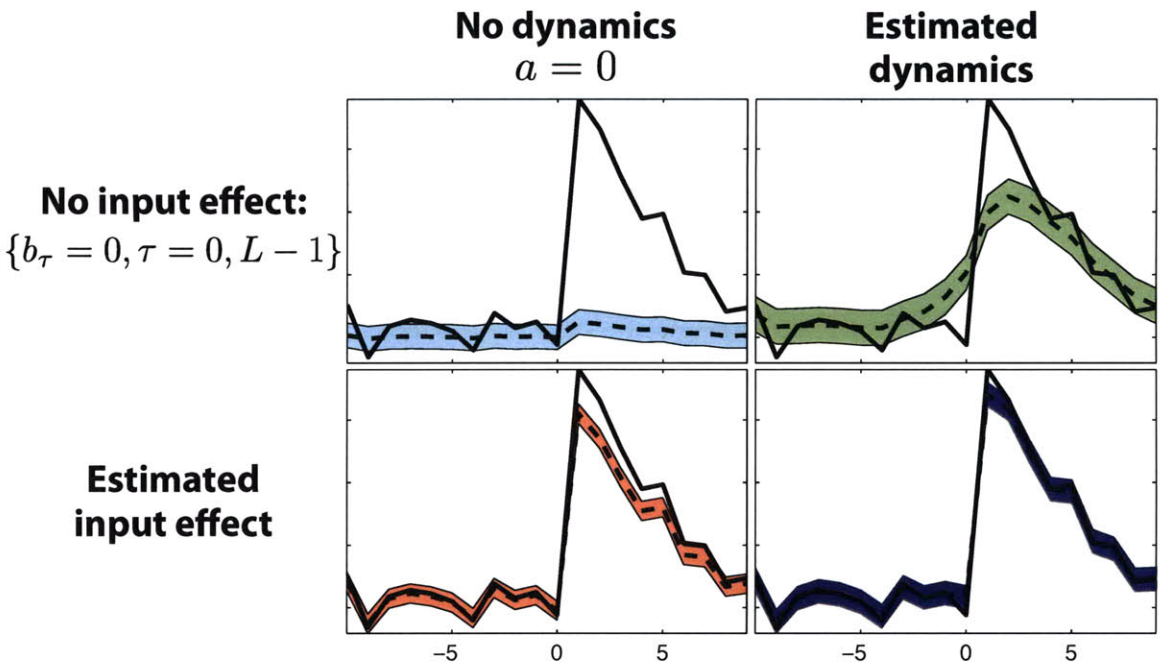


Figure 4-1: imple 1D problem showing the need for separate dynamics and input effect estimation: data generated using Eq. 4.1 and 4.2. Plots show the true stimulus average (solid black line), the estimated effect (dashed black), and corresponding confidence intervals (solid color area).

In our simulation,

$$\mathbf{b}_\tau = \begin{cases} 1, & \tau = 1 \\ 0, & \tau > 1 \end{cases} \tag{4.3}$$

Because u_t is not autorregresive and only takes the values 0 (no stimulus present) or 1 (present), the dynamics of x_t will be very different when the stimulus is presented (evoked activity) and when it is not (resting-state): there will be a big spike whenever the input u_t jumps to 1. We will need to estimate both the input effect and the dynamics to get a correct estimate of the evoked activity. To illustrate this, we run the Kalman Smoother paired with the EM algorithm, and then plot the stimulus-locked averages of the estimates of x_t (Figure 4-1). The precision of the estimates for the averaged evoked response is conveyed by their covariances, computed using [46]. We show this estimates in four different scenarios:

- Neither a nor b are estimated and are set to 0. In this case only σ_v^2 is estimated

and the results are very biased towards 0 (as it was to be expected, see [20]). To accommodate for the two dynamics, EM overestimates σ_v^2, resulting in less precise estimates.

- When a is estimated and $b = 0$, we get a better estimate that does not show the bias towards 0. Now, because the stimulus effect is much faster than the baseline dynamics, the estimated input effect is blurred in time to accommodate for this difference. Also σ_v is overestimated to accommodate for the differences in stimulus onset, making for less precise estimates.

- When $a = 0$ and b is estimated, we recover a scaled version of the input effect, due to the bias induced by neglecting the estimation of a, as happened in the first case. This scenario is formally equivalent of doing MNE estimation with EM estimating the source space covariance, and will be used as means of comparison later on in this chapter.

- If we estimate both the dynamics and the input effect, we get a estimate that is less biased and more precise, as is to be expected when the estimated model better reflects the underlying dynamics.

4.2 Methods

4.2.1 StimEM model

In StimEM, the observed data $\mathbf{y}(t)$ are modeled with the state-space model

$$\begin{aligned} \mathbf{x}(t) &= \mathrm{A}\mathbf{x}(t-1) + \mathrm{B}\mathbf{u}(t) + \mathbf{v}(t), && t = 1, \ldots, n, \\ \mathbf{y}(t) &= \mathrm{C}\mathbf{x}(t) + \mathbf{w}(t), && t = 1, \ldots, n, \end{aligned} \tag{4.4}$$

where $\mathbf{x}(t) \in \mathbb{R}^{n_s \times 1}$ is the brain activity at time t, $\mathrm{A} \in \mathbb{R}^{n_s \times n_s}$ is the state transition matrix, $\mathrm{B} \in \mathbb{R}^{n_s \times n_u}$ is the input effect matrix, $\mathbf{u}(t) \in \mathbb{R}^{n_u n_{lag} \times 1}$ is the input (see below), C is the observation matrix, $\mathbf{v}(t) \sim \mathcal{N}(\mathbf{0}, \mathrm{Q})$ is the process noise, and $\mathbf{w}(t) \sim \mathcal{N}(\mathbf{0}, \mathrm{R})$ is the detector noise.

Each of the elements of $\mathbf{u}(t)$ corresponds to a combination of lag and input, This formulation of B and $\mathbf{u}(t)$ assumes that the stimulus effect is a function of lag (time from stimulus presentation) and the type of stimulus. Specifically, the input effect is supposed to be equal across trials, and additive to the background activity (signal-plus-noise, or SPN, assumption). This formulation does not include habituation effects, responses to oddball stimuli, and event-related synchronization and de-synchronization, or any other source of trial-to-trial variability. Since these input effect non-stationarities are known to be small and slow-acting ([18, 88]), so this model provides with a good approximation as long as the number of trials included in the analysis do not span long in time.

To reduce the number of parameters of the model, we will parameterize the state transition matrix as a linear combination of candidate transition matrices $\mathrm{A}^{(i)}$,

$$\mathrm{A} = \sum_{i=0}^{n_A} \alpha_i \mathrm{A}^{(i)}, \tag{4.5}$$

so that estimating A is reduced to estimating the vector of associated coefficients $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_{n_A}]$. The particular form of each candidate matrix $\mathrm{A}^{(i)}$ is not specified. In practice, we would try matrices designed following physiological considerations. In

this work we use candidate matrices whose elements are described by

$$\mathrm{A}_{i,j} = \begin{cases} exp(-(\frac{d(i,j)}{\tau_i})^2) - exp(-4) & d(i,j) < 2\tau \\ 0 & d(i,j) > 2 \end{cases} \tag{4.6}$$

where $d(i,j)$ is the distance along the cortical surface mediating from source location i and j. To compute $d(i,j)$ we use the triangulated surface of the *pia mater* recovered by FreeSurfer. Cortical distances $d(i,j)$ are then computed as the length of the shortest path through the nodes of said triangulated surface. The formulation warrants a finite support for correlations among consecutive time steps, and makes such correlations decay with cortical distance as suggested by anatomy.

The source activity is correlated over time only through the state-transition matrix A, and the innovations in the source activity are modeled similar in all source locations, thus the covariance Q is a multiple of the identity matrix,

$$\mathrm{Q} = \sigma_v^2 \mathrm{I}. \tag{4.7}$$

The measurement noise covariance R can be estimated from sensor readouts with no subject in the MEG scanner (empty-room measurements), and will be fixed for the experiments presented in this chapter. Including its estimation in the EM would be straightforward, but would greatly increase the number of degrees of freedom of the StimEM model.

4.2.2 Parameter estimation

In order to recover the input effect B, the estimates of $\mathbf{x}(t)$ have to be accurate. For that to happen we need estimates of all the parameters of the model in 4.4. While the observation matrix C and the measurement noise covariance R can be estimated from anatomical MRI and empty room measurements, respectively, we still need to estimate α, the input effect B, and the process noise covariance, parameterized by σ_v^2. This set of parameters $\Theta = \{\mathrm{B}, \sigma_v^2, \boldsymbol{\alpha}\}$ are estimated from the data using the Expectation-Maximization (EM) algorithm.

Because of the structure of our SSM, the E step uses a Kalman smoother (KS) to recover MAP estimates of the source activity $\mathbf{x}(t)$. The KS is a two-pass algorithm that first computes the filtered estimates and then proceeds backwards to incorporate future measurements in the estimate. The filter was originally described in [18] while [2] presented a particularly elegant derivation of the smoother. The combined application of EM and KS was first introduced and thoroughly described in the classic work of [82].

We introduced a couple of modifications to both the E and M steps to reduce computation time and guarantee stability. In each run of the KS (the E step) we check for observability and stochastic controllability using the methods described by [39], to ensure that the KS will behave properly. We then employ the doubling algorithm as explained in [26] to obtain the steady-state filter and smoother covariance. If the steady-state filter and smoother covariance exist, we use the steady-state values for both once steady-state is reached, reducing the time spent in computations. The M-step is similar to that described in [9], but the update for $\boldsymbol{\alpha}$ in the presence of the input effect is an original contribution of this thesis that can be found in Section 2.4.2.

4.2.3 Initial values

Because the EM algorithm tends to show slow convergence, it is important to find initial values for the parameters to estimate that are close to the real underlying values. Experimental evidence showed that B converged quite fast, and thus was set to zero on the initial parameters. On the other hand, the initial values for σ_v^2 and $\boldsymbol{\alpha}$ can prevent EM to converge fast if set too far from its final values, so there is a need to estimate them, albeit approximately, before starting the EM iterations.

Our first approach was fitting a first order MVAR to the observed data $\mathbf{y}(t)$ so that

$$\mathbf{y}(t) = \mathrm{D}\mathbf{y}(t) + \mathbf{n}(t) \tag{4.8}$$

using the Nutall-Strand (multivariate Burg) algorithm as implemented in the ARFIT Matlab© package (*c.f.* [75]). After the fitting we recover an estimate for D and for

the covariance of $\mathbf{n}(t)$, Λ_n.

Using the equations of the SSM, we can transform Eq. 4.8 into

$$\mathrm{CA}\mathbf{x}(t-1) = \mathrm{DC}\mathbf{x}(t-1) + \left(\mathrm{C}\mathbf{w}(t) + \mathbf{v}(t) - \mathrm{D}\mathbf{v}(t-1) + \mathbf{n}(t)\right), \tag{4.9}$$

since $\mathbf{v}(t)$ and $\mathbf{w}(t)$ are zero mean, we can approximate

$$\mathrm{CA} \simeq \mathrm{DC}, \tag{4.10}$$

and this way we can get our initial parameters for $\{\alpha_i\}$ solving

$$\sum_{i=1}^{n_A} \alpha_i \mathrm{CA}_i \simeq \mathrm{DC}. \tag{4.11}$$

We can further identify

$$\mathbf{n}(t) \simeq \mathrm{C}\mathbf{w}(t) + \mathbf{v}(t) - \mathrm{D}\mathbf{v}(t-1), \tag{4.12}$$

and then equating the variance of both sides of the equation we can fit σ_v^2 in

$$\sigma_v^2 \mathrm{CC}^\mathsf{T} \simeq \Lambda_n - \mathrm{R}\left(\mathrm{I} + \mathrm{DD}^\mathsf{T}\right). \tag{4.13}$$

This first approach did not work well, since the estimation of Eq. 4.11 involves the inversion of a badly conditioned matrix for the choice of candidate dynamics that was made in this study.

Our second approach made use of the observation that the sum of the estimated weights, $\sum_{i=1}^{N_a} \hat{\alpha}_i$ did converge quickly to its real value no matter what the initial conditions. Hence, a first run of the KS+EM determines the value of such quantity, and the initial matrix A is set to an even contribution of each candidate matrix totaling the estimate for the sum of the coefficients. Then we can compute the steady-state value for the process noise covariance solving the discrete version of the

Lyapunov equation

$$\Lambda = \mathrm{A}\Lambda\mathrm{A}^{\mathsf{T}} + \mathrm{Q}. \tag{4.14}$$

In our model, $\mathrm{Q} = \sigma_v^2\mathrm{I}$, and the solution of this equation is

$$\Lambda = \sigma_v^2 \sum_{i=0}^{\infty} (\mathrm{A})^i (\mathrm{A}^{\mathsf{T}})^i = \sigma_v^2 \Lambda_1, \tag{4.15}$$

Using the value of A to compute Λ_1, we can estimate the value of σ_v^2 by fitting the experimental and theoretical data covariance Σ_y,

$$\sigma_v^2 \mathrm{C}\Lambda_1\mathrm{C}^{\mathsf{T}} \simeq \Lambda_y. \tag{4.16}$$

This second method for estimating the initial values did produce better results on simulated data, and was the one used in the experimental results shown later on.

4.2.4 Data acquisition and preprocessing

The MEG data were acquired with a 306-channel MEG system (Vectorview, Elekta-Neuromag, Helsinki, Finland). The coils of the MEG channels are arranged in a hemispherical mosaic with 102 locations. At each location, a magnetometer measures the normal component of the magnetic field, while two planar gradiometers measure the two perpendicular off-diagonal gradients of the same component (see [36]). Bandwidth was set at 0.1 to 334 Hz, and data were digitized at 1798 Hz. The relative position of the head and MEG array was determined at the beginning of each acquisition by feeding currents to the coils and by locating them on the basis of the magnetic fields measured with the MEG sensors ([87]). The auditory stimuli consisted of 60 dBSL monaural 1kHz sinusoidal tone bursts (400 ms duration). Tone pips were presented independently, alternating in each ear, at inter-stimulus interval of 0.9 – 1.1 s, with each tone occurring at least 100 times. Hearing thresholds were determined individually for each frequency.

Signal-space projectors (SSP, [85]) where applied to the data to suppress the ECG and eye-movement related artifacts. These SSP used the main PCA directions of the

data only at the times when a given artifact was maximal, and the source of the artifact was identified using the concurrently acquired ECG and EOG of the subject. Data were then band-pass filtered from 1 to 40 Hz, and downsampled 15 times so that the Nyquist frequency (81.72 Hz) was twice that of the maximum frequency component in the signal. SSP removes a n_{SSP}-dimensional noise subspace from the data, and therefore is a rank-reducing operation which makes the covariance singular. This would cause trouble in the KS+EM algorithm. To ensure a full-rank covariance we excluded from subsequent analysis the n_{SSP} channels whose exclusion least reduced the the power in the signal subspace.

4.3 Results

4.3.1 Simulations

A simulated data set of 3000 time points was constructed using an observation equation identical to that of the subject under study: the matrix C was given by the forward model coming from the BEM, and the measurement noise covariance R was estimated from the same empty-room measurements used in the real data analysis.

The input effect B was decoupled in the product of a spatial and temporal component. For the spatial component, a set of 7 contiguous sources were selected, with the central source location contained within the auditory cortex, whose location was previously delineated using a brain atlas. We generated six regions in this fashion, and each was associated with a different input and a different temporal response of 400ms. The maximum amplitude of the temporal response for each location was chosen so that the maximum activity generated by any input would peak at 10 standard deviations from the noise in any element of $\mathbf{y}_t$, a level similar to that observed in the experimental data. We used equation 4.5 to generate 3 candidate transition matrices $\mathrm{A}^{(i)}$ with τ = 7.5 mm, 15 mm, 22.5 mm, respectively. In each dataset, a weighted version of one of the candidate dynamics was used as the state-transition matrix, $\mathrm{A} = 0.0\mathrm{A}^{(1)}$, $\mathrm{A} = 0.99\mathrm{A}^{(1)}$, $\mathrm{A} = 0.99\mathrm{A}^{(2)}$, $\mathrm{A} = 0.99\mathrm{A}^{(3)}$. The process noise activity σ_v^2 was chosen so that the regularized SNR,

$$SNR = \frac{1}{n_d} trace(\mathrm{R}^{-1/2}\mathrm{CQ}(\mathrm{R}^{-1/2}\mathrm{C})^{\mathsf{T}}) = 5. \tag{4.17}$$

The simulated inputs were administered each 800ms, and assigned at random.

The input effect was then estimated with the full StimEM algorithm. A second estimate was obtained fixing the transition matrix to zero, which is equivalent to a minimum-norm estimate (MNE, [37]) with a data-driven estimate for the signal to noise ratio. This second estimate is static, but since uses the EM algorithm, generates a MAP estimate of the process noise covariance. Both estimates should be equivalent when there is no dynamic in the data. We ran the EM algorithm for 25 iterations, on

the static and dynamic cases, and found the per-iteration increments of log-likelihood decreased monotonically for all data sets. In order to do model selection, we computed the Akaike information criterion (AIC, [1]) and the Bayesian information criterion (BIC, [79]). We should select the estimation method producing the smallest value for AIC or BIC in order to avoid overfitting, hence only the difference in values among the static and dynamic estimate is reported (ΔAIC, ΔBIC); when the difference is positive, the model should be chosen. We also report the mean error ℓ_2-norm of the error (MLE) on the estimates for both methods. The MLE is normalized by the norm of the real data,

$$M\bar{L}E = \frac{\sum_{i=1}^{3000} |\mathbf{x}(t) - \hat{\mathbf{x}(t)}|}{\sum_{i=1}^{3000} |\mathbf{x}(t)|}, \tag{4.18}$$

where $|\cdot|$ denotes the ℓ_2 norm. All results are summarized in table 4.4.

α_1	α_2	α_3	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	Δll	ΔAIC	ΔBIC	MLE
.00	.00	.00	.01	.02	.02	$-4.77 \cdot 10^0$	$+15.54 \cdot 10^0$	$+15.01 \cdot 10^0$	1.18%
			–	–	–	$+4.77 \cdot 10^0$	$\mathbf{-15.54 \cdot 10^0}$	$\mathbf{-15.01 \cdot 10^0}$	**1.18%**
.99	.00	.00	.37	.33	.30	$+2.37 \cdot 10^4$	$\mathbf{-4.74 \cdot 10^4}$	$\mathbf{-4.74 \cdot 10^4}$	**4.21%**
			–	–	–	$-2.37 \cdot 10^4$	$+4.74 \cdot 10^4$	$+4.74 \cdot 10^4$	8.26%
.00	.99	.00	.36	.10	.53	$+1.38 \cdot 10^4$	$\mathbf{-2.76 \cdot 10^4}$	$\mathbf{-2.76 \cdot 10^4}$	**3.41%**
			–	–	–	$-1.38 \cdot 10^4$	$+2.76 \cdot 10^4$	$+2.76 \cdot 10^4$	14.21%
.00	.00	.99	.35	-.02	.65	$\mathbf{+7.18 \cdot 10^3}$	$\mathbf{-1.43 \cdot 10^4}$	$\mathbf{-1.43 \cdot 10^4}$	**3.68%**
			–	–	–	$-14.36 \cdot 10^3$	$+1.43 \cdot 10^4$	$+1.43 \cdot 10^4$	12.38%

Table 4.1: Results of StimEM on the simulated data sets. Successive rows show the resulting likelihood, information criteria and error of the static and dynamic versions of StimEM applied to different datasets. In each data set, the version producing lower BIC is shown in bold.

The input-effect estimates is computed from the MAP estimates of the SSM parameters as follows

$$\mathbf{e}^{(i)}(\tau) = \hat{\mathbf{A}}\mathbf{e}(t-1) + \mathrm{b}^{(i,\tau)} \tag{4.19}$$

where $\mathbf{e}^{(i)}(\tau)$ is the input effect for input i at lag τ from stimulus onset, and $\mathrm{b}^{(i,\tau)}$ is the column of $\hat{\mathrm{B}}$ corresponding to input i, lag τ. The input-effect estimates for the

different stimuli can be seen in Fig. 4.4. Each row is normalized by the maximum value of the estimate across all lags considered, and thresholded at half that value.

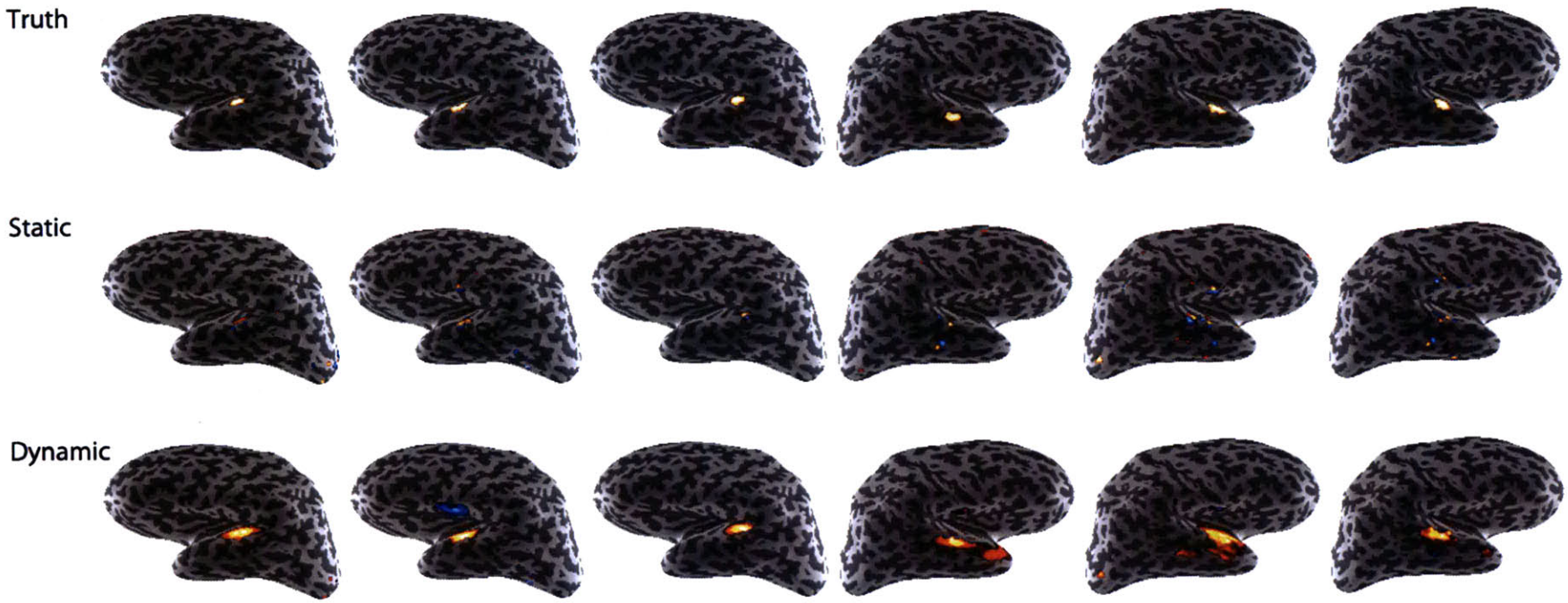


Figure 4-2: Resulting topographic maps of input effect on the simulated data set with $\tau = 15\text{m}m$. Maps shown are for the lag where the true effect reaches its maximum value. Upper row shows the true location of the effect, center row shows the static estimates where $\boldsymbol{\alpha} = [.0, .0, .0]$, lower row shows dynamic estimates with $\boldsymbol{\alpha} = \hat{\boldsymbol{\alpha}}$ (see 4.4). Each map is normalized to the maximum value across all, and thresholded at half this value.

4.3.2 Experimental data

We analyzed 25000 contiguous data points from the experimental data set described in the methods section using StimEM, containing 20 to 25 stimuli presentations. We run the EM algorithm for 25 iterations, and used equation 4.5 to generate 3 candidate transition matrices $A^{(i)}$ with $\tau = 7.5$ mm, 15 mm, 22.5 mm, respectively. StimEM got an estimate for $\boldsymbol{\alpha} = \boldsymbol{\alpha} = [.24, .75, .00]$ after 25 iterations, and both BIC and AIC deemed necessary to include the dynamics, with $\Delta ll = +1.72 \cdot 10^5$, $\Delta AIC = +3.44 \cdot 10^5$, $\Delta BIC = +3.44 \cdot 10^5$. The input-effect estimates were spatially similar for the four different stimuli, with the input-effect for stimulus 3 shown in Figure 4.4, at different lags from stimulus presentation. In the figure, both time series have been normalized to the maximum value of the estimates across time-points, and thresholded to half that value in order to display the spread of the estimates.

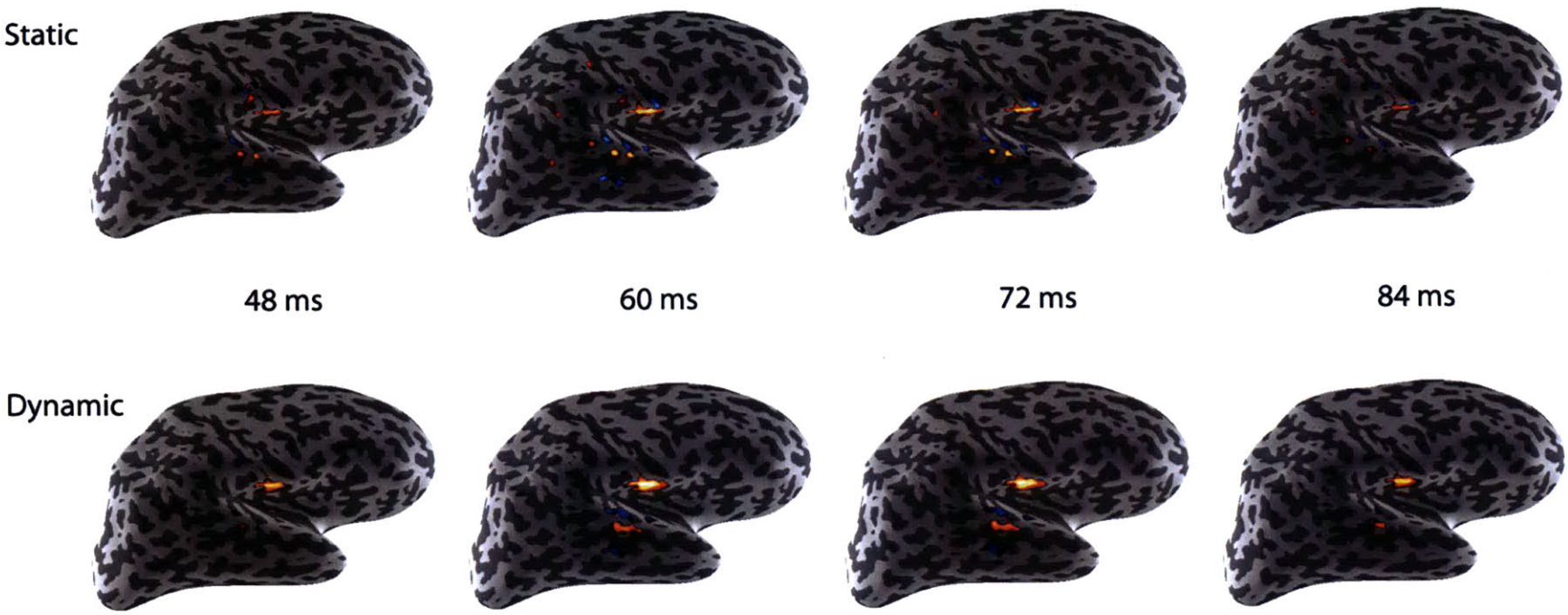


Figure 4-3: Input-effect estimated from experimental data. Upper row shows static estimates where $\boldsymbol{\alpha} = [.0, .0, .0]$, lower row shows dynamic estimates with $\hat{\boldsymbol{\alpha}} = [.24, .75, .00]$. Each row is normalized to the maximum value across all sources and lags, and thresholded at half this value. The dynamic solution produced higher likelihood and was chosen by both selection criteria.

4.4 Discussion

In this chapter, we presented StimEM, an algorithm based in the accelerated SS framework introduced in Chatper 2 that estimates the effect of experimentally controlled inputs on the cortical activity of a human subject using the MEG recordings obtained during an evoked-potential study. The stimulus effect is naturally incorporated in the state-space formulation of the inverse problem, and can be recovered using the EM algorithm as was described in Chapter 2. In contrast with previously proposed methods for stimulus effect estimation ([28, 50, 66, 22, 25]), StimEM estimates both the input effect and the neural dynamics, and does not impose separable, algorithm-specified neural dynamics. To reduce the number of parameters describing the neural dynamics, StimEM models the state-transition matrix of the SS model as a linear combination of candidate dynamics specified by the researcher, and retrieves ML estimates of the corresponding weights using the EM algorithm. Finally, StimEM uses information criteria to decide if the inclusion of the dynamics is supported by the data.

We first used simulations to better characterize the behavior of StimEM in the state-space models that MEG creates, since previous works have avoided using the full KS+EM algorithm due to its high computational cost. We were able to analyze MEG using the accelerated, distributed implementation that was introduced in Chapter 2, using no approximations. The results obtained on simulation data, show in table , proved that (1) StimEM is able to detect the candidate dynamics we used to generate the data, (2) StimEM did not impose the dynamics when the underlying data was temporally white, (3) when using the dynamics, StimEM estimates showed decreased error and localization, and (4) the coefficients weighting each dynamic were not correctly retrieved from the data. Together, (1)-(3) suggest that StimEM is improving the estimates by using the dynamics only when they are needed. The topographic maps seen in figure showed better spatial localization of activity, and the errors in table showed small error across all temporal samples, in accordance with the theoretical analysis by [51]. The incorrect values for the coefficients might be due to the

non-orthogonal candidate functions used in the simulation, which make non-unique the combination of weights producing a given dynamic; in light of the decreased error and better spatial localization, is safe to assume that the particular value for the estimated weights does not impair the ability of StimEM to use the candidate dynamics to improve the spatial localization of the cortical estimates.

When applied to experimental MEG data acquired during an evoked-potential study, StimEM (1) found evidence supporting the candidate dynamics and (2) used the dynamics to recover activity estimates that are more focal than its static counterpart, as shown in the topographic maps of figure . The estimated neural activity shown a phantom source in the inferior parietal and frontal lobe that was also seen in the simulation results, and hence is a consequence of the ill-posedness of the forward model, consequence of the subject's anatomy and positioning with respect to the MEG sensors. However, the phantom sources seen in the dynamic solution as compared with the static, the increase in log-likelihood, BIC and AIC when dynamics are included, and the behavior of SimEM in simulated data strongly suggest that the focality of the resulting dynamic estimates will correspond to estimates with less error.

It is important to remark that the specification of the candidate dynamics is an open problem that falls out of the scope of this thesis. To produce the results on this chapter, we used simple spatiotemporal dynamics suggested by neurophysiology that are unlikely to capture all the dynamics of the underlying neural activity. However, unlike other dynamic methods, StimEM can accommodate any linear dynamic in its formulation. Future work will investigate the use of dynamics suggested by diffusion tensor imaging (DTI), causal analysis of intracranial EEG, or other estimates obtained from neurophysiology or experimental analysis. Including these more accurate approximations to neural dynamics can improve the estimates dramatically. However, the simple dynamics used in this study already demonstrated that StimEM provides with better source localization for MEG evoked-potential studies. Moreover, the extension of StimEM to EEG or combined EEG+MEG studies would require little modification of the underlying framework to accommodate for the estimation of

the unknown EEG measurement error covariance.

4.5 Summary

In this chapter we introduced StimEM, an algorithm that uses state-space modeling of the MEG forward model to estimate both the stimulus effect and neural dynamics using only MEG recordings acquired during evoked-potential studies. This is made possible by the accelerated SSM framework introduced in Chapter 2, that lets us apply KS+EM to the high-dimensional state-space required for MEG time-series analysis. A simulation study proved that StimEM can successfully detect underlying neural dynamics on state-space models of the dimensions MEG requires, and in doing so it can improve the accuracy and localization of the estimated neural activity when compared to static methods. When applied to experimental MEG data, StimEM identified a dynamic component on the recordings and used the identified dynamics to produce estimates that were more localized in space than those produced by static estimation methods.

Chapter 5

Conclusions and future work

5.1 Conclusions

This thesis provided methods for state-space modeling of MEG time-series. This approach has been avoided in the past due to the high computational costs of SSM estimation algorithms when applied to the high-dimensional state-spaces required to model MEG. We provided an accelerated framework for analysis that reduces the computational complexity of two classic algorithms, the Kalman smoother and the expectation-maximization algorithm, applied to time-invariant SSMs. In Chapter 2, we offered a computational analysis of the two algorithms, and produced a new framework that reduces the number of computations, and the time required per computation. To accomplish this task, we exploited the steady-state covariances in the Kalman smoother to reduce the number of covariance updates required, and used an acceleration algorithm proposed by [ ] to halve the number of EM iterations. We also developed a distributed implementation of the algorithm using a distributed memory linear algebra library and custom distributed software. This approach scales very well for matrix multiplications and inversions, which are the main operation in the KS step in term of computations. Hence, this reduced the time spent on a given KS step almost linearly with the number of computing nodes used.

In Chapter 3, we used the accelerated framework to model the spatiotemporal covariance of MEG measurements obtained from human subjects in resting-state,

producing a new algorithm we called KronEM (Kronecker Product modeling using KS+EM). KronEM characterizes the spatiotemporal covariance of MEG recordings using an parameterization that efficiently describes the rhythmicity present in resting-state neural activity, modeling the MEG time-series as a sum of components composed of a time-invariant spatial signature and a temporal second-order autorregresive process. Unlike previous attempts at modeling resting-state activity, the KronEM algorithm estimates the number of such components using the data, and is able to identify an arbitrary number of them. We illustrated these properties in a simulation study, and then analyzed MEG recordings collected from a human subject in resting state. The KronEM algorithm recovered components consistent with well-known physiological rhythmic activity. We then compared the resulting topographic maps of frequency with multi-taper based ones, and showed that KronEM-based maps better localize naturally occurring rhythms. These results showed how the KronEM algorithm is a useful addition to traditional single-trial frequency analysis techniques.

In Chapter 4, we used our accelerated framework to estimate the effect of different stimuli in neural activity by analyzing MEG data recorded from human subjects in evoked-potential studies. This lead to a new algorithm we called StimEM (Stimulus effect estimation using KS+EM). StimEM estimates neural activity using MEG recordings made in evoked-potential studies, in which the subject is repeatedly presented with a stimulus and only the stimulus effect is of interest. In contrast with other dynamic source-localization techniques, StimEM accepts an arbitrary description of neural dynamics, parameterized as a weighted sum of user-defined candidates, and finds the MAP estimate of the weights. Using the estimated dynamics, StimEM generates a time-resolved ML estimate of the evoked-potential activity in the cortex. We used StimEM to identify dynamics in a simulated data set of realistic dimensions, and showed that the estimates improved substantially when dynamics are taken into account. We next analyzed experimental MEG data from an auditory evoked-potential study and showed that StimEM identified dynamics consistent with neurophysiology and neuroanatomy and improved the localization of the evoked cortical response.

In summary, this thesis establishes the feasibility of full-scale analysis of high-dimensional state-space models using a distributed-memory implementation of accelerated KS+EM algorithms. The thesis produced two novel algorithms to analyze MEG data in resting-state and evoked potential studies, and showed that SSM analysis improves substantially on previous non-SSM based techniques.

5.2 Future work

Because prior work has already shown how combined inference from MEG and EEG improves spatial resolution (*e.g.* [51, 60]), a very interesting direction would be to address the problem of combined inference from EEG and MEG in the dynamic setting. Unlike MEG, where empty-room noise provides with a very straightforward characterization of measurement noise, the properties of the measurement noise in EEG depend on the skin-electrode interface, making it impossible to acquire measurement noise readings alone. One possible way to address this would be to extend KronEM to estimate the EEG measurement noise covariance along with the dynamics, assuming the EEG noise to be temporally white. Since both EEG and MEG recordings originate from the same dynamic process, the algorithm would exploit the characterization of the process noise obtained from MEG alone to estimate the EEG sensor noise. Once an EEG measurement noise covariance has been extracted from the data, both StimEM and KronEM can be applied to the combined EEG+MEG data set with no modifications, and should produce estimates that are better resolved in space than those based on MEG alone.

Another development that follows naturally from this thesis would be to join KronEM and StimEM by modeling the input effect and the rhythmic processes concurrently. This could improve the estimates in situations in which strong rhythmic activity contaminates the stimulus-locked responses. Perhaps more importantly, a combined KronEM+StimEM SSM could also provide information about the cortical distribution of rhythmic activity; however, for this to work the neural dynamics should be incorporated on top of the autoregressive processes that dictate the evo-

lution of the state-space in KronEM. This is because the autorregresive processes impose dynamics that are separable in space and time, and hence would not improve the localization as discussed in [50]. Adding both the neural dynamic modeling and the autoregressive component would increase the dimensionality of the state-space, and perhaps some modifications of the accelerated framework would be needed to exploit the sparsity of the matrices describing the resulting state-space. This combined algorithm would localize cortical sources of rhythmic background activity, and could better retrieve underlying evoked potentials.

Bibliography

[1] H Akaike. A New Look at the Statistical Model Identification. *IEEE Trans Automatic Control*, 19(6):716–723, 1974.

[2] C Ansley and R Kohn. A geometrical derivation of the fixed interval smoothing algorithm. *Biometrika*, 69(2):486–487, 1982.

[3] S Baillet and L Garnero. A Bayesian approach to introducing anatomo-functional priors in the EEG/MEG inverse problem. *IEEE Transactions Biomed Eng*, 44(5):374–85, 1997.

[4] C Baumgartner, E Pataraia, G Lindinger, and L Deecke. Magnetoencephalography in focal epilepsy. *Epilepsia*, 41(Suppl 3):S39–S47, 2000.

[5] S Bechstein, F Petsche, M Scheiner, D Drung, F Thiel, A Schnabel, and T Schurig. Digitally controlled high-performance DC SQUID readout electronics for a 304-channel vector magnetometer. *Journal of Physics: Conference Series*, 43:1266–1269, 2006.

[6] F Bijma, J C de Munck, and R M Heethaar. The spatiotemporal MEG covariance matrix modeled as a sum of Kronecker products. *Neuroimage*, 27(2):402–15, 2005.

[7] F Bijma, J C de Munck, H M Huizenga, and R M Heethaar. A mathematical approach to the temporal stationarity of background noise in MEG/EEG measurements. *Neuroimage*, 20(1):233–43, 2003.

[8] S Borman. The expectation maximization algorithm: a short tutorial. *http://www.isi.edu/natural-language/teaching/cs562/2009/readings/B06.pdf*, 2004.

[9] T H Bullock, M C McClune, J Z Achimowicz, V J Iragui-Madoz, R B Duckrow, and S S Spencer. Temporal fluctuations in coherence of brain waves. *Proc Natl Acad Sci USA*, 92(25):11568–72, 1995.

[10] G Buzsáki and A Draguhn. Neuronal oscillations in cortical networks. *Science*, 304(5679):1926–9, 2004.

[11] A C N Chen, W Feng, H Zhao, Y Yin, and P Wang. EEG default mode network in the human brain: spectral regional field powers. *Neuroimage*, 41(2):561–74, 2008.

[12] D Cohen. Magnetoencephalography: Evidence of Magnetic Fields Produced by Alpha-Rhythm Currents. *Science*, 161(3843):784, 1968.

[13] D Cohen. Large-volume conventional magnetic shields. *Rev. Phys. Appl*, 5:53–58, 1970.

[14] D Cohen. Magnetoencephalography: Detection of the Brain's Electrical Activity with a Superconducting Magnetometer. *Science*, 175(4022):664, 1972.

[15] D Cohen, B N Cuffin, K Yunokuchi, R Maniewski, C Purcell, G R Cosgrove, J Ives, J G Kennedy, and D L Schomer. MEG versus EEG localization test using implanted sources in the human brain. *Annals of Neurology*, 28(6):811–817, 1990.

[16] D Cohen and E Halgren. Magnetoencephalography (neuromagnetism). *Encyclopedia of Neuroscience*, pages 1–7, 2003.

[17] Jean Daunizeau, Stefan J Kiebel, and Karl J Friston. Dynamic causal modelling of distributed electromagnetic responses. *Neuroimage*, 47(2):590–601, 2009.

[18] J C de Munck, F Bijma, P Gaura, C A Sieluzycki, M I Branco, and R M Heethaar. A maximum-likelihood estimator for trial-to-trial variations in noisy MEG/EEG data sets. *IEEE transactions on bio-medical engineering*, 51(12):2123–8, 2004.

[19] AP Dempster, NM Laird, and DB Rubin. Maximum likelihood from incomplete data via EM algorithm. *J Roy Stat Soc B Met*, 39(1):1–38, 1977.

[20] N Desai, E Brown, and S Burns. Source localization of MEG generation using spatio-temporal Kalman filter. *MIT MSc Thesis*, 2005.

[21] A Destexhe, D Contreras, and M Steriade. Spatiotemporal analysis of local field potentials and unit discharges in cat cerebral cortex during natural wake and sleep states. *J Neurosci*, 19(11):4595–608, 1999.

[22] A Dogandzic and A Nehorai. Estimating evoked dipole responses in unknown spatially correlated noise with EEG/MEG arrays. *IEEE Transactions on Signal Processing*, 2000.

[23] N Forss. Magnetoencephalography (MEG) in epilepsy surgery. *Acta neurochirurgica*, Supplement(68):81–84, 1997.

[24] M D Fox, A Z Snyder, J L Vincent, M Corbetta, D C Van Essen, and M E Raichle. The human brain is intrinsically organized into dynamic, anticorrelated functional networks. *Proc Natl Acad Sci USA*, 102(27):9673–8, 2005.

[25] K Friston, R Henson, C Phillips, and J Mattout. Bayesian estimation of evoked and induced responses. *Human brain mapping*, 27(9):722–35, 2006.

[26] I Fukumori, J Benveniste, C Wunsch, and D B Haidvogel. Assimilation of sea-surface topography into an ocean circulation model using a steady-state smoother. *J Phys Oceanogr*, 23(8):1831–1855, 1993.

[27] A Galka, T Ozaki, H Muhle, U Stephani, and M Siniatchkin. A data-driven model of the generation of human EEG based on a spatially distributed stochastic wave equation. *Cognitive neurodynamics*, 2(2):101–13, 2008.

[28] A Galka, O Yamashita, T Ozaki, R Biscay, and P Valdés-Sosa. A solution to the dynamical inverse problem of EEG generation using spatiotemporal Kalman filtering. *Neuroimage*, 23(2):435–53, 2004.

[29] D B Geselowitz. On bioelectric potentials in an inhomogeneous volume conductor. *Biophysical Journal*, 7(1):1, 1967.

[30] D B Geselowitz. On the magnetic field generated outside an inhomogeneous volume conductor by internal current sources. *Magnetics, IEEE Transactions on*, 6(2):346–347, 1970.

[31] P Gloor. Neuronal generators and the problem of localization in electroencephalography: Application of volume conductor theory to electroencephalography. *Journal of clinical neurophysiology*, 2(4):327–354, 1985.

[32] I F Gorodnitsky and B D Rao. Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm. *IEEE Transactions on signal processing*, 45(3):600–616, 1997.

[33] F Greensite. The temporal prior in bioelectromagnetic source imaging problems. *IEEE transactions on bio-medical engineering*, 50(10):1152–9, 2003.

[34] D A Gusnard, M E Raichle, and M E Raichle. Searching for a baseline: functional imaging and the resting human brain. *Nat Rev Neurosci*, 2(10):685–94, 2001.

[35] J Hadamard. *Lectures on Cauchy's Problem in Linear Partial Differential Equations.* Dover, 1923.

[36] M S Hämäläinen, R Hari, R Ilmoniemi, and J Knuutila. Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain. *Reviews of Modern Physics*, 65(2):413–497, 1993.

[37] M S Hämäläinen and R J Ilmoniemi. Interpreting magnetic fields of the brain: minimum norm estimates. *Med Biol Eng Comput*, 32(1):35–42, 1994.

[38] M S Hämäläinen and J Sarvas. Realistic conductivity geometry model of the human head for interpretation of neuromagnetic data. *IEEE transactions on bio-medical engineering*, 36(2):165–171, 1989.

[39] M L J Hautus. Controllability and observability conditions of linear autonomous systems. *Proc. of the Koninklijke Nederlandse Akademie van Wetenschappen, Series A*, pages 443–448, 1969.

[40] H Helmholtz. Ueber einige Gesetze der Vertheilung elektrischer Ströme in körperlichen Leitern mit Anwendung auf die thierisch-elektrischen Versuche. *Annalen der Physik und Chemie*, 165(6):211–233, 1853.

[41] B Horwitz and D Poeppel. How can EEG/MEG and fMRI/PET data be combined? *Human brain mapping*, 17(1):1–3, 2002.

[42] H M Huizenga, J C de Munck, L J Waldorp, and R P Grasman. Spatiotemporal EEG/MEG source analysis based on a parametric noise covariance model. *IEEE transactions on bio-medical engineering*, 49(6):533–9, 2002.

[43] C M Hurvich, R Shumway, and C L Tsai. Improved estimators of Kullback-Leibler information for autoregressive model selection in small samples. *Biometrika*, 77(4):709–719, 1990.

[44] C M Hurvich and C L Tsai. Regression and time-series model selection in small samples. *Biometrika*, 76(2):297–307, 1989.

[45] S Ikeda. Acceleration of the EM algorithm. *Systems and Computers in Japan*, 2000.

[46] P T De Jong and M J Mackinnon. Covariances for smoothed estimates in state-space models. *Biometrika*, 75(3):601–602, 1988.

[47] T Kailath and B Hassibi. *Linear estimation*. Prentice-Hall, 2000.

[48] R Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(D):35–45, 1960.

[49] S Kullback. *Information theory and statistics*. Dover, 2nd edition, 1997.

[50] C Lamus, M S Hämäläinen, S Temereanca, E N Brown, and P L Purdon. Estimation and identification in spatiotemporal dynamic distributed models for the meg inverse problem. *Under Review*, Aug 2010.

[51] C Lamus, M S Hämäläinen, S Temereanca, E N Brown, and P L Purdon. How dynamic models of brain connectivity can help solve the meg inverse problem: Analysis of source observability. *Under Review*, Aug 2010.

[52] C Lamus, C Long, and M Hamalainen. Parameter estimation and dynamic source localization for the magnetoencephalography (MEG) inverse problem. *Proceedings of ISBI*, 2007.

[53] H Laufs, K Krakow, P Sterzer, E Eger, A Beyerle, A Salek-Haddadi, and A Kleinschmidt. Electroencephalographic signatures of attentional and cognitive default modes in spontaneous brain activity fluctuations at rest. *Proc Natl Acad Sci USA*, 100(19):11053–8, 2003.

[54] M Lauritzen and L Gold. Brain function and neurophysiological correlates of signals used in functional neuroimaging. *J Neurosci*, 23(10):3972–80, 2003.

[55] D A Leopold, Y Murayama, and N K Logothetis. Very slow activity fluctuations in monkey visual cortex: implications for functional brain imaging. *Cereb Cortex*, 13(4):422–33, 2003.

[56] C Loan. The ubiquitous Kronecker product. *Journal of computational and applied mathematics*, 123:85–100, 2000.

[57] T A Louis. Finding the observed information matrix when using the EM algorithm. *J Roy Stat Soc B Met*, 44(2):226–233, 1982.

[58] A N Mamelak, N Lopez, M Akhtari, and W W Sutherling. Magnetoencephalography-directed surgery in patients with neocortical epilepsy. *Journal of neurosurgery*, 97(4):865–873, 2002.

[59] G J McLachlan and T Krishnan. *The EM algorithm and extensions.* John Wiley Sons, 2008.

[60] A Molins, S Stufflebeam, E Brown, and M Hämäläinen. Quantification of the benefit from integrating MEG and EEG data in minimum ℓ_2-norm estimation. *Neuroimage*, 42(3):1069–1077, 2008.

[61] J C Mosher, R M Leahy, and P S Lewis. EEG and MEG: forward solutions for inverse methods. *IEEE Trans on Biomedical Engineering*, 46(3):245–259, 1999.

[62] J De Munck, H Huizenga, L Waldorp, and R Heethaar. Estimating stationary dipoles from MEG/EEG data contaminated with spatially and temporally correlated background noise. *IEEE Transactions on Signal Processing*, 2002.

[63] S Murakami and Y Okada. Contributions of principal neocortical neurons to magnetoencephalography and electroencephalography signals. *The Journal of physiology*, 575(3):925–936, 2006.

[64] S S Nagarajan, H Attias, K Hild, and K Sekihara. A Graphical Model for Estimating Stimulus-Evoked Brain Responses in Noisy MEG data with Large Background Brain Activity. *Invited paper at International Conference on biomagnetism and Non-invasive Functional Source Imaging*, 2005.

[65] S S Nagarajan, H T Attias, K E Hild, and K Sekihara. A graphical model for estimating stimulus-evoked brain responses from magnetoencephalography data with large background brain activity. *Neuroimage*, 30(2):400–16, 2006.

[66] P L Nunez. *Neocortical dynamics and human EEG rhythms.* Oxford University Presss, 1995.

[67] P L Nunez and R B Silberstein. On the relationship of synaptic activity to macroscopic measurements: Does co-registration of EEG with fMRI make sense? *Brain topography*, 13(2):79–96, 2000.

[68] W Ou, M S Hämäläinen, and P Golland. A distributed spatio-temporal EEG/MEG inverse solver. *Neuroimage*, 44(3):932–46, 2009.

[69] R D Pascual-Marqui, C M Michel, and D Lehmann. Low resolution electromagnetic tomography: a new method for localizing electrical activity in the brain. *Int J Psychophysiol*, 18(1):49–65, 1994.

[70] K Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 6*, 2(11):559–702, 1901.

[71] K. B. Petersen and M. S. Pedersen. *The Matrix Cookbook.* Technical University of Denmark, 2008.

[72] S M Plis, J S G, S C Jun, J Paré-Blagoev, D M Ranken, C C Wood, and D M Schmidt. Modeling spatiotemporal covariance for magnetoencephalography or electroencephalography source analysis. *Physical review E, Statistical, nonlinear, and soft matter physics*, 75(1 Pt 1):011928, 2007.

[73] M E Raichle, A M MacLeod, A Z Snyder, W J Powers, D A Gusnard, and G L Shulman. A default mode of brain function. *Proc Natl Acad Sci USA*, 98(2):676–82, 2001.

[74] H E Rauch, F Tung, and C T Striebel. Maximum likelihood estimates of linear dynamic systems. *AIAA journal*, 3(8):1445–&, 1965.

[75] A Schlögl. A comparison of multivariate autoregressive estimators. *Signal processing*, 86:2426–2429, 2006.

[76] U Schmitt, A K Louis, F Darvas, H Buchner, and M Fuchs. Numerical aspects of spatio-temporal current density reconstruction from EEG/MEG data. *IEEE transactions on medical imaging*, 20(4):314–24, 2001.

[77] A Schnabel, M Burghoff, S Hartwig, F Petsche, U Steinhoff, D Drung, and H Koch. A sensor configuration for a 304 SQUID vector magnetometer. *Neurology and Clinical Neurophysiology*, 2004:70, 2004.

[78] A Schönhage and V Strassen. Schnelle multiplikation grosser zahlen. *Computing*, 7(3–4):281–292, 1971.

[79] G Schwarz. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

[80] K Sekihara, F Takeuchi, S Kuriki, and H Koizumi. Reduction of brain noise influence in evoked neuromagnetic source localization using noise spatial correlation. *Physics in medicine and biology*, 39(6):937–46, 1994.

[81] D Sharon, M S Hämäläinen, R B H Tootell, E Halgren, and J W Belliveau. The advantage of combining MEG and EEG: Comparison to fMRI in focally stimulated visual cortex. *Neuroimage*, 36(4):1225–35, 2007.

[82] R H Shumway and D S Stoffer. An approach to time series smoothing and forecasting using EM algorithm. *Journal of time series analysis*, 3(4):253–264, 1982.

[83] S Taulu, J Simola, M Kajola, E N Oy, and F Helsinki. MEG Recordings of DC Fields Using the Signal Space Separation Method (SSS). *Neurology and Clinical neurophysiology*, 2004:35, 2004.

[84] Nelson J Trujillo-Barreto, Eduardo Aubert-Vázquez, and William D Penny. Bayesian M/EEG source reconstruction with spatio-temporal priors. *Neuroimage*, 39(1):318–35, 2008.

[85] M A Uusitalo and R J Ilmoniemi. Signal-space projection method for separating MEG or EEG into components. *Med Biol Eng Comput*, 35(2):135–140, 1997.

[86] K Uutela, M Hämäläinen, and E Somersalo. Visualization of magnetoencephalographic data using minimum current estimates. *Neuroimage*, 10(2):173–80, 1999.

[87] K Uutela, S Taulu, and M S Hämäläinen. Detecting and correcting for head movements in neuromagnetic measurements. *Neuroimage*, 14(6):1424–31, 2001.

[88] B D Van Veen, W van Drongelen, M Yuchtman, and A Suzuki. Localization of brain electrical activity via linearly constrained minimum variance spatial filtering. *IEEE transactions on bio-medical engineering*, 44(9):867–80, 1997.

[89] J Z Wang, S J Williamson, and L Kaufman. Magnetic source images determined by a lead-field analysis: the unique minimum-norm least-squares estimation. *IEEE transactions on bio-medical engineering*, 39(7):665–675, 1992.

[90] J Z Wang, S J Williamson, and L Kaufman. Magnetic source imaging based on the minimum-norm least-squares inverse. *Brain topography*, 5(4):365–371, 1993.

[91] K F K Wong, A Galka, O Yamashita, and T Ozaki. Modeling non-stationary variance in EEG time series by state space GARCH model. *Comput Biol Med*, 36(12):1327–35, 2006.

[92] K Xu and C Wikle. Estimation of Parameterized Spatio-Temporal Dynamic models. *Journal of Statistical Planning and Inference*, 2007.

[93] J E Zimmerman. SQUID instruments and shielding for low-level magnetic measurements. *Journal of Applied Physics*, 48(2):702–710, 1977.

[94] J Zumer, H Attias, and K Sekihara. A probabilistic algorithm integrating source localization and noise suppression for MEG and EEG data. *Neuroimage*, 37(1):102–115, 2007.