Stochastic techniques for modelling brain connectivity in neuroimaging

Pedro Eduardo Andrade Santos Osório

Dissertação para obtenção do Grau de Mestre em Engenharia Biomédica

Júri
Presidente: Prof Fernando Henrique Lopes da Silva
Orientadora: Prof Patrícia Margarida Piedade Figueiredo
Orientador: Prof Carlos Jorge Ferreira Silvestre
Vogal: Prof João Manuel Lage de Miranda Lemos
Vogal: Dr Paulo André Nobre Rosa

Novembro 2011
Acknowledgments

I must thank my supervisors Patricia Figueiredo and Carlos Silvestre for the time spent guiding me through this work.

I thank Paulo Rosa for being my unofficial supervisor, for all the suggestions and reviewing of my work.

Special thanks to all my friends who encourage me, in particular to my girlfriend Roma Olišauskaitė who never lets me give up and always inspires me to work harder, not only with words but by setting the example.

Finally, I thank my mother for making all this possible.
Abstract

A fundamental question in neuroscience is how different brain areas communicate with each other. Dynamic Causal Modelling (DCM) is a generic formalism to study effective brain connectivity based on neuroimaging data, particularly functional magnetic resonance imaging (fMRI). The interactions between neuronal activity at different brain areas are modelled as a first-order differential equation plus a generative model of how the neuronal activity is transformed into the measured fMRI signal - through the haemodynamic response function (HRF). The problem of discriminating between different structures of connectivity can be solved by state-of-the-art Bayesian methods, whereby parameter estimation is followed by model selection. These methods are computationally expensive and may converge to parameter values that are valid only for a particular dataset. In this thesis, the well-known multiple model kalman filter (MMKF) is applied in a number of different ways to perform model selection in an efficient manner, upon simulated data from models with different connectivity structures. The problem of estimating the HRF is first addressed and the methodology is then extended to the full problem of estimating the DCM. The results show that the MMKF seems to be very accurate at choosing the correct connectivity structure between two parametrized models, even taking into account the output non-linearity (the HRF). In conclusion, this work provides the first demonstration of the applicability of MMKF approaches to the problem of estimating effective brain connectivity based on DCM for fMRI.

Keywords: fMRI, Dynamic Causal Modelling, Multiple Model Estimation, Kalman Filter
Resumo

Uma questão fundamental em neurociência é a forma como diferentes áreas do cérebro comunicam entre si. Dynamic causal modelling (DCM) é um formalismo para estudar a conectividade efectiva do cérebro, particularmente ressonância magnética funcional (fMRI). As interações da atividade neuronal em diferentes áreas cerebrais são modeladas por uma equação diferencial de primeira ordem e um modelo da conversão da actividade neuronal no sinal MRI medido - através da resposta hemodinâmica (HRF). A discriminação entre diferentes estruturas de conectividade pode fazer-se usando métodos Bayesianos recentes, em que a estimação de parâmetros precede a seleção de modelos. Estes métodos são dispendiosos a nível computacional e podem convergir para valores de parâmetros que só são válidos para um conjunto de dados particular. Nesta tese, o filtro Kalman multi-modelo (MMKF) é usado para seleção de modelos eficiente em dados simulados por modelos com conectividades diferentes. O problema de estimar a HRF é tratado em primeiro lugar e a metodologia é alargada à estimação do DCM. Os resultados mostram que o MMKF parecer ser bastante preciso na escolha da estrutura de conectividade entre dois modelos parametrizados, mesmo tendo em conta a não linearidade da saída (a HRF). Em conclusão, este trabalho fornece a primeira demonstração da aplicabilidade de abordagens MMKF ao problema de estimar a conectividade efectiva do cérebro com base no DCM para fMRI.

Palavras-Chave: fMRI, Dynamic Causal Modelling, Estimação Multi-Modelo, Filtro de Kalman
# Contents

1 Introduction 1

1.1 Brain Connectivity 2

1.1.1 DCM for fMRI 3

1.1.2 Estimating Effective Brain Connectivity with DCM 4

1.2 Multiple-Model Estimation 7

1.2.1 Multiple-Model Kalman Filter 8

1.2.1.1 Kalman Filter 8

1.2.1.2 The multiple-model 9

1.3 Organization of the Thesis 11

1.4 Contributions 11

2 Haemodynamic Response Function Estimation 13

2.1 Introduction 14

2.2 Methods 15

2.2.1 Multiple-Model Kalman Filter 15

2.2.1.1 Bilinearization 15

2.2.1.2 Discretization 16

2.2.2 Parameters and Preprocessing 17

2.2.3 Simulations 19

2.2.4 Empirical data 19

2.3 Results and discussion 20

2.3.1 Simulations 20

2.3.2 Empirical data 20

3 Dynamic Causal Model Estimation 25

3.1 Introduction 26

3.2 Methods 26
List of Figures

1.1 Dynamic causal modelling ........................................... 3
1.2 The haemodynamic response function .............................. 5

2.1 The output of 6 different haemodynamic response functions to a unitary input ... 18
2.2 Probabilities in HRF simulation, SNR=15 .......................... 21
2.3 Probabilities in HRF simulation, SNR=5 .......................... 21
2.4 Probabilities in HRF simulation, SNR=1 .......................... 22
2.5 Probabilities for visual task ......................................... 22
2.6 Best model fit for visual task ...................................... 23
2.7 Probabilities for motor task ........................................ 23
2.8 Best model fit for motor task ...................................... 24

3.1 DCM Models differing in intrinsic connectivity ....................... 28
3.2 Input signals for the intrinsic connectivity scenario .................. 28
3.3 DCM Models differing in modulating connectivity ...................... 29
3.4 Input signals for the modulating connectivity scenario ............... 29
3.5 Probabilities in intrinsic scenario, when the true model is number 2 ........ 34
3.6 Probabilities in modulating scenario when the true model is number 2 ........ 34
3.7 Intrinsic scenario, 50s of data, correct model 1 ....................... 35
3.8 Intrinsic scenario, 200s of data, correct model 1 .................... 35
3.9 Intrinsic scenario, 50s of data, correct model 2 ....................... 36
3.10 Intrinsic scenario, 200s of data, correct model 1 ...................... 36
3.11 Modulating scenario, 50s of data, correct model 1 .................... 37
3.12 Modulating scenario, 200s of data, correct model 1 ................... 37
3.13 Modulating scenario, 50s of data, correct model 2 .................... 38
3.14 Modulating scenario, 200s of data, correct model 2 ................... 38
3.15 Classification score for different input characteristics ............... 39
3.16 Intrinsic scenario, parameter space covering, 50s of data, correct model 1 ........ 40
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.17</td>
<td><em>Intrinsic scenario</em>, parameter space covering, 200s of data, correct model 1</td>
<td>41</td>
</tr>
<tr>
<td>3.18</td>
<td><em>Intrinsic scenario</em>, parameter space covering, 50s of data, correct model 2</td>
<td>41</td>
</tr>
<tr>
<td>3.19</td>
<td><em>Intrinsic scenario</em>, parameter space covering, 200s of data, correct model 2</td>
<td>42</td>
</tr>
<tr>
<td>3.20</td>
<td><em>Intrinsic scenario</em>, efficient parameter space covering, 50s of data, correct model 1</td>
<td>43</td>
</tr>
<tr>
<td>3.21</td>
<td><em>Intrinsic scenario</em>, efficient parameter space covering, 200s of data, correct model 1</td>
<td>43</td>
</tr>
<tr>
<td>3.22</td>
<td><em>Intrinsic scenario</em>, efficient parameter space covering, 50s of data, correct model 2</td>
<td>44</td>
</tr>
<tr>
<td>3.23</td>
<td><em>Intrinsic scenario</em>, efficient parameter space covering, 200s of data, correct model 2</td>
<td>44</td>
</tr>
<tr>
<td>3.24</td>
<td><em>Intrinsic scenario</em>, efficient parameter space covering, 200s of data, correct model 1, for several levels of SNR</td>
<td>45</td>
</tr>
<tr>
<td>3.25</td>
<td><em>Intrinsic scenario</em>, efficient parameter space covering, 200s of data, correct model 2, for several levels of SNR</td>
<td>45</td>
</tr>
<tr>
<td>A.1</td>
<td>Inputs</td>
<td>62</td>
</tr>
</tbody>
</table>
List of Tables

2.1 Biophysical parameters for the 6 HRF models considered . . . . . . . . . . . . . . . 17

3.1 Evidence Labels . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31

3.2 Performance score (normalized to 100) on the modulating scenario . . . . . . . . 46

3.3 Performance score (normalized to 100) on the intrinsic scenario . . . . . . . . . 47
# List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC</td>
<td>Bayesian model comparison</td>
</tr>
<tr>
<td>BOLD</td>
<td>blood oxygenation level dependent</td>
</tr>
<tr>
<td>CBF</td>
<td>cerebral blood flow</td>
</tr>
<tr>
<td>DCM</td>
<td>Dynamic causal modelling</td>
</tr>
<tr>
<td>EEG</td>
<td>electroencephalography</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>FIR</td>
<td>finite impulse response</td>
</tr>
<tr>
<td>fMRI</td>
<td>functional magnetic resonance imaging</td>
</tr>
<tr>
<td>GCM</td>
<td>Granger causality mapping</td>
</tr>
<tr>
<td>GLM</td>
<td>general linear model</td>
</tr>
<tr>
<td>HRF</td>
<td>haemodynamic response function</td>
</tr>
<tr>
<td>KF</td>
<td>Kalman filter</td>
</tr>
<tr>
<td>LCTV</td>
<td>linear continuous time-varying</td>
</tr>
<tr>
<td>LDTV</td>
<td>linear discrete time-varying</td>
</tr>
<tr>
<td>MEG</td>
<td>magnetoencephalography</td>
</tr>
<tr>
<td>MRI</td>
<td>magnetic resonance imaging</td>
</tr>
<tr>
<td>MMKF</td>
<td>multiple-model kalman filter</td>
</tr>
<tr>
<td>SISO</td>
<td>single-input single-output</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to Noise Ratio</td>
</tr>
</tbody>
</table>
SVO Set-valued observer

TR repetition time
Introduction

Contents

1.1 Brain Connectivity .................................................. 2
  1.1.1 DCM for fMRI .................................................... 3
  1.1.2 Estimating Effective Brain Connectivity with DCM .......... 4

1.2 Multiple-Model Estimation ........................................... 7
  1.2.1 Multiple-Model Kalman Filter ................................... 8

1.3 Organization of the Thesis ......................................... 11

1.4 Contributions ....................................................... 11
1.1 Brain Connectivity

The first interpretation of brain connectivity comes from the connections that neurons establish among themselves. Since a neuron serves little purpose when isolated and is able to easily communicate with other neurons when connected through synapses, by means of action potentials (electrochemical pulses that propagate along a neuron), it is natural to look at the brain as a network of these elementary units. The mapping of all these networks in the healthy human brain is known as the human connectome, in the same light as the human genome and proteome, and its completion would be a landmark in the development of science.

Brain connectivity is, however, a concept that is inherently ill-defined [1]. Since the end of the nineteenth century, the brain attracted the work of famous anatomists such as Cajal [2], that studied the structure of the brain at the micro level, thus proposing the theory that neurons are indeed discrete components that form the basis of our brain. This was a fundamental contribution for the initial development of modern neuroscience. It was one step in the direction of the anatomical/structural view of brain connectivity. That interpretation is indeed a mapping of how the neurons are physically organized in the brain. There are still several groups studying this mapping as it has not been completed for human beings [3].

On the other hand, functional connectivity cannot be observed in a strict sense, because it does not occur at the level of neuron-to-neuron interaction. Instead, it is a statistical concept, where the brain activity in certain areas is related to the activity in other areas in a mathematical sense.

A related concept is that of effective connectivity, where it is explicit that the activity at a certain brain region directly causes activity at another one. There is, therefore, the concept of causality involved.

This thesis focuses on the estimation of effective connectivity. There are two types of popular approaches for the estimation of such connectivity: the model-driven and the data-driven ones. In the case of the model-driven approaches, there is an underlying model proposed for how the activity and interaction between brain areas occurs, such as Dynamic causal modelling (DCM) [4], whereas in the data-driven techniques measures of causality between time-series are employed, such as Granger causality mapping (GCM) [5].

In more detail, there are a number of different options when modelling effective connectivity, reviewed in [6]. The choice of the observation equation (how brain states translate to a measured signal) can range from direct output of the states [7], with the addition of noise [8], spatial smearing [9], linear haemodynamic response function (HRF) [10] and non-linear HRF [11] [12]. There are also several ways to model the evolution of brain states in time for effective connectivity, linear and non-linear GCM [13] [14], bilinear, non-linear and hierarchical DCM [4] [15] [16], neural mass models [17] and neural fields [18] [12].

This thesis will focus on model-driven estimation of effective connectivity, more specifically DCM.
1.1.1 DCM for fMRI

DCM is the most widely used model-driven approach to effective connectivity estimation [19]. In figure 1.1 an example of a DCM with five brain areas is displayed. These have (directed) connections between them, determining which areas influence the activity of others. These reflect the latent or intrinsic connectivity and may be uni- or bi-directional, representing feedforward and feedbackward influences between regions. There are two types of inputs: those that directly impact on the activity of a specific brain area, such as sensory stimuli, ($u_1$) and those that modulate the connections between different brain areas, such as a cognitive state like attention, ($u_2$).

The activity of each brain area cannot be directly observed, and a specific neuroimaging technique must be used to obtain measures which can be related to it. The function that describes the measured output of the system as a function of the underlying neuronal activity is called the forward model (it explains how observations are synthesized from the underlying hidden neural activity, as opposed to the inverse model, which would produce the neural activity given the observed signal).

The most commonly used techniques used to measure brain activity in humans are electroencephalography [EEG] / magnetoencephalography [MEG], which non-invasively record electric/magnetic field changes on the scalp, and fMRI, which is sensitive to variations of the blood oxygenation in re-
response to neuronal activity.

More than a century ago, Roy and Sherrington [20] first hinted at the connection between neuronal activity and cerebral blood flow (CBF), as well as blood oxygenation in the brain. One hundred years later, in 1990, Ogawa et al. [21] proposed the first non-invasive method for measuring cerebral blood oxygenation levels. It relies on the usage of specific magnetic resonance imaging (MRI) pulse sequences that increase the effects of paramagnetic deoxyhemoglobin, resulting in what is known as the blood oxygenation level dependent (BOLD) contrast. This method, combined with the fact that neuronal activity influences cerebral blood flow and oxygenation levels, led to its application as the first functional imaging method for the brain, the functional magnetic resonance imaging (fMRI).

In order to make inferences about the underlying neuronal activity responsible for the fMRI signal, it is crucial to have a quantitative description of the neurovascular coupling responsible for the latter. This is the so-called HRF. In 1994, a framework was proposed [22], wherein the HRF is modelled as a finite impulse response (FIR). In order to find correlations between a tentative neuronal activity signal (contrast) and the observed BOLD signal, the linear convolution of the former with the proposed HRF is compared with the latter. This was the first HRF based treatment of MRI time-series, later expanded as the general linear model (GLM) [23], which is still used frequently by the MRI community.

Despite the widespread acceptance of GLM and linear convolution to model BOLD signals, the neurovascular coupling cannot be accurately modelled in this way, as this dynamical system exhibits marked nonlinearities. This is especially important if the neuronal activity signal is composed of short and frequent impulses that happen in rapid event-related designs [24]. In 1998, Buxton proposed the Balloon Model [25], a nonlinear dynamical system which relates blood flow (f) with the BOLD signal (y), having volume (v) and deoxyhemoglobin content (q) as state variables. In 2000, Friston et al. [11] complemented this model (see Figure 1.2) by expressing the blood flow as a function of the synaptic activity (u), adding another state variable to the system: the flow-inducing signal (s).

1.1.2 Estimating Effective Brain Connectivity with DCM

Since the original paper, in 2003, describing DCM [4], there have been a number of developments that allow for DCM to estimate brain connectivity more accurately and in a wider range of experimental settings. First, the specification of model comparison which is essential to distinguish between different brain connectivity structures [26]. Then, in 2006, it was extended to EEG and MEG through the use of neural mass models [27]. In 2007, there were modifications to the HRF equations to account for better knowledge of the biophysical and MRI acquisition parameters [28][29]. After that, came improvements to the DCM state equations, first with the modelling of inhibitory and excitatory subpopulations on each brain region [30] and then non-linear DCM which

---

1Hemoglobin without bound oxygen
The hemodynamic model

neuronal input

state variables
\[ x = \{ z, s, f, v, q \} \]

activity-dependent signal
\[ \dot{s} = z - \kappa s - \gamma(f - 1) \]

flow induction
\[ \dot{f} = s \]

changes in volume
\[ \tau v = f - v^{1/\alpha} \]

changes in dilb
\[ \tau q = f E(f, \rho)/\rho - v^{1/\alpha} q/v \]

hemodynamic response

\[ y = \lambda(v, q) \]

Figure 1.2: The haemodynamic response function, taken from [4]
allowed the modelling of cases where the activity of a brain region modulates the connectivity between two other regions of the brain [15]. In 2009, the framework was extended to steady-state responses, that is, for data acquired with no designed experimental brain stimulation [31].

Estimating effective brain connectivity with DCM requires the estimation of which models are superior to others in explaining the data. This usually involves inversion of the model. For this reason, techniques that deal with the inversion of the observation model (the HRF), are important. Apart from the one by Friston [11], alternatives have been proposed based on the extended Kalman filter [32], sequential Monte Carlo methods [33] and the unscented Kalman filter [34] as well as multiple-model Set-valued observer (SVO) [35].

Regarding the estimation of brain connectivity, the typical process based on DCM follows a two-step approach:

- A number of plausible models are proposed and the parameters of these models are estimated using a variational Bayes algorithm [36][37]. In the process, not only the neural connectivity but also the forward model parameters are estimated.

- These models are assessed using Bayesian model comparison (BMC) [38], and the best one is chosen.

In BMC, one has a set of models $\mathcal{M} = \{M_1, ..., M_N\}$, the i-th model is parametrised by a vector $\theta^{(i)}$ and assigns a probability to the observed data $y$, given its parameters, $p(y|M_i, \theta^{(i)})$. The goal is to estimate the probability of each model given this data, for any set of parameters. Using Bayes’ theorem:

$$p(M_i|y) \propto p(y|M_i) \cdot p(M_i) \quad (1.1)$$

In the special case where there is equal a priori probability for all models ($\forall i \ p(M_i) = \frac{1}{N}$), the probability of each model given the data is proportional to the marginal likelihood (also known as the evidence for the model), $p(y|M_i)$. It’s then possible to use quotients of these terms for models $i, j$, known as Bayes factors, to compare the evidence for them:

$$BF_{ij} = \frac{p(y|M_i)}{p(y|M_j)} \quad (1.2)$$

The evidence for a model can be computed:

$$p(y|M_i) = \int p(y|M_i, \theta^{(i)}) p(\theta^{(i)}|M_i) \ d\theta^{(i)} \quad (1.3)$$

However, (1.3) is analytically integrable only in toy problems and numerical evaluation doesn’t provide good results for high-dimensional parameter spaces.

An approximation to the logarithm of (1.3) (known as log-evidence) can be obtained by using a variational approach. The idea is to define $q$, a distribution over the parameters that is supposed
to approximate \( p(\theta^{(i)}|y, M_i) \), which can be decomposed into factors (mean-field approximation):

\[
q(\theta^{(i)}) = \prod_j q_j(\theta^{(i)}_j)
\]  

(1.4)

Where each factor is approximated by its first and second moments (as a gaussian distribution) using the Laplace approximation. In this situation, the log-evidence is:

\[
\log p(y|M_i) = F(q) + D_{KL}(q(\theta^{(i)}):p(\theta^{(i)}|y, M_i))
\]  

(1.5)

\[
F(q) = \langle \log(p(\theta^{(i)}|M_i) + \log(p(y|\theta^{(i)}, M_i)) \rangle_q + \langle \log(q(\theta^{(i)})) \rangle_q
\]  

(1.6)

Where (1.5) separates the log-evidence of the model into a free energy term, \( F(q) \) and the Kullback-Leibler divergence \( D_{KL} \), between the distribution \( q \) over the parameters and the true distribution of the parameters for this model, given the data. (1.6) defines the free energy.

Since the Kullback-Leibler divergence measures the information divergence between two distributions and it is nonnegative, maximizing \( F(q) \) with respect to \( q \), is a way of finding a lower-bound to the log-evidence. Furthermore, maximizing \( F(q) \) minimizes the divergence between the approximated distribution \( q \) and the real one.

With (1.2), BMC can use these log-likelihoods to compare models. Since a lower-bound is used (and the tightness of the bound is different for different models and not available), the model comparison is not as reliable as it could be and this is admitted as a weakness of the comparison process by the authors [40].

This estimation process was extended to group studies [41]. Network connectivity was only recently explored from the point of view of discovery rather than selection from a set of a few models in [42], which claims to be able to explore the structure of a connectivity network in an efficient manner, by evaluating the log-evidence of a fully connected model, and using it to approximate those of other connectivity structures, thus pruning those that do not agree with the data. Also, recently, two new approaches to the DCM model inversion using generalised filtering [43] or the cubature Kalman filter [44], which provide time-varying estimates of the model parameters unlike the variational Bayesian method, and are also capable of dealing with steady-state data.

### 1.2 Multiple-Model Estimation

Under measurement errors, typical state estimation techniques require the exact specification of the model parameters. When these are not available due to model uncertainty, it is not possible to build a state observer through classical estimation tools. [45] [46] [47]

A solution which is more robust to parameter uncertainty is the multiple-model estimation [48]. For problems where it is impossible to determine \( a \) priori the parameters of a dynamic system,
and in applications that require an accurate state estimate (for example, control [49] [50] [51]) there may be the need to develop several models of the target system, whose parameters form a parameter set.

Specifically for the case of estimating connectivity structure using DCM, given that the underlying biology is a lot more complex, it may just be that the neural model is too coarse resulting, among other things in a posterior distribution of the parameters that is clearly multi-modal. Since the standard DCM estimation approach tries to estimate the optimal parameters for a given connectivity structure with a product of Gaussian distributions that maximize the free energy, if the true conditional distribution is actually multi-modal and the mean-field assumption doesn’t hold, it will result in a lower bound whose of the model log-evidence whose tightness is unknown and thus compromise the model comparison. If, on the other hand, one starts with the assumption that several parameter configurations may be correct, these issues may be obviated and thus allow for a more robust model selection in the context of effective connectivity estimation.

Instead of the standard fitting of one model, several models representative of the possible (parameter) states of the system can be used. The states of each of these models can be estimated by different types of observers, ranging from Kalman filters (KF) [52] to SVOs [53] [54] [55] which are based on model falsification instead of model selection and have been applied to the problem of estimating the correct HRF for fMRI [35] [56].

For the reasons above, multiple-model estimation seems like a good idea for tackling the problem of identifying network connectivity in DCM. Given the choice between several multiple-model estimation algorithms, this thesis focuses on the multiple-model kalman filter (MMKF), the multiple-model counterpart of KF, a very well studied filter, known to be the minimum variance unbiased estimator for linear systems, which has also been applied (with modifications) extensively as a non-optimal filter to non-linear systems.

1.2.1 Multiple-Model Kalman Filter

1.2.1.1 Kalman Filter

Since the seminal paper by Kalman [52], the filter with the author’s name has been deployed successfully as a tool for state estimation in many areas of engineering and science. The KF requires a state space description of a discrete-time (or continuous-time) dynamical system:

\[
\begin{align*}
x[k+1] &= A[k]x[k] + B[k]u[k] + w[k] \\
y[k] &= C[k]x[k] + \epsilon[k]
\end{align*}
\]  

(1.7) (1.8)

In (1.7) \(x[k]\) is the vector of state variables at time \(k\) and \(u[k]\) is the vector of inputs to the system. The evolution of the state is governed by the matrices \(A[k]\) and \(B[k]\), which are time-dependent, so
the system is referred to as linear discrete time-varying (LDTV). Furthermore, the state changes need not be deterministic and the stochastic part is taken to be a zero-mean, normally distributed process, white and uncorrelated with the state variables and inputs, \( w[k] \sim \mathcal{N}(0, Q[k]) \).

Eq. (1.8) refers to the measurement process. Matrix \( C[k] \) determines the linear mixtures of state variables that are observed. Similarly to the state evolution, the measurement process is not deterministic, and origins noisy observations. The noise is also assumed to be zero-mean, normally distributed and uncorrelated with the state and inputs, \( \epsilon[k] \sim \mathcal{N}(0, R[k]) \) and white.

The KF expresses the states as (multivariate) normally distributed random variables. One can aid the KF to a faster convergence by providing accurate prior beliefs about the state variables. This is done by defining their distribution before the first measurement as \( x[0] \sim \mathcal{N}(\hat{x}[0], P[0]) \).

Essentially, \( \hat{x}[0] \) is the best possible estimate prior to the measurements of the state variables and \( P[0] \) the uncertainty about each of them (diagonal terms) and covariance between different state variables (off-diagonal terms).

At each time \( k \), the KF predicts the new distribution of the state variables after one time step by applying directly (1.7). Given that these variables have a multivariate normal distribution, this can be done analytically.

After receiving the measurement at time \( k+1 \), the previous distribution is updated. In a Bayesian framework, this involves computing the posterior distribution of the state variables, where the prior distribution is \( \mathcal{N}(\hat{x}[k+1|k], P[k+1|k]) \) and the likelihood given the measurement is also normally distributed and determined by (1.8). Since the prior and the likelihood are both gaussian, this can once more be computed exactly and is called the update step.

The difference between the measurement and the predicted measurement, \( r[k+1] \), is called residual or innovation. \( S[k+1] \) is the covariance matrix of the residual and \( K[k+1] \) the optimal kalman gain which is multiplied by the residual to obtain the updated mean of the state variables.

### 1.2.1.2 The multiple-model

The KF is optimal for a well-known dynamic model. In real-world scenarios, however, there is always uncertainty about the parameters of the model and they may also vary with time. For a given KF, the true parameters may vary within a small range around the design parameters and still allow for a correct estimation. One technique proposed in the control literature (see [57]) is to discretize the parameter space and have a set of kalman filters, each one corresponding to a different model, a point in that space. A probability distribution over the models is kept and the robust estimate is a sum of the state estimates of each KF weighted by the probability distribution.

This is appropriately called MMKF [58] [59] [60].

An interesting point about the MMKF is that despite being primarily a tool for state estimation, it inherently performs model selection (through the probability distribution over the set of models).

Explicitly, for an MMKF with \( n \) models - \( \mathcal{H}_1 \cdots \mathcal{H}_n \) - if the full data up to time \( k \) is defined as:
Where model $m$ has an assigned probability at time $k$:

$$P_m[k] \equiv P(H = H_m|Y[k]) \quad (1.10)$$

Then, the state estimate yielded by the MMKF is a weighted average of all estimates:

$$\hat{x}[k|k] = \sum_{i=1}^{n} P_i[k] \hat{x}_i[k|k] \quad (1.11)$$

The prior probability over the models must be defined (typically a uniform distribution, if there’s no reason to believe that some regions of the parameter space are more relevant a priori). All that is needed is a recursive formula that allows the computation of the probability of each model at time $k+1$ given the probabilities at time $k$:

$$P_i[k+1] = \frac{p(y[k+1]|H_i, u[k], Y[k]) \cdot Z}{P_i[k]} \quad (1.12)$$

(1.12) is a direct application of Bayes’ rule and $Z$ is a normalizing constant.

It can be shown that defining:

$$\beta_i[k] \equiv \frac{1}{(2\pi)^{m/2} \sqrt{det S_i[k]}} \quad (1.13)$$

$$w_i[k] \equiv r_i^T[k] S_i^{-1}[k] r_i[k] \quad (1.14)$$

(1.12) can be rewritten as:

$$P_i[k+1] = \frac{B_i[k+1] e^{-\frac{1}{2} w_i[k+1]}}{Z} \cdot P_i[k] \quad (1.15)$$

The fundamental assumption is that one of the models in the MMKF corresponds to the true model (otherwise it wouldn’t be possible to define a probability function of correctness over the models - the probability that the correct model does not belong to the selected set would be nonzero, the function wouldn’t sum to one and hence it wouldn’t be a probability distribution). In this case, the MMKF converges to the true model given a reasonable level of noise.

However, even if the true model is not in the set considered in the MMKF (which will happen almost surely in a continuous parameter space), this approach still has its merits and it will converge almost surely to the model that is closest to the real system in terms of the measure defined by Baram.

---

2Where $m$ is the number of state variables
1.3 Organization of the Thesis

As a preamble, in Chapter 2 the MMKF is tested in the problem of selecting between a number of HRF models (that is, sets of parameters), and the required adaptations to the HRF are made. In Chapter 3, the modifications of the HRF are incorporated to make the DCM suitable for the MMKF and number of algorithms for selecting the model with the correct structure using MMKF are developed and tested.

In Chapter 4, the conclusions and suggestions for future work are presented.

1.4 Contributions

The main contribution of this thesis is the application of multiple-model estimation to the problem of effective brain connectivity estimation for the first time. More specifically, this thesis tests a new approach to estimate the correct connectivity structure of DCM models which assumes that there isn’t a set of optimal parameters for a given structure through the use of MMKF. The correct estimation of connectivity structures with known parameters is demonstrated and some steps in the direction of structure estimation without parameters are also accomplished. The subproblem of choosing between a set of different HRF is also tackled with success.
Haemodynamic Response Function Estimation

Contents

2.1 Introduction ................................................. 14

2.2 Methods ..................................................... 15

2.2.1 Multiple-Model Kalman Filter ......................... 15

2.2.2 Parameters and Preprocessing ......................... 17

2.2.3 Simulations .............................................. 19

2.2.4 Empirical data ........................................... 19

2.3 Results and discussion ................................... 20

2.3.1 Simulations ............................................. 20

2.3.2 Empirical data ........................................... 20
2.1 Introduction

The set of equations that comprise the HRF model (Figure 1.2) is:

\[
\dot{s} = \epsilon u - k_s s - k_f (f - 1) \quad (2.1a)
\]

\[
\dot{f} = s \quad (2.1b)
\]

\[
\dot{v} = \frac{1}{\tau} (f - v^{\alpha}) \quad (2.1c)
\]

\[
\dot{q} = \frac{1}{\tau} \left( f \frac{1 - (1 - E_0)^{\frac{1}{2}}}{E_0} - v^{(\frac{1}{2} - 1)} q \right) \quad (2.1d)
\]

\[
y = V_0 [k_1 (1 - q) + k_2 (1 - \frac{q}{v}) + k_3 (1 - v)] \quad (2.1e)
\]

Recapitulating the definitions from the introduction, the HRF is the nonlinear dynamical system described by the equations above, and relating blood flow \((f)\) with the BOLD signal \((y)\). Its state variables are the volume \((v)\) and deoxyhemoglobin content \((q)\). Friston [11] complemented this model by expressing the blood flow as a function of the synaptic activity \((u)\), adding another state variable to the system: the flow-inducing signal \((s)\). All of these except for the flow-inducing signal (Hz) are normalized with respect to the resting state.

Eq. (2.1a), denotes the time-derivative of the flow-inducing signal, which depends on:

1. The neuronal activation, whose influence is mediated by the efficacy parameter \(\epsilon\)

2. The flow-inducing signal decay with its associated time constant \(k_s\)

3. The autoregulatory feedback from the blood flow with time constant \(k_f\)

Eq. (2.1b) is the simplest way to say that the blood flow is determined by the flow-inducing signal. Eq. (2.1c) models the evolution of the volume as the difference between the incoming flow and the outgoing flow, where the outgoing flow is \(v^{\frac{1}{\alpha}}\). This makes explicit the balloon-like behaviour of the blood vessel, as the outflow depends on the existing blood volume in the compartment (the distended balloon expels more blood) and a parameter, the Grubb’s exponent \((\alpha)\), that governs the stiffness of said balloon. \(\tau\) is the mean transit time through the blood vessel.

Eq. (2.1d) has the greatest complexity among the 5. The deoxyhemoglobin content change is equal to the amount of this substance delivered to the vein minus the quantity that is taken out. The latter is simply the outflow times the deoxyhemoglobin concentration. The former is the incoming flow times the normalized fraction of oxygen extracted from incoming blood.

The fraction of oxygen extracted \((E(f, E_0))\) decreases with the incoming blood flow, because it depends on the total amount of oxygen delivered.

Finally, the rate of deoxyhemoglobin content change also depends on the transit time \(\tau\).
Eq. (2.1e) expresses the output as a nonlinear function of deoxyhemoglobin content and blood volume.

Estimating the parameters of the HRF is not trivial as it is a highly nonlinear system, with parameter identifiability issues (due to the number of parameters that need to be estimated from a single-input single-output (SISO) system). The low sensitivity of the output to local variation in the parameters has been previously shown [62] and justifies an approach, such as the MMKF, where the correct HRF is chosen from a set of models representative of the different physiologically plausible HRFs.

Given a set of different models, all described by (2.1), but with different parameter values, where the input to the system is assumed to be known (corresponding to the fMRI experiment design), the goal is to determine which one is the closest to the true HRF.

## 2.2 Methods

In the following subsections the methods required to apply MMKF to the selection of an HRF are described. In 2.2.1, the conversion (discretization and bilinearization) of the HRF to a system that is mathematically compatible with the MMKF, then in 2.2.2, the different HRF models are presented as well as three preprocessing steps: the estimation of neural efficacy for each model and offset of the data, followed by the estimation of the Signal to Noise Ratio (SNR). In 2.2.3 the computer simulations are described and in 2.2.4 the application of the method to empirical data.

### 2.2.1 Multiple-Model Kalman Filter

In order to accomplish this, the MMKF will be used with some modifications since the HRF is neither linear nor discrete in time. Hence, the required adjustments are:

- (Bi-)Linearization
- Discretization

#### 2.2.1.1 Bilinearization

The bilinearization was first proposed in [11], the same paper that formulated the full model, and has been repeatedly used since then. Defining:

\[
x = [s, f, v, q]
\]

\[
x = F(x, \theta, u)
\]

\[
y = \lambda(x)
\]
The bilinear approximation to the function F can be found by using Taylor’s formula around \( x = x^* \) and \( u = 0 \):

\[
\dot{x} \approx F(x^*, \theta, 0) + \frac{\partial F(x^*, \theta, 0)}{\partial x}(x - x^*) + u \left( \frac{\partial^2 F(x^*, \theta, 0)}{\partial x \partial u}(x - x^*) + \frac{\partial F(x^*, \theta, u)}{\partial u} \right) \tag{2.5}
\]

Noting that the function F is linear on u, \( \frac{\partial^2 F(x^*, \theta, 0)}{\partial x \partial u} \) must be 0, and the above can be rewritten as:

\[
\dot{x} \approx \left( F(x^*, \theta, 0) - \frac{\partial F(x^*, \theta, 0)}{\partial x} \cdot x^* \right) + \frac{\partial F(x^*, \theta, 0)}{\partial x} \cdot x + u \cdot \frac{\partial F(x^*, \theta, u)}{\partial u} \tag{2.6}
\]

With the above reordering of the terms, it becomes obvious that (with a simple variable change) F is bilinear in \( x \) and \( u \), and therefore a matrix form of this approximation can be obtained as follows:

\[
\begin{align*}
\ddot{x} & \equiv [1 \ x]^T \tag{2.7} \\
A & \equiv \begin{bmatrix} 0 & 0 \\
F(x^*, \theta, 0) - \frac{\partial F(x^*, \theta, 0)}{\partial x} \cdot x^* & \frac{\partial F(x^*, \theta, 0)}{\partial x} \end{bmatrix} \tag{2.8} \\
B & \equiv \begin{bmatrix} 0 \\
\frac{\partial F(x^*, \theta, u)}{\partial u} \end{bmatrix} \tag{2.9} \\
\dot{x} & = A\ddot{x} + uB\ddot{x} = (A + uB)\ddot{x} \tag{2.10}
\end{align*}
\]

One must not forget that (2.11), the output, is also nonlinear. Since the final system may be time-varying, as the MMKF allows it, its linearization proceeds in a straightforward fashion:

\[
y = \left[ \lambda(x) - \frac{\partial \lambda(x)}{\partial x} \cdot x \quad \frac{\partial \lambda(x)}{\partial x} \right] \cdot \ddot{x} \tag{2.11}
\]

The equations (2.10) and (2.11) constitute a linear continuous time-varying LCTV system, because A, B, C and u depend on time, and it is now a step closer to being suitable for MMKF treatment.

### 2.2.1.2 Discretization

Perhaps more important than the linearization of the model is the discretization, as this step is required to implement any model in a digital computer. After a discretization period \( \Delta T \) is chosen, it is assumed that the continuous time-varying matrices and the input remain constant during this sampling interval:

\[
A[k] = A(t) + B(t)u(t), \ t \in [k\Delta T, k\Delta T + \Delta T] \quad \forall k \tag{2.12}
\]
One can then integrate (2.10) during this discretization step, to obtain:

\[ \tilde{x}(k\Delta T + \Delta T) = e^{A[k] \Delta T} \tilde{x}(k\Delta T) \quad (2.13) \]

\[ \tilde{x}[k + 1] = \tilde{A}[k] \tilde{x}[k] \quad (2.14) \]

The continuous output equation (2.11) can be directly discretized:

\[ C[k] = \begin{bmatrix} \lambda(x[k]) - \partial \lambda(x[k]) / \partial x \cdot x[k] & \partial \lambda(x[k]) / \partial x \end{bmatrix} \quad (2.15) \]

\[ y[k] = C[k] \tilde{x}[k] \quad (2.16) \]

It can be seen that (2.14) and (2.16) fit perfectly into the LDTV framework expressed in (1.7) and (1.8), and are thus finally in a form that allows the use of MMKF.

### 2.2.2 Parameters and Preprocessing

A set of 6 physiologically plausible HRF models are used that display the stereotypical known features of the HRF, specifically: fast, medium or slow response with or without undershoot. The parameters of these models are in the table 2.1.

<table>
<thead>
<tr>
<th>HRF</th>
<th>(k_s)</th>
<th>(k_f)</th>
<th>(\tau)</th>
<th>(\alpha)</th>
<th>(E_0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.400 s(^{-1})</td>
<td>0.100 s(^{-1})</td>
<td>2.080 s</td>
<td>0.320</td>
<td>0.340</td>
</tr>
<tr>
<td>2</td>
<td>0.220 s(^{-1})</td>
<td>0.110 s(^{-1})</td>
<td>2.180 s</td>
<td>0.320</td>
<td>0.340</td>
</tr>
<tr>
<td>3</td>
<td>0.650 s(^{-1})</td>
<td>0.200 s(^{-1})</td>
<td>0.880 s</td>
<td>0.320</td>
<td>0.340</td>
</tr>
<tr>
<td>4</td>
<td>0.450 s(^{-1})</td>
<td>0.300 s(^{-1})</td>
<td>1.580 s</td>
<td>0.320</td>
<td>0.340</td>
</tr>
<tr>
<td>5</td>
<td>0.950 s(^{-1})</td>
<td>0.450 s(^{-1})</td>
<td>0.200 s</td>
<td>0.320</td>
<td>0.340</td>
</tr>
<tr>
<td>6</td>
<td>0.750 s(^{-1})</td>
<td>0.600 s(^{-1})</td>
<td>0.500 s</td>
<td>0.320</td>
<td>0.340</td>
</tr>
</tbody>
</table>

Table 2.1: Biophysical parameters for the 6 HRF models considered.

For each of the models, the neuronal efficacy \(\epsilon\) is determined depending on the experimental data (input design and observed BOLD response). The procedure is as follows:

1. Compute a feature from the observed BOLD signal (such as the amplitude of the maximum FFT bin), call it \(F_0\)
2. Initialize \(\epsilon_{min}\) and \(\epsilon_{max}\)
3. Set \(\epsilon = (\epsilon_{min} + \epsilon_{max}) / 2\)
4. Simulate the response of the model with the current \(\epsilon\) to the experimental input
5. Subsample the result at the same sample rate as the experimental data
Figure 2.1: The output of 6 different haemodynamic response functions to a unitary input

6. Compute the same feature as in step (1) on this data, call it F

7. If $|F - F_0| < \text{tolerance}$, accept the current $\epsilon$ and exit the algorithm

8. If $F < F_0$, set $\epsilon_{\text{min}} = \epsilon$, else set $\epsilon_{\text{max}} = \epsilon$

9. Go to step (3)

This is, essentially, a binary search on the parameter $\epsilon$ trying to minimize the absolute difference between a feature computed for the current model and the same feature computed on the observed response.

The HRF models described in table 2.1 and subject to the above explained normalization for a unitary input can be visualized in figure 2.1. These haemodynamic responses differ essentially in the presence of an undershoot and the response time, as can be seen in the figure.

Typically, the BOLD response is obtained as a zero mean signal, although the haemodynamic model assumes the baseline to be 0 and most of the signal to be above that value. This means that the model so far presented would not fit the experimental data as it is. The approach taken in this thesis, consists in assuming that the zero mean signal must be shifted vertically, that is, an offset $v_{\text{off}}$ must be added to the experimental data. In order to find the best value for a given model that has response $y_i$, we can just minimize the sum of squared errors:

$$\hat{v}_{\text{off}} = \arg\min_v \sum_{j=1}^{N} (y_i(j) - (y_{\text{exp}}(j) + v))^2$$  \hspace{1cm} (2.17)

It’s easily seen that the offset that minimizes the sum of squared errors is:
\[ \hat{v}_{\text{off}} = \frac{\sum_{j=1}^{N} (y_{i}(j) - y_{\text{exp}}(j))}{N} \] (2.18)

It is thus possible to obtain a set of these offsets, one for each model. In order to choose the offset that should be applied, one could go for the obvious choice, which is the mean. However, in order to guarantee robustness to outliers (i.e. models that are far away from the real response), the median offset is chosen in this work. Another alternative could be to feed each model with data that had the corresponding offset.

The MMKF requires an estimate of the noise in the measurement model. In order to obtain this estimate, one of the HRFs is used to filter the input for a given experiment and the Fast Fourier Transform (FFT) of the output is then computed. By finding the bins that contain the least amount of signal energy (up to 5% cumulatively), one can use these to detect the frequencies where an output signal from the same experiment is supposed to contain mostly noise and thus estimate the SNR of other data, where the SNR is defined as:

\[ \text{SNR} = \frac{\mu_{\text{signal}}}{\sigma_{\text{signal}}} \] (2.19)

In summary, the data is used to automatically normalize the models, select the offset value and estimate the SNR before entering the MMKF procedure.

2.2.3 Simulations

Data were simulated using the parameters of the sixth model in table 2.1 and using as input signal 300s of a rectangular wave with period 8s and duty cycle of 50%. The integration step is 0.1s but the signal is sampled to get a simulated repetition time (TR) of 3s. These data were corrupted with band limited Gaussian noise with three levels of standard deviation to produce data with SNR of 1, 5 and 15. All preprocessing steps were performed as described above.

2.2.4 Empirical data

Two sets of empirical data collected from different experiments were used to test the model. In the first experiment, visual stimuli lasting 18s alternated with fixation periods with roughly the same duration, for a total of 366s, were used. The value of TR was 3s. After pre-processing of the raw signal, and performing the identification of an activated voxel on the visual cortex with the usual tools (GLM), its corresponding time-series, consiting of 122 points, was used as the experimental data (y) for the model selection procedure.

In the second experiment, the subjects were asked to perform a motor task for 30s followed by rest for the same amount of time repeatedly, leading to a total acquisition time of 5 minutes with a TR of 2s. An activated voxel from the motor cortex was used to obtain the time-series (with 150 points) used in the model selection procedure after the usual preprocessing steps.
2.3 Results and discussion

2.3.1 Simulations

Figures 2.2, 2.3 and 2.4 display the results of applying MMKF to the problem of HRF model selection when the correct model belongs to the set of tested models. The results clearly show that the selection is successful as model 6 is given a very high probability at the end of the data. As expected, higher SNR leads the method to converge faster (almost 100% after 50s for SNR=15), but even for SNR=1, the system converges to a probability around 90% in less than 200s.

Also worth noting is the fact that across all levels of SNR the second most likely model (though much less than the selected one) is consistently model 5. Inspecting figure 2.1 this seems natural as model 5 is the only other model to have a time-to-peak as short as the one featured by the correct model.

2.3.2 Empirical data

Figures 2.5 and 2.6 show the evolution of the probability of each model and the fit of the best model, respectively, on the visual task, while figures 2.7 and 2.8 show the probability of each model and the best fit, respectively, on the motor task.

On the empirical data, it is harder to draw conclusions as the ground-truth is not available. One can, however, discern that not all instances of this problem are created equal. Effectively, on the first problem, model 3 is selected after processing only a fraction of the data, whereas on the motor task several models attain significant levels of probability. In the end, only models 1 and 2 (high time-to-peak) survive, and even though the final probability for model 1 is almost 80%, it is not completely clear that it is the correct model.

One must bear in mind that with experimental data the correct model will never be in the set of models that the algorithm can choose from and this may lead to indefiniteness such as the one observed for the motor task. In this particular case it may be that models 1 and 2 are close in terms of Baram proximity [61] to the true system generating the data and thus the MMKF doesn’t converge to any of them. This result clearly illustrates the need for strong a priori hypotheses regarding the set of plausible models to be tested.
Figure 2.2: Probability of the different haemodynamic models, when the simulated model is number 6 with SNR=15

Figure 2.3: Probability of the different haemodynamic models, when the simulated model is number 6 with SNR=5
Figure 2.4: Probability of the different haemodynamic models, when the simulated model is number 6 with SNR=1

Figure 2.5: Evolution of the posterior probability of each of the 6 models, given the experimental data of the visual task
Figure 2.6: The output of the model with highest posterior probability (red) compared with the experimental data (blue) and the timing of the stimuli (green) for the visual experiment.

Figure 2.7: Evolution of the posterior probability of each of the 6 models, given the experimental data of the motor task.
Figure 2.8: The output of the model with highest posterior probability (red) compared with the experimental data (blue) and the timing of the stimuli (green) for the motor task
Chapter 3

Dynamic Causal Model Estimation

Contents

3.1 Introduction ................................................. 26
3.2 Methods ....................................................... 26
   3.2.1 Multiple-Model Kalman Filter ........................... 27
   3.2.2 Simulations ............................................ 27
3.3 Results and Discussion ..................................... 33
   3.3.1 Simulations with known parameters .................... 33
   3.3.2 Simulations with unknown parameters .................. 33
   3.3.3 Simulations with unknown parameters - covering the parameter space . 40
   3.3.4 Summary of the results for unknown parameters ........ 46
3.1 Introduction

The DCM abstracts the functional connectivity between brain areas that explains how each area influences the neuronal activity in other areas:

\[
\dot{z}(t) = (A + \sum_j u_j(t)B^{(j)})z(t) + Cu(t)
\] (3.1)

In this model, each region has a state - the neural activity that changes with time, through a first-order differential equation. The matrix that multiplies these states in the differential equation is called intrinsic connectivity matrix. Its diagonal terms are negative and correspond to the self-regulation of each brain area - the magnitude of these indicating the slower or faster dynamics of the system. The off-diagonal terms are the intrinsic connections between the brain areas. Each input has an associated modulation matrix, where each entry defines how (and if) this input modulates a specific connection. Finally, there is a matrix defining which brain area(s) the inputs directly influence.

Here, \( z(t) \) is the vector of neuronal activity at different brain areas, and \( u(t) \) is the vector of all inputs at time \( t \) (the inputs are any known external stimuli that can influence brain activity, within a controlled experiment). \( A \) has negative terms on the diagonal (natural rate of decay of neuronal activity in each area) and the non-zero terms off-diagonal represent intrinsic communication between brain areas for the given task. Each \( B \) controls the influence the \( j \)-th input has on the connections between brain areas (for instance, when a subject is told to pay attention to stimuli, certain connections may be strengthened and others weakened). Matrix \( C \) represents the direct influence of stimuli on each brain area.

The \( z(t) \) at each area is fed as the input to a system of equations such as (2.1). This comprises the model of how an experiment with a certain design \( u(t) \) influences the neuronal activity \( z(t) \) which itself causes the MRI signal \( y(t) \).

The main goal of this chapter is to determine which brain areas have a functional connection during a certain task or which connections a certain stimulus modulates. This amounts to determining the structure of matrices \( A \) and \( B \), that is, which off-diagonal terms are zero.

3.2 Methods

The MMKF will again be used to select between a number of models. This time taking into account the coupled system defined by (3.1) and (2.1). Starting with 3.2.1, the discretization of DCM and coupling with HRF such that the full system can be represented in a MMKF are presented. In section 3.2.2, the models and parameters are first described and then the several algorithms used to select between the models are outlined.
3.2.1 Multiple-Model Kalman Filter

As in the previous chapter, (3.1) must be adapted so as to fit into the MMKF. We start by obtaining a discrete-time model that describes the dynamics in (3.1). It is continuous in time, and so must be discretized.

Defining \( F(t) = A + \sum_j u_j(t)B^{(j)} \), and assuming the inputs are continuous in the interval \( t \in [k\Delta T, k\Delta T + \Delta T[ \forall k \), it is possible to integrate (3.1) at the sampling times on each of the intervals:

\[
z(k\Delta T + \Delta T) = e^{F(k\Delta T)\Delta T} \cdot z(k\Delta T) + (e^{F(k\Delta T)\Delta T} - I)C \cdot F(k\Delta T)^{-1}u(k\Delta T) \quad (3.2)
\]

\[
z[k + 1] = e^{F[k]T}z[k] \cdot z[k] + (e^{F[k]T} - I)C \cdot F[k]^{-1}u[k] \quad (3.3)
\]

In order to apply the MMKF to the full system in one pass, the DCM and HRF are integrated into one augmented state vector, where (for a DCM with \( n \) brain areas) the first \( n \) entries correspond to the vector \( z \) and the state vectors for the HRF (each with 5 elements) are stacked below. The matrices of the dynamical system are assembled in a compatible way, thus completing the definition of the LDTV that a KF requires.

3.2.2 Simulations

3.2.2.1 Models and Parameters

In the following simulations that evaluate different methods to estimate connectivity, the MMKF is faced with the problem of selecting between two slightly different connectivity structures. In the following there are two scenarios, both from [26].

The first one can be seen in figure 3.1. The only difference between the models is that the second one exhibits backward connectivity. Since this scenario tests a difference between intrinsic connectivity structures, it is henceforth referred to as the intrinsic scenario. The inputs to this model in all experiments can be seen in figure 3.2.

The second scenario is depicted in figure 3.3. Both models have the same intrinsic structure but the first one has an input modulating a connection on the left side of the brain whereas the other has the same input modulating a connection on the right side of the brain. This scenario will be, from now on, described as the modulating scenario. The inputs to this model in all experiments can be seen in figure 3.4.

Unless otherwise noted, a simulation from any model in these scenarios consists of 200s of data generated with an integration step of 0.1s and posterior sampling to simulate TR=3s. In the context of this thesis for the estimation of the DCM structure, the HRF is assumed known and it corresponds to the fourth set of parameters in table 2.1.
Figure 3.1: Two models that only differ at intrinsic connectivity: Left - Model that only has forward connections, Right - Model that has forward and backward connections. Taken from [26]

Figure 3.2: The three input signals used for the intrinsic connectivity scenario
Figure 3.3: Two models that are the same in intrinsic connectivity and model interconnected areas from the two hemispheres of the brain. Left - Model where the connection from the lower to the higher area is modulated on the left hemisphere, Right - Model where the connection from the lower to the higher area is modulated on the right hemisphere. Taken from [26]

Figure 3.4: The two input signals used for the *modulating scenario*
Gaussian white noise is added to the outputs to generate data with different SNR. Unlike the HRF case, the SNR is not estimated but assumed known (i.e. possibly estimated from the signal in regions where there is no activity).

### 3.2.2.2 Known parameters

The first approach to gauge the performance of MMKF in estimating the underlying structure of a DCM is to generate data from each of the models, for a given scenario. For each model in the scenario, repeat 100 times:

1. Simulate data from this model and add noise for the intended SNR
2. Set up a MMKF with all the models in this scenario and run it

Compute the accuracy of the algorithm for the given model.

This is the simplest case possible, since the only difficulty is the added noise. The nonlinearity (the HRF) is known and one of the models that the algorithm can pick is equal to the model that generated the data. For the known parameters, the results are presented as box plots for an SNR of 1 in 3.2.

### 3.2.2.3 Unknown parameters

The second approach consists of generating data, not from the exact model depicted in the figures above, but from a model that has the same structure but random parameters in a given range. For each model in the scenario, repeat 500 times:

1. Generate a model with this structure and each parameter sampled from $U([0.1, 0.9])$ guaranteeing stability
2. Simulate data from the above model and add noise for the intended SNR
3. Set up a MMKF with all the models in this scenario having all parameters as close to 0.5 as possible, ensuring stability, and run it

Compute the accuracy of the algorithm for the given model.

This case is a lot closer to reality since one cannot know a priori the exact parameterization of a given model. For a given connectivity structure, the parameters of the model may span a relatively wide range of values. When ensuring stability, the worst case scenario is considered where all inputs are set to 1, and the eigenvalues of the matrix that multiplies $z(t)$, (previously denoted as $F(t)$), must respect:

$$\max(Re(\lambda)) < 0$$  \hspace{1cm} (3.4)

\[500 \text{ instead of } 100 \text{ because each one is actually a different dynamical system}\]
For the second and third approaches where the parameters of the model are assumed unknown, results are presented for three different levels of noise, where the SNR ranges from 0.25 to 10 and the algorithm’s performance is evaluated after a fraction of the data has been processed, as well as after the whole data. Each of the 500 Monte-Carlo runs falls into one of 8 bins that characterize the strength of the evidence for model 1 vs model 2, according to the final probability attributed to each by the algorithm as expressed in Table 3.1.

<table>
<thead>
<tr>
<th>Evidence</th>
<th>Probability of model 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very strong 1</td>
<td>[0.99, 1]</td>
</tr>
<tr>
<td>Strong 1</td>
<td>[0.95, 0.99]</td>
</tr>
<tr>
<td>Positive 1</td>
<td>[0.75, 0.95]</td>
</tr>
<tr>
<td>Weak 1</td>
<td>[0.5, 0.75]</td>
</tr>
<tr>
<td>Weak 2</td>
<td>[0.25, 0.5]</td>
</tr>
<tr>
<td>Positive 2</td>
<td>[0.05, 0.25]</td>
</tr>
<tr>
<td>Strong 2</td>
<td>[0.01, 0.05]</td>
</tr>
<tr>
<td>Very strong 2</td>
<td>[0, 0.01]</td>
</tr>
</tbody>
</table>

Table 3.1: Evidence Labels

**Dependence on input**  The input signals in all of the previous simulations are fixed and periodic. In particular, for the intrinsic case, the input signal that is fed to the first neuronal area has a period of 10 seconds and a duty cycle of 20% (figure 3.2). Performing the same Monte-Carlo simulations described in the previous section, using \(\text{SNR} = 1\), but varying the two parameters (period and duty cycle) of the input signal to the first neuronal area, one can try to determine how they influence the accuracy of the algorithm. This is performed for the *intrinsic scenario* and the classification of the algorithm for each set of input parameters is scored according to the following formula:

\[
S = \#(P(X) > 0.75) - \#(P(X) < 0.25)
\] (3.5)

That is, for each experiment where there is at least positive evidence for the correct model, one point gets added to the score, while each experiment with at least positive evidence for the incorrect model deducts one point from the score. All experiments that have weak evidence for one of the models do not influence the score.

Hence, for 100 Monte-Carlo runs, the highest score is 100 (the algorithm attributes at least positive evidence to the correct model every time), the lowest is -100 (the algorithm attributes at least positive evidence to the wrong model every time), and a score of 0 is expected from a random algorithm (equal number of experiments with positive evidence for the correct and wrong model).

### 3.2.2.4 Unknown parameters - covering the parameter space

The last approach generates the data exactly in the same way as the second one, but differs in the algorithm used to select the models. It does not have a one-to-one correspondence between models
in the scenario and KFs in the MMKF where the models in the MMKF have all parameters equal to 0.5, thus partitioning the parameter space in half.

Instead, for a given model, it generates all the possible models where each parameter can take two values (partitioning the space in three equal parts) hence a model with \( n \) connections\(^2\) generates \( 2^n \) different representations in the MMKF (minus the representations which are unstable).

This allows for a much better covering of the parameter space, hence increasing the likelihood of the MMKF converging to one of the KFs that correspond to the correct connectivity structure.

For each model in the scenario, repeat 500 times:

1. Generate a model with this structure and each parameter sampled from \( U([0.1, 0.9]) \) guaranteeing stability

2. Simulate data from the above model and add noise for the intended SNR

3. Set up a MMKF with \( 2^n \) models for each model in the scenario, ensuring stability, and run it

4. Compute the assigned probability to each model in the scenario, by summing the probabilities the MMKF attached to all the models with the same structure

Compute the accuracy of the algorithm for the given model.

Obviously, with the number of models in the MMKF being exponential in the number of connections, the time required to execute this algorithm is far larger than the other ones. For this reason, only 200 simulations are done for the intrinsic scenario. However, since the algorithm is run offline, this is not dramatic.

### 3.2.2.5 Improving computational efficiency

Since the number of KFs increases exponentially in this approach, a way to improve the efficiency is by ignoring (i.e. not running) KFs that have gone below a certain level of probability in the MMKF. In this way, a good number of models can be eliminated in the initial integration steps of the MMKF and allow the rest of the process to run much faster.

**Modelling errors as added variance in the measurements** Since none of the KFs in the MMKF accurately models the real model that generated the data, the covariance of the Gaussian noise that was added to the simulated data, may not be the best matrix to use as the covariance of the noise for the measurements in each KF. In fact, knowledge about these modelling errors may be introduced in the covariance matrix of each KF by increasing its diagonal values, or equivalently initiating the MMKF algorithm as if the SNR were smaller than the actual level.

This may be specially important for high level SNR; therefore, the third approach was ran for the intrinsic scenario and data with SNR=10, but running the MMKF algorithm different values of

\(^2\)Remember that in this context each connection in a model corresponds to a parameter.
3.3 Results and Discussion

3.3.1 Simulations with known parameters

It is clear from Figure 3.5 that in almost all cases, model 2 ended up with probability 1, which leads to an unusual box plot. In the end, only 3 out of the 100 simulations had model 1 with a higher probability, and in 91 of the simulations, model 2 had a probability higher than 99%.

The results are not as clear as for the modulating scenario in figure 3.6. Nonetheless, the first model is preferred just 8 times out of 100 and the second model has a final posterior probability over 99% in 56 out of 100 simulation tests.

In the first approach where the parameters are known, the algorithm performs very well. This can be seen in figures 3.5 and 3.6. As expected, the correct model is given a very high probability in most cases. There is an output nonlinearity, but since the correct model with the exact same parameters can be chosen by the MMKF, this cannot be considered a hard classification task.

3.3.2 Simulations with unknown parameters

The second approach, closer to reality, exposes some of the weaknesses of the algorithm. In the intrinsic scenario, and when data are generated from model 1, the results in figures 3.7 and 3.8 seem close to perfect regardless of the SNR and even the amount of data processed. However, when data are generated from model 2, figures 3.9 and 3.10 still report that model 1 is widely preferred, which is incorrect.

This may mean that, since the parameters used to represent the models do not cover the parameter space adequately, not only does data from the structure of model 1 fit those parameters, but also data generated by model 2 will fit them in most cases.

In the end, at the highest SNR and using the whole data from the intrinsic scenario, the second approach returns at least strong evidence for the correct model 64% of the time, and strong evidence for the wrong model 32.5% of the time. This, while not great, may seem reasonable, but if one remembers that it fails clearly when data are generated from model 2, it turns out that the algorithm is unusable in this setup.

Still using the second approach, now for the modulating scenario, the results are more even, almost symmetrical (as expected) when data are generated by model 1 or 2 (figures 3.11, 3.12, 3.13 and 3.14). Low SNR typically result in a higher portion of the simulations being classified as weak evidence for one of the models, while higher SNR tends to produce results with stronger evidence. Using the whole data also improves the results of the classification.
Figure 3.5: Box plots of the posterior probabilities for each of the models in the intrinsic connectivity selection where the correct model is model 2. As usual the median, 25th and 75th percentiles as well as the whiskers (encompassing 1.5 times the distance between 25th and 75th percentiles) and outliers are shown. In this case the values are so compressed that one can only discern the median line and the outliers.

Figure 3.6: Box plots of the posterior probabilities for each of the models in the modulation selection where the correct model is model 2. As usual the median, 25th and 75th percentiles as well as the whiskers (encompassing 1.5 times the distance between 25th and 75th percentiles) and outliers are shown.
Figure 3.7: Bar plots of the evidence for each *intrinsic scenario* model when the correct is 1, after 50s of data.

Figure 3.8: Bar plots of the evidence for each *intrinsic scenario* model when the correct is 1, after 200s of data.
Figure 3.9: Bar plots of the evidence for each intrinsic scenario model when the correct is 2, after 50s of data.

Figure 3.10: Bar plots of the evidence for each intrinsic scenario model when the correct is 2, after 200s of data.
Figure 3.11: Bar plots of the evidence for each modulating scenario model when the correct is 1, after 50 s of data.

Figure 3.12: Bar plots of the evidence for each modulating scenario model when the correct is 1, after 200 s of data.
Figure 3.13: Bar plots of the evidence for each modulating scenario model when the correct is 2, after 50 s of data.

Figure 3.14: Bar plots of the evidence for each modulating scenario model when the correct is 2, after 200 s of data.
Figure 3.15: Surf plot of the classification score for different inputs in the *intrinsic scenario* when the correct model is the one without backward connectivity (1).

With $\text{SNR}=10$ and using the whole data, the second approach returns at least strong evidence for the correct model 66% of the time whereas the wrong model only has strong evidence 20% of the time.

This approach seems very unreliable for the *intrinsic scenario*, yet the results are quite acceptable (though not great) for the *modulating scenario*.

### 3.3.2.1 Dependence on input

Figure 3.15 depicts the performance of this approach for the *intrinsic scenario* when model 1 generates the data, depending on the input parameters. It is clear that the parameters chosen initially are close to the peak performance of the algorithm for this data, which shows a tendency to decrease as the period of the input signal is diminished.

In general it should be possible to perform similar *a priori* studies to determine optimal input characteristics that maximize distinguishability of different connectivity models. In practice, there has been some work on how to determine the effect of experimental designs on activity detection or HRF estimation using [MRI] [64] and how to optimize these parameters [65]. Recently, there has been some work on the distinguishability of different HRFs in the framework of multiple-model SVOs [56].
Evidence for Model 1 vs Model 2

Figure 3.16: Bar plots of the evidence for each intrinsic connectivity model with parameter space covering, when the correct is 1, after 50 s of data.

The more interesting problem of optimizing of the experimental design for the estimation of effective connectivity (that is, distinguishing between different models or classes of models) has only very recently been tackled [66].

3.3.3 Simulations with unknown parameters - covering the parameter space

The third approach consists of a classification task as hard as the one in the second approach (thus more realistic than the first one) but the algorithm is slightly modified in the expectation that it can overcome the weaknesses of the latter approach.

By partitioning the parameter space, it is more likely that one of the KFs in the MMKF will be close to the model generating the data, hence theoretically making the algorithm more robust and more capable of choosing correctly between different connectivity structures.

In the intrinsic scenario, the results in figures 3.16 and 3.17 relative to data generated by model 1 are worse than the ones obtained by the second approach, but still particularly good especially with the whole data and $SNR=10$.

On the other hand, for data generated by model 2 in figures 3.18 and 3.19 the algorithm now behaves correctly, as opposed to what happened in the second approach. Again, the classification accuracy increases with the $SNR$. Notably, with $SNR=10$ and using the whole data, this approach returns at least strong evidence for the correct model 91.5% of the time, which is strikingly better than the results of the previous approach.
Figure 3.17: Bar plots of the evidence for each intrinsic connectivity model with parameter space covering, when the correct is 1, after 200 s of data.

Figure 3.18: Bar plots of the evidence for each intrinsic connectivity model with parameter space covering, when the correct is 2, after 50 s of data.
3.3.3.1 Improving computational efficiency

First, it was confirmed that the time to run MMKF on one instance of data are greatly reduced by using the computationally efficient approach (anything from 50 to 90% of reduction). The degree of optimization depends on the data, but is higher and more reliable with higher SNR. This was expected as higher SNR makes the MMKF trust the data more and hence reduce the probability of the inadequate models, faster, eventually eliminating them.

As for the actual results, they are depicted in 3.20 and 3.21 when data are generated by model 1 and 3.22 and 3.23 when data are generated by model 2.

The results are slightly better when data are generated by model 1, but this was also the case in the unoptimized third approach. For SNR=10 this approach returns at least strong evidence for the correct model in 74% of the cases. This was expected as it is possible that eliminating some of the models eliminated early on, may result in a slight decrease of accuracy. Less expected was the fact that the results were better for SNR=1. In fact, with the lower SNR there was strong evidence in favor of the correct model in 84.5% of the cases.

3.3.3.2 Modelling errors as increased variance in the measurements

Changing the covariance of the measurement error in the MMKF was partly motivated by the results just discussed. In a framework where eliminating KFs is the source of optimization, and this elimination is faster for greater SNR this can become a source of error if (as frequently happens in the simulations) none of the KFs corresponds closely to the model generating the data.

As can be seen in figures 3.24 and 3.25 for data generated by models 1 and 2, respectively, increas-
Figure 3.20: Bar plots of the evidence for each intrinsic connectivity model with efficient parameter space covering, when the correct is 1, after 50 s of data.

Figure 3.21: Bar plots of the evidence for each intrinsic connectivity model with efficient parameter space covering, when the correct is 1, after 200 s of data.
Figure 3.22: Bar plots of the evidence for each intrinsic connectivity model with efficient parameter space covering, when the correct is 2, after 50 s of data.

Figure 3.23: Bar plots of the evidence for each intrinsic connectivity model with efficient parameter space covering, when the correct is 2, after 200 s of data.
Figure 3.24: Bar plots of the evidence for each intrinsic connectivity model with efficient parameter space covering when the correct model is 1, after 200 s of data. SNR in the simulation is 10 and the SNR provided to the MMKF is in the range 1 to 10.

Figure 3.25: Bar plots of the evidence for each intrinsic connectivity model with efficient parameter space covering, SNR=10 and SNR used in the MMKF when the correct model is 2, after 200 s of data.
ing the covariance of the measurement error results in a direct improvement of the classification quality.

When the data has an SNR of 10 and the MMKF is given a covariance matrix equivalent to an SNR of 1, and thus attributes more uncertainty to the data, there is at least strong evidence for the correct model in 85.75% of the cases (higher, though not significantly, than the results obtained with SNR=1).

This seems to indicate that when optimization is desired, too high SNRs do not improve the results significantly but must be accounted for, by modifying the covariance matrix of the measurement errors to the MMKF.

### 3.3.4 Summary of the results for unknown parameters

In the following tables, the results for model selection with unknown parameters are shown. The score metric is the same as the one used to study the dependence on input (section 3.2.2.3) normalized to guarantee all results fall in the range \([-100, 100]\). On table 3.2 the results for the modulating scenario can be seen and the advantages of having higher SNR as well as more data for model identification are very evident.

<table>
<thead>
<tr>
<th>SNR</th>
<th>Seconds of data</th>
<th>Single parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25</td>
<td>50</td>
<td>13</td>
</tr>
<tr>
<td>1</td>
<td>200</td>
<td>22</td>
</tr>
<tr>
<td>10</td>
<td>50</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>41</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>47</td>
</tr>
</tbody>
</table>

Table 3.2: Performance score (normalized to 100) on the modulating scenario

In table 3.3 one can observe the results for the intrinsic scenario, with both algorithms (plus the more efficient version of parameter coverage). The lower scores for the single parameter algorithm are due to its consistent selection of model 1 when model 2 is the correct one, and the amount of data is not particularly relevant. For the algorithm that uses more than one filter per model thus covering the parameter space, the results are much better and both the SNR and amount of data impact them directly. Finally, for the optimized version, the results are quite similar except when the full dataset is available and the SNR is high. This, as discussed before, can be solved by running MMKF with higher noise variance (or equivalently lower SNR): in this case the score is 72.5 for the full dataset and SNR of 10.
<table>
<thead>
<tr>
<th>SNR</th>
<th>0.25</th>
<th>1</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seconds of data</td>
<td>50</td>
<td>200</td>
<td>50</td>
</tr>
<tr>
<td>Single parameter</td>
<td>24</td>
<td>31</td>
<td>29</td>
</tr>
<tr>
<td>Parameter Covering</td>
<td>35</td>
<td>56</td>
<td>67</td>
</tr>
<tr>
<td>Efficient Covering</td>
<td>–</td>
<td>–</td>
<td>68</td>
</tr>
</tbody>
</table>

Table 3.3: Performance score (normalized to 100) on the *intrinsic scenario*
Chapter 4

Conclusion

Contents

4.1 Discussion .............................................. 50

4.2 Future Work ............................................ 50
4.1 Discussion

In this thesis, the MMKF, a multiple model approach to estimation was investigated as a tool to select between models of underlying brain activity from BOLD signal.

First, the problem of estimating an HRF was addressed. This model with strong biophysical underpinning is a nonlinear dynamical system that is bilinearized and discretized in order to fit the MMKF. The method performs well on this task, effectively selecting the correct HRF from a set of plausible models in simulation. The method is also shown to be applicable to experimental data through pre-processing. The results emphasize the critical importance of an appropriate choice of the set of models used for the selection.

Then, the main problem is tackled, i.e. that of determining the structures of functional connectivity between brain areas from BOLD. The DCM dynamical system is integrated and then coupled with the discretized HRF into a full system that corresponds to a single KF. The MMKF works particularly well for the problem similar to the one in the HRF where one of the KFs was designed for a system that corresponds exactly to the model generating the simulated data.

The second case is the one where the data are generated from a model with a given connectivity structure but random parameters. For this more realistic case, simply assigning a KF with parameters in the middle of the parameter range for each connectivity structure, yields between very poor to acceptable results, depending on the models. For the several Monte-Carlo simulations performed, it does not seem to be reliable enough for model selection.

This led to assigning $2^n$ KFs for each connectivity structure, where each parameter may take one of two values, partitioning its range in 3 intervals. This approach is much more successful, yielding accuracy rates around 90% for the scenario that had very poor results when using a single KF for each connectivity structure. This significantly improved performance is attained at the expense of a very computationally expensive method that may be inappropriate for larger networks.

Running the MMKF with an uncertainty in the measurements larger than the one implied by the SNR seems to be beneficial to the accuracy, as it takes into account the modelling errors and does not eliminate plausible models too early.

In summary, a new approach for the estimation of neural connectivity from fMRI based on models DCM is proposed using multipled-model estimation techniques (MMKF), which may be more robust to the multi-modality of the posterior probabilities of the parameters than the classical technique.

4.2 Future Work

An important topic that was not addressed in this thesis is the actual comparison of performance in simulated datasets of the MMKF algorithm and the current state-of-the-art algorithms that
actually try to estimate the parameters of the DCM in order to perform model identification. This would elucidate more clearly the possible weaknesses/strengths of the proposed methodology.

Regarding the development of the method, there are several points that need to be addressed in the MMKF algorithm. Preprocessing (for determining the appropriate SNR) and making the algorithm insensitive to temporal scaling in the DCM matrices, as well as to the amplitude of the signals, are essential steps, if the algorithm is to be used on experimental data. The current limitation of relying on a known HRF for each area should also be tackled.

The only way of performing reliable model selection described in this thesis is exponential in the number of connections of the network. Even with the technique that improves the efficiency (up to ten-fold), the asymptotic behaviour remains unchanged which precludes from using such algorithm in networks with more than a few connections. This is a critical shortcoming and only improving the asymptotic behaviour will make the algorithm useful for larger networks.

Following the simulations that allowed a preliminary study of the performance of the algorithm for different input characteristics, it would be interesting to automate the inference/search for the optimal experimental design in order to unveil a specific underlying connectivity structure.

Furthermore, there are many applications of connectivity estimation where the input is unknown or one has very little information about it. The development of algorithms for estimation of connectivity that work under these circumstances is very important.
Bibliography


Appendix A

Time-varying DCM stability

In this appendix, the claim made in chapter 3 that the stability of an invariant linear system corresponding to the activation of all modulating inputs during an experiment (which can be verified easily through the computation of the linear system’s matrix eigenvalues), is sufficient to guarantee the stability of the time-variant linear system corresponding to DCM in any experiment.

A.1 Positive Linear Systems and Positively Perturbed Linear Systems

Definition 1. An autonomous positive linear system is an autonomous linear system which has a matrix $A$ with nonnegative off-diagonal entries.

$$\dot{x} = Ax, \forall i \neq j, A_{ij} \geq 0$$

Lemma 1. Any trajectory of a positive linear system that goes into the positive orthant of $\mathbb{R}^n$, never leaves it:

$$\exists t_0 \forall i x_i(t_0) \geq 0 \Rightarrow \forall t > t_0 \forall i x_i(t) \geq 0$$ (A.1)

Proof. From [67]

It is sufficient to show that if the state goes to the boundary of the positive orthant it doesn’t leave the orthant.

In order to have one element $x_i$ of the state become negative, there must be a time $t' \geq t_0$ such
that:

\[ \forall j \quad x_j(t') \geq 0 \quad (A.2) \]
\[ x_i(t') = 0 \quad (A.3) \]
\[ \dot{x}_i(t') < 0 \quad (A.4) \]

Let us show that, for a positive linear system, (A.2) and (A.3) imply the negation of (A.4):

\[ \dot{x}_i(t') = \sum_j A_{ij} x_j(t') = \sum_{j \neq i} A_{ij} x_j(t') + A_{ii} x_i(t') \geq 0 \quad (A.5) \]

Definition 2. A positively perturbed system is a dynamical system where all the entries of the matrix \( A \) are greater or equal to the entries of the original system’s matrix \( A \).

\[ \dot{x} = Ax \rightarrow \text{original system} \quad (A.6) \]
\[ \dot{x}' = A'x' \rightarrow \text{perturbed system} \quad (A.7) \]
\[ \forall i,j A'_{i,j} \geq A_{i,j} \quad (A.8) \]

A positively perturbed positive linear (PPPL) system is a positive linear system which is positively perturbed.

Lemma 2. If all state elements of a PPPL system are greater than those of the original system at time \( t_0 \), and in the positive orthant, they remain greater:

\[ \forall i, x_i'(t_0) \geq x_i(t_0) \Rightarrow \forall i, x_i'(t) \geq x_i(t) \quad (A.9) \]

Proof. Let us start by proving that if the state elements of the PPPL system are all greater than those of the original system at a certain time \( t \geq t_0 \), the same if true for the derivatives of the state elements:

\[ \forall i, x_i'(t) \geq x_i(t) \Rightarrow \forall i, \dot{x}_i'(t) \geq \dot{x}_i(t) \quad (A.10) \]

Proof.

\[ \dot{x}_i'(t) = \sum_j A'_{ij} x_j'(t) \geq \sum_j A_{ij} x_j(t) = \dot{x}_i(t) \quad (A.11) \]

Where \( A'_{ij} \geq A_{ij} \) because the system is positively perturbed, \( x_j'(t) \geq x_j(t) \) because of our assumption, and \( x_j(t) \geq 0 \) because the system is linear positive and the state was in the positive orthant at \( t_0 \leq t \). \qed
Let us define:

\[ d(t) = x'(t) - x(t) \]  
\[ \dot{d}(t) = \dot{x}'(t) - \dot{x}(t) \] 

(A.12)  

(A.13)

Proving Lemma 2 is equivalent to proving:

\[ \forall_i d_i(t_0) \geq 0 \Rightarrow \forall t \geq t_0 \forall_i d_i(t) \geq 0 \]  

(A.14)

Let us prove it by contradiction. Assume that there is a time \( t_e > t_0 \), the earliest where the PPPL system has a smaller state element than the original system:

\[ \exists t_e (\forall t_0 \leq t < t_e \forall_i d_i(t) > 0) \land (\exists_i d_i(t_e) < 0) \]  

(A.15)

Since solutions of linear systems are differentiable in their domains so is each of the elements of \( d(t) \), and thus by the mean value theorem:

\[ \exists t_c \in [t_0, t_e] \frac{d_i(t_e) - d_i(t_0)}{t_e - t_0} = \dot{d}_i(t_e) < 0 \]  

(A.16)

(A.16) is the negation of the consequent of (A.10), which implies the negation of the antecedent:

\[ \exists_i d_i(t_e) < 0 \]  

(A.17)

And notice that this produces a contradiction with our assumption (A.15) that \( t_e \) was the earliest time at which the PPPL system had a smaller state.

\[ \square \]

A.2 DCM stability

In DCM, taking into account only the elements that multiply the state in the differential equation:

\[ \dot{x}(t) = (A + \sum_k B^{(k)}u_k(t))x(t) \]  

(A.18)

Where \( B^{(k)} \) represents the matrix associated with the \( k^{th} \) input, not the \( k^{th} \) power of a matrix \( B \). \( A \) is a matrix with nonnegative off-diagonal entries, and all the elements of matrices \( B^{(k)} \) are nonnegative. The inputs \( u_k(t) \) are either 0 or 1 at each \( t \), and they are piecewise constant (Fig.A.1). Defining \( t_0 = 0 \) and \( \{t_m\}, m = 1...M \) the M time instants where one of the inputs changes its value, and \( t_{M+1} \) the end of the experiment, we have M+1 time intervals \( [t_m, t_{m+1}] \) where (A.18) represents a time-invariant linear system:
∀t_m \leq t < t_{m+1} \quad \dot{x}(t) = \left( A + \sum_k B^{(k)} u_k(t_m) \right) x(t)
\tag{A.19}

For all $m$, the time-invariant system defined is positive linear and it’s easy to define a PPPL system for all time-intervals:

\forall m \forall i,j \quad A_{ij}^{(m)} \leq A_{ij} + \sum_k B_{ij}^{(k)} = A'_{ij} \tag{A.20}

**Definition 3.** An autonomous linear system is uniformly exponentially stable iif:

\exists \gamma, \lambda > 0 \forall t_0, x_0 \forall t \geq t_0 \quad ||x(t)|| \leq \gamma e^{-\lambda(t-t_0)} ||x(t_0)|| \tag{A.21}

**Theorem 1.** A time-invariant autonomous linear system:

\dot{x} = Ax \tag{A.22}

is (uniformly) exponentially stable iif all eigenvalues of $A$ have negative real parts.

Proof. See [47]

**Theorem 2.** If all the eigenvalues of matrix $A'$ have negative real part, the original DCM system is uniformly exponentially stable.

Proof. Each of the elements of $x$ represents the activity of the brain in a certain area and 0 means no activity, therefore negative values don’t make sense in this model. For the initial state:

\forall i \quad x_i'(t_0) = x_i(t_0) \geq 0 \tag{A.23}

In the first time-interval $[t_0, t_1[$, the original system is represented by matrix $A^{(0)}$. According to Lemmas [1] and [2]
∀_{t_0 \leq t \leq t_1} \forall_i x'_i(t) \geq x_i(t) \geq 0 \quad (A.24)

In particular:

∀_i x'_i(t_1) \geq x_i(t_1) \geq 0 \quad (A.25)

Lemmas 1 and 2 can be applied to the interval \([t_1, t_2]\), and inductively to all intervals \([t_m, t_{m+1}]\), \(m=1...M+1\), resulting in:

∀_{t_0 \leq t \leq t_{M+1}} \forall_i x'_i(t) \geq x_i(t) \geq 0 \quad (A.26)

The above expression, implies:

∀_{t_0 \leq t \leq t_{M+1}} \|x'(t)\| \geq ||x(t)|| \quad (A.27)

Application of Theorem 1 for our assumptions (all eigenvalues of \(A'\) have negative real part) means that the PPPL system associated with \(A'\) is uniformly exponentially stable.

On the other hand, from (A.27), it is clear that if the PPPL system with state \(x'(t)\) respects Definition 3 then so does the original system with state \(x(t)\).