UNDERSTANDING WHOLE BRAIN ACTIVITY THROUGH BRAIN NETWORK MODELS

A Dissertation
Presented to
The Academic Faculty

by

AMRIT KASHYAP

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
Biomedical Engineering

Georgia Institute of Technology and Emory University
December 2020

COPYRIGHT © 2020 BY AMRIT KASHYAP

UNDERSTANDING WHOLE BRAIN ACTIVITY THROUGH BRAIN NETWORK MODELS

Approved by:

Dr. Shella Keilholz, Advisor
Department of Biomedical Engineering
Georgia Institute of Technology and Emory University

Dr. Lena Ting
Department of Biomedical Engineering
Georgia Institute of Technology and Emory University

Dr. Christopher Rozell
Department of Electrical Engineering
Georgia Institute of Technology

Dr. Sergey Plis
School of Computer Science
Georgia State University

Dr. Bruce Crosson
Department of Neurology
Emory University

Dr. Madeleine Hackney
Department of Medicine
Emory University

Date Approved: December 2, 2020

ACKNOWLEDGEMENTS

I would like to first and foremost thank my advisor Professor Shella Keilholz in guiding me through all the technical and academic hurdles. I have learnt an immeasurable amount of skill and experience under her tutelage. I would also like to thank Professor Madeleine Hackney who has pushed me further into solving problems in the clinical setting than I was ever planning on attempting. I learned a lot from her and Professor Bruce Crosson, on how to translate my research in order to serve a broader clinical audience. Professor Lena Ting has also been an excellent mentor, and I have gained insight of being able to bridge the scientific, clinical and engineering challenges from her. I would also like to thank Professor Constantine Dovorolis and Professor Chethan Pandarinath in insightful comments on the Autoencoder and network propagation of brain signals, as well as Professor Chris Rozell in his insight in creating sparsity models based on biological resource constraints. Professor Sergey Plis has helped me the foremost in terms of understanding modern Machine Learning tools and without his help and guidance I would also not been able to develop the algorithms listed in this thesis. Lastly, I would like to thank the institutions of Georgia Tech and Emory University in providing me all the tools to attempt this endeavor and the city of Atlanta for being my wonderful home the last six years.

Lastly, I would like to thank my parents, Dr. Sai Ramakrishna and Padmini Jyotishmati and all my friends especially those in Atlanta. They have supported me emotionally and spiritually through this journey and without them I would not have been able to complete this dissertation.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii

LIST OF TABLES vii

LIST OF FIGURES viii

SUMMARY ix

CHAPTER 1. Introduction: Representing whole brain activity using a network-based model 1
1.1 Background 1
1.2 Brain Network Model 2
1.2.1 Parcellation 3
1.2.2 Structural Network 4
1.2.3 Dynamical System 5
1.2.4 Firing Rate Model 5
1.2.5 Non-linear Model 6
1.2.6 Comparison to resting state fMRI 7
1.3 Applications 8
1.4 Gap 9
1.5 Organization of the Thesis 11

CHAPTER 2. Dynamic properties of simulated brain network models and empirical resting-state data 14
2.1 Introduction 18
2.2 Methods 23
2.2.1 Methods General 24
2.2.2 Structural Connectome 25
2.2.3 Brain Network Models 27
2.2.4 Converting to BOLD via Balloon Windkessel Model 29
2.2.5 Pre-processing rs-fMRI 30
2.2.6 Dynamic Analysis Techniques 31
2.3 Results 35
2.3.1 Reproduction of Previous Literature Results 35
2.3.2 Point Process and Coactivation Rates 38
2.3.3 Quasiperiodic Pattern Algorithm Comparison 39
2.3.4 K-means on Sliding Windowed Matrices 42
2.3.5 Recurrence Quantification Analysis 48
2.4 Discussion 50
2.4.1 Analysis of Average Functional Connectivity and Power Spectra 50
2.4.2 Coactivation Patterns/ Point Process 51
2.4.3 Quasiperiodic Patterns analysis 52
2.4.4 K-means Analysis 53
2.4.5 Recurrence Analysis 54

2.4.6 Overall Discussion 55
2.4.7 Limitations 56
2.5 Conclusion 58

CHAPTER 3. Modeling resting state fmri evolution using Neural Ordinary differential Equations 59
3.1 Introduction 63
3.2 Methods 67
3.2.1 Neural Ordinary Differential Equations 67
3.2.2 Tensorflow Implementation 70
3.2.3 Brain Network Model 72
3.2.4 Simulated Spiral Data 74
3.2.5 Experimental Data 76
3.2.6 Metrics and Evaluation 78
3.2.7 Null Models 78
3.3 Results 79
3.3.1 Validation of the Neural ODE Algorithm on Spiral Data 79
3.3.2 RNN Parameter Estimations on Spiral Data 80
3.3.3 Metric Choices and RNN Parameter Estimations on fMRI data 82
3.3.4 Differentiating Between Dynamical systems on Spiral Data 85
3.3.5 Differentiating Between BNMs due to differences in Parameterization in the Firing Rate Model 88
3.3.6 Differentiating Between BNM due to differences in Structural Connectivity 90
3.3.7 Nonlinear vs Linear BNM 92
3.4 Discussion 93
3.4.1 Validation on the Spiral Dataset and Comparison to other Architectures 94
3.4.2 Parameterization of the Firing Rate Model 95
3.4.3 Parameterization of the Firing Rate Model while changing the Structural Connectivity 96
3.4.4 Comparison between Non-linear and Linear Models 97
3.4.5 Assumptions and Limitations 98
3.5 Conclusion 99

CHAPTER 4. Modeling resting state fmri evolution using RNN Inference and BRain Network Models 100
4.1 Introduction 104
4.2 Methods 107
4.2.1 Mathematical Background of the Whole Architecture 108
4.2.2 Brain Network Model 109
4.2.3 Long Short-Term Memory and Implementation 112
4.2.4 Experimental Data 115
4.2.5 Dynamical analysis techniques 117
4.3 Results 117
4.3.1 One Time Step prediction 117
4.3.2 Multiple Time Step prediction 119
4.3.3 Analysis of Long Simulations 121
4.3.4 Quasi Periodic Pattern Analysis of Simulated Data 123

4.3.5 K-means analysis of Simulated Data 125
4.3.6 Comparison to the Machine Learning Null Model in terms of Long Term Metrics 127
4.4 Discussion 128
4.4.1 Predicting Moment to Moment Variations 129
4.4.2 Evaluations on Long Term Dynamics 130
4.4.3 Errors across different ROIs 132
4.4.4 Comparison to other Machine Learning and Time Forecasting models 133
4.4.5 Limitations 134
4.5 Conclusion 135

CHAPTER 5. Evaluating Our Contributions to Future and CuRRent applications of Brain network Models 136
5.1 Analysis of Brain activity at the Systems Level 136
5.2 Gaps in developing better Brain Network Models 137
5.3 Results and Significance 139
5.4 Relationship to existing Literature 141
5.5 Future Applications in modeling Parkinson's Disease 143
5.6 Neuroethics 145
5.7 Conclusion 147

REFERENCES 149

LIST OF TABLES

Table 2.1	Parameters of the Brain Network Models	29
Table 2.2	Table of Regions of Interest	31
Table 2.3	Dynamic Analysis Techniques	32
Table 2.4	Comparison between models across different analysis Techniques	56
Table 3.1	R-squared Accuracy at the 3rd timestep across different RNN Parameters	83

LIST OF FIGURES

Figure 1.1 Representing rs-fMRI using a Brain Network Model 3
Figure 2.1 Methods General 24
Figure 2.2 Comparison of Structural Connectome Data 26
Figure 2.3 Hemodynamic Response Function 30
Figure 2.4 Comparison of our methodology with the original BNM Kuramoto Model... 35
Figure 2.5 Comparison of Average FC and the Frequency Spectrum 37
Figure 2.6 Average FC and Point Process across Parameters 38
Figure 2.7 Point Process across Brain Network Models 39
Figure 2.8 QPP Algorithm Comparison 41
Figure 2.9 QPP Algorithm Comparison across Parameters 42
Figure 2.10 K-Means Algorithm Comparison 44
Figure 2.11 K-means Analysis Across Parameters 45
Figure 2.12 K-means Analysis with different number of Clusters 46
Figure 2.13 K-means Centroids 47
Figure 2.14 Recurrence Quantification Analysis Comparison 49
Figure 2.15 Recurrence Quantification Analysis Across Parameters 50
Figure 3.1 Schematic for Neural ODE Algorithm 67
Figure 3.2 Neural ODE Architecture 71
Figure 3.3 Generation of Spiral Dummy Data 76
Figure 3.4 Infer Initial condition for each Timestep on Spiral Data 80
Figure 3.5 Effect of Network Size on Initial Condition Predictions 82
Figure 3.6 Individual Variability and Generalization Across Time 85
Figure 3.7 Differentiating between Dynamical Systems on Spiral Data 87
Figure 3.8 Effects of Global Coupling without Noise in the Firing Rate Model 88
Figure 3.9 Effect of varying Global Coupling and Noise on 4th Timestep Accuracy of the Firing Rate Model 90
Figure 3.10 Effects of Global Coupling, Noise, and Structural Connectivity on 4th Timestep Accuracy of the Firing Rate Model 91
Figure 3.11 Performance of Nonlinear sigmoidal Firing Rate Model without Noise 93
Figure 4.1 Schematic of Neural ODE algorithm 108
Figure 4.2 Brain Network Model 112
Figure 4.3 Tuning Parameters 113
Figure 4.4 One Time Step Prediction 119
Figure 4.5 Multiple Time Step Prediction 121
Figure 4.6 Inference FC and Spectrum Comparison 123
Figure 4.7 Quasi Periodic Pattern Comparison 125
Figure 4.8 Comparison of K-means Centroids 127
Figure 4.9 Difference between Null Model and BNA Long Term Predictions 128

SUMMARY

The central nervous system coordinates many neural subpopulations connected via macroscale white matter architecture and surface cortical connections to produce complex behavior depending on environmental cues. The activity occurs over different scales, from information transfer between individual neurons at the synapse level, to macroscale coordination of neural populations used to maximize information transfer between specialized brain regions. The whole brain activity measured through functional Magnetic Resonance Imaging (fMRI), allows us to observe how these large neural populations interact over time. Researchers have developed a set of Brain Network Models (BNMs), that simulate brain activity using the macroscale structure and different mathematical models to represent populational neural activity. These simulations have been able to reproduce properties of fMRI especially those averaged over long periods of time. These models represent a step towards an individualized model of brain activity, which is of clinical interest, as they can be constructed from individual estimates of the structural network. To find a good BNM to fit the individual fMRI data, however, is a difficult problem as BNMs represent a large family of mathematical models. Moreover, a large set of BNMs have reproduced time averaged metrics that have been used so far to compare the models with the fMRI data. In this thesis, we extend previous work on BNM research by establishing new dynamic metrics that would allow us to better differentiate between BNMs simulations on how well they reproduce measured fMRI dynamics (Chapter 2). In Chapter 3, we directly compare transient short-term trajectories by synchronizing the outputs of a BNM in relation to observed fMRI timeseries using a novel Machine Learning

Algorithm, Neural Ordinary Differential Equations (ODE). Finally, we show that the Neural ODE can be used as its own stand-alone generative model and is able to simulate more realistic fMRI signals as they are able to reproduce complex metrics that previous models have not been able to recapitulate (Chapter 4). In short, we demonstrate that we have made progress in developing and quantifying BNMs and advanced the research of more realistic whole brain simulations.

CHAPTER 1.

INTRODUCTION: REPRESENTING WHOLE BRAIN ACTIVITY USING A NETWORK-BASED MODEL

1.1 Background

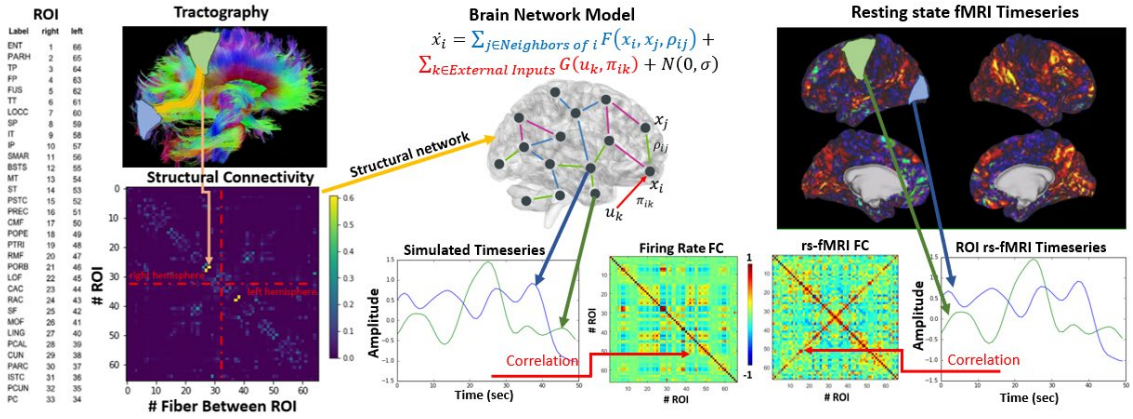
Whole brain activity represents the coordination of many distinct neural populations that allow humans to display a plethora of different behaviors, even though each distinct element of the brain is specialized to perform specific tasks. Although the central nervous system is often described as highly interconnected, the notion of modularization where specific function localizes to a cortical area has been around since nervous systems were first studied by Brodmann in 1909. Brodmann, using staining and microscopy, discovered that the micro architecture varied in different areas of the cortex, and some of the 52 areas that he identified have now become famous for specific functions, such as the primary visual area (Brodmann area 17), primary auditory area (Brodmann area 42), or primary motor cortex (Brodmann area 4). The sensory and motor areas also represent the main input and output connections into the cortex, and adjacent areas specialize in processing more abstract representation of the information. It is theorized that they eventually combine the information from multiple modalities to form our own internal representation of the world. Supporting this theory, recently it has been shown that all the major hubs for network processing are at geodesical equidistant distances on the cortical fold from sensory motor areas, suggestive of a natural gradient from areas interacting directly with information from the outside world to higher order processing (Margulies et al., 2016). However, it would be misrepresentative to describe this system as just a unidirectional flow from sensory

motor areas to higher order areas, as the whole brain activity itself, is really of a network origin. Our sensory systems influence our current internal representation of the world, and the internal representation of the world acts as a prior in decoding our sensory information and encoding our motor commands. Therefore, in order to represent whole brain cortical activity, researchers turned towards network models known as Brain Network Models (BNM), where nodes represent modular components of the nervous system and simulate their activity across an estimated structural network.

1.2 Brain Network Model

Whole Brain activity measured in the low frequency range detected via functional magnetic resonance imaging (fMRI) (~0.01~0.Hz) is thought to be the result of large neural populations interacting via the network as opposed to tracking the activity of real time decoding and encoding of information that are thought to occur in the gamma frequency regime (60-80Hz) (Deco et al., 2008, Friston et al., 2003). The activity measured in this frequency band is considered to reflect averaged properties of a large neural population, such as the mean firing rate. The increase in firing of a neural population triggers a hemodynamic response, where oxygenated blood is transported for metabolic consumption and is subsequently measured using fMRI. Theoretical Network models have been constructed to recapitulate fMRI as shown in Figure 1.1 and are described in the following sections.

Figure 1.1 Representing rs-fMRI using a Brain Network Model



Left panel shows the construction of the network. The cortex is parcellated to 66 regions of interest (ROI) (left most) which act as nodes in the network. The number of white matter tracts in the Diffusion Tensor Weighted image are counted between each pair of regions of interest. The Structural Connectivity Matrix summarizes the relationship. In order to simulate the whole brain activity, the activity in each region of interest represented by x_i changes as a function of the network neighbors and external inputs if it's connected to one of the sensory motor regions. Rs-fMRI is simulated by setting the input term is set to zero for all time. For example, to simulate a simple linear Firing rate model Ax, where A is the graph Laplacian of the structural matrix, we can generate a timeseries by taking a random initial condition and then integrating it according to the equations. To compare it to resting state fMRI, we first average the activity of an entire ROI region from a registered rs-fMRI scan with the same atlas as used in the tractography. Then we measure functional connectivity as the correlation between each pair of ROI timeseries. The resulting functional connectivity (FC) matrices, estimated by correlating the timeseries of all ROI to each other, between the simulated and the empirical timeseries are then compared

1.2.1 Parcellation

The network is defined using a cortical parcellation of regions that are thought to activate as a unit for a particular task, for example the primary visual cortex (Desikan-Kiliany 2005, Honey et al., 2008). These form the nodes of the brain network model. Parcellation of the brain is performed to reduce computational complexity and to preserve the modularity of brain regions, providing a certain degree of interpretability associated with the corresponding region. The earliest parcellation of the cortex was constructed by Brodmann in 1909 using staining. Modern parcellations are constructed through a variety

of different methods from histology to using fMRI functional parcellations. There is no established standard parcellation for the cortex, as brain regions are defined by the task that they perform and there is no consensus on the exact function of certain parts of the cortex. This is especially true in the higher order areas such as the prefrontal and parietal regions. However, the Brain Network framework is thought to hold regardless of how the neural parcellations are exactly defined, and it is it is theorized that larger models with finer cortical representations simulate more realistic resting state fMRI. In this document, we utilized the Desikan-Killiany atlas throughout, in order to be consistent with older Brain Network Models and make comparisons with published results (Cabral 2011).

1.2.2 Structural Network

The *structural network* is then defined by counting the number of long-range white matter fibers detected with diffusion-weighted MRI between each pair of brain regions. The estimation of these structural networks has become possible utilizing Diffusion Weighted Imaging and modern tractography algorithms (Bihan & Breton, 1985, Tournier, Calamante, & Connelly, 2012). The diffusion tensor of water molecules along myelinated axon fibers is estimated at each point using imaging techniques. In order to entangle fibers that cross each other in three dimensions, spherical deconvolution techniques are used to reconstruct long tracts of the underlying white matter architecture, which is shown in Figure 1.1 (Tournier, Calamante, & Connelly, 2012). Based on the parcellation, the number of tracts between regions of interest are then calculated. The resulting matrix is usually symmetric but then is normalized by dividing by the surface area of the receiving cortical surface, which results in a slight asymmetry of the matrix (Honey et al., 2008).

1.2.3 Dynamical System

The dynamical system of the Brain Network Model represents how the change in neural activity in the lth region of interest $\dot{x}_l$ is modelled in relation to the rest of the network and its own activity. The first term represents the network component which is described by a function F that depends on its own activity, activity in its neighbour x_j, and the physical properties of the fiber represented by the vector ρ (i.e., the number of fibers between regions, the delay in propagation). The second term consists of a function G that represents external input, whose activity is represented by a k-dimensional vector u representing all sub-cortical and sensory inputs, and the vector π representing again the physical properties that project these inputs (i.e., thalamic tracts into cortex). The last term represents noise from the neuronal populations or from omitted higher order terms from the network equations.

$$\dot{x}_l = \sum_{j \in Neighbors\ of\ l} F(x_l, x_j, \rho_{ij}) + \sum_{k \in \text{Task inputs}} G(u_k, \pi_{lk}) + N(0, \sigma)\ Eq.1$$

For resting state activity, the assumption is that u_k (t) = 0 ∀ t and the first term dominates the change in activity.

1.2.4 Firing Rate Model

The function F for example can be as simple as the linear Firing Rate model

$$\dot{x}_l = -x_l + k \sum_{j \in Neighbors\ of\ i} w_{ij} * x_j\ \ Eq.2$$

where wij represents the number of fibers between i and j, and k represents the global coupling parameter. The firing rate model, is a linear model and can be represented as A $*$ x. The matrix A is the graph Laplacian (A = k(W – I)), where k is chosen to be less than 1/largest eigen value of A and SN is the structural matrix as measured through tractography using diffusion tensor imaging (Methods section X) (Hagmann et al., 2008). The dynamical system does not become unstable due to all of its eigenvalues being less than zero when k is set to be less than 1/(largest eigen value) of the structural network (Mesbahi & Egerstedt, 2010). Therefore, under zero input the network propagation dies out over subsequent timesteps and the dynamics collapses into the well-studied consensus equation where the dynamics decay to a mean value. We utilize the Firing Rate model in all analysis and it represents our baseline BNM.

1.2.5 Non-linear Model

More complex non-linear models have been developed that contain more state variables for each neural population and more complex dynamics to describe the interaction between these variables coupled with the network. In Sanz Leon et al., 2015, one of the first to compile a list of different BNM, there are six distinct BNM described in great detail using vastly different equations and state variable to describe the dynamics. In general, the BNM usually have an exponential decay term where the lack of input causes the mean activity of a region to decay, a network term on how other regions influence the current nodes dynamics and an oscillator term. In this thesis, we simulate two well-known nonlinear models, the Kuramoto oscillator and the Wilson Cowan Model (Cabral et al., 2011, Sanz Leon et al., 2015). Both these models are thought to simulate signals that are more representative of empirical resting state fMRI data.

1.2.6 Comparison to resting state fMRI

The full model allows task inputs to the brain, but BNMs simulations are usually compared to resting state fMRI measurements and are simulated with no external inputs. In resting state fMRI (rs-fMRI), the human participant is scanned while lying still in the fMRI scanner (Biswas et al., 1995). During resting state, task input into the sensorimotor areas is assumed to be minimized, and the resulting whole brain activity is thought to be dominated by neural populations interacting with each other through the network architecture. However, since resting state is not defined with respect to a task-triggered event, it is particularly hard to characterize since its dynamics are ever changing with time (Keilholz et al., 2017). In order to circumvent the problem of unknown timing, researchers have instead had to rely on time averaged metrics such as functional connectivity (FC), which computes the correlation between all the nodes of the brain network over a long period of time. These metrics were then subsequently used to evaluate the simulations as shown in Figure 1.1.

It should be noted here that BNMs are not the only generative model to explain rs-fMRI. A continuous approach of modeling the cortical areas as a surface known as the Neural Field Model (NFM) produces similar results (Jirsa 2007, Sanz Leon et al., 2015). Theoretically, BNMs become NFMs in the limit when the brain parcels become small enough to approximate a continuous surface. In practice, BNMs are usually employed due to the smaller number of variables needed for coarser representation of the network. Regardless, both theories reproduce aspects of the rs-fMRI signal as a product of underlying neural populations coupled with a macroscale cortical connectivity model.

1.3 Applications

BNMs are gaining the attention of the scientific community as a possible generative model that could explain the rs-fMRI signal, because of their potential in clinical and behavioral fMRI studies. Generative models for whole brain activity from individual neural populations are sought after since they would provide a link between modeling local changes to neural populations to resulting macro scale network activity. The models themselves utilize latent variables such as mean firing rate that can be meaningfully manipulated using Deep Brain Stimulations (DBS) on targeted local populations. Currently there is one clinical trial with the use of BNM in modeling Epilepsy and predicting and preventing seizures (Jirsa et al., 2014). The goals listed on their grant are to use BNM to 1) model the spatial temporal propagation of the epilepsy, 2) identify a brain region where electrical stimulation would result in a seizure, and 3) model the effects of surgical resection of brain areas used to recover healthy brain dynamics (Ritter et al., 2013). BNMs have also been used in Parkinson's Disease (PD), where they are useful in modeling network changes caused by localized damage to the Basal Ganglia by PD. Another clinical study in PD, has also simulated the effects of a DBS implant, used BNM to model in ON and OFF stimulation on the measured rs-fMRI activity (Saenger et al., 2017). In these clinical applications better models, would allow us to safely test and vary parameters such as electrode placement and stimulation parameters without going through the dangers of surgery. Moreover, neural pathology is often highly variable and different patients respond differently to treatment and it is imperative to be able to identify non-responders. Generative models such as BNM that are built from individually measured DWI and

structural T1 imaging, might represent a step in the right direction in building models that are sensitive to individual differences.

BNMs are also being used in order to aid interpretability during task studies. Traditional task studies use rest activity between task blocks as a linear regressor, modeling the rest periods as flat lines when the task input was not present. Resting state fMRI is rarely ever constant, and therefore it confounds the interpretation of the task response. Using a more complex model for resting state fMRI, might improve our approximation of task variables of interest especially to behavioral variables that are of clinical interest. BNMs have modeled task input with respect to the network as a control affine system where the task input adds to the current network dynamics (Kim & Basset 2019). Models that have solved for the task amplitude while simulating network resting state activity have had more success in matching with simultaneously acquired behavioral data (Ritter et al., 2019). This adds evidence to the notion that BNM are key to understanding how modular components interact via the network.

1.4 Gap

While there is a consensus that a complex network model provides a mechanistic explanation for coordinated whole brain activity, it is challenging to determine which approximation model and parameterization fits the rs-fMRI well enough for its intended application and is sensitive enough to model any desired clinical differences. BNMs differ in their description of the node dynamics, which can be represented with many different functions (F) and under different parameterizations, ranging from incredibly complex biophysical models down to very simple systems that use a single variable to describe the

average activity of the neural mass coupled with the structural network. The simpler models have fewer parameters, are easier to fit to the data, and are more computationally efficient, but complex models are better able to characterize the non-linear properties of the observed dynamics in the rs-fMRI data (Hansen et al., 2014, Sanz Leon et al., 2015). There is no way to directly compare these different models to the measured rs-fMRI data, as there is no temporal event to synchronize all the systems. Early studies used FC analysis in order to compare averaged properties of long simulation of the models with the measured data. However, using FC analysis on rs-fMRI has shown numerous shortcomings, especially in characterizing changes that occur over short periods of time, and more complex methods have been proposed in order to better characterize resting state dynamics (Shakil, Lee, & Keilholz, 2016). Moreover, numerous BNMs of differing complexity are all able to reproduce some aspects of FC, suggesting that the time-averaged properties are insufficient for distinguishing between models and parameterizations (Cabral et al., 2017). The main goal of this thesis is to improve our ability to recapitulate features of rs-fMRI with BNMs, while addressing these gaps in 1) selecting between different models and parameterizations and 2) comparing them with the measured rs-fMRI data. We first establish dynamic metrics that are better at differentiating between BNMs simulated with different dynamical systems and parameterizations. Moreover, these metrics represent relevant features of rs-fMRI and provide landmarks that should be reproduced by any generative model. Then we address the problem of synchronization of the models to the data, by training a Machine Learning oracle to estimate the initial conditions of the BNM for a given rs-fMRI data point, allowing for a direct comparison of how the BNM and rs-fMRI evolve over time. Finally, we show that the trained Machine Learning component

can be used as its own generative model to synthesize trajectories and that it is able to reproduce dynamic metrics observed in rs-fMRI which were not reproduced by the traditional BNMs.

1.5 Organization of the Thesis

In Chapter 2, we discuss using dynamic analysis techniques to identify better metrics to compare BNMs with each other and with measured rs-fMRI data. This work was published in 2019 Network Neuroscience (Kashyap & Keilholz 2019). As functional connectivity provides only a time-averaged characterization of rs-fMRI data, newer methods focus on more complex dynamic metrics (Keilholz et al., 2017). These metrics characterize more complex transient aspects of rs-fMRI signal that are thought to originate from neural sources. We hypothesized that these dynamic metrics would be better in differentiating between BNMs as they thought to arise from network interactions. We therefore simulated two widely-used BNMs, the simpler linear Firing Rate Model and the more complex Kuramoto oscillator model, and applied six state of the art dynamical analysis techniques: FC, Power Spectrum, Point Process, Quasi Periodic Patterns, K-means analysis, and Recurrent Quantification analysis. In addition, we artificially set the parameters of the BNMs in operating regimes where they are also known to be further from empirical rs-fMRI and tested the effectiveness of the metrics to distinguish between poorer and better parameterizations. We concluded that the dynamic metrics that operated over short time frames were the best at differentiating between models and were a harder metric for the simulated models to reproduce. The time averaged metrics such as FC that operated over longer periods of time were reproduced by several BNMs, and under that metric the simpler firing rate model looked better than the more realistic Kuramoto model. Both the

established FC metric, as well as our newer dynamical metrics are then used to characterize the performance of other generative models in this thesis.

In the third Chapter, we describe using a novel Machine Learning approach known as Neural Ordinary Differential Equations, to solve the issue of synchronization. Using a recurrent neural network, we train the Machine Learning oracle to solve for the distribution of the initial conditions of a candidate BNM with respect to the observed data. After sampling to generate an initial condition, we integrate our brain network equations and then compare how the predicted trajectory compares with future measurements of the rs-fMRI data. Using this framework, we showed that we can differentiate correctly between candidate dynamical systems for a dummy spiral dataset in order to find the best model for the data. Our hypothesis is that when we use the differences in trajectories of candidate BNMs and the measured rs-fMRI data, we can better differentiate between models that try to recapitulate rs-fMRI. In order to test this, we perturbed the structural matrix of the BNM with noise and showed that the original BNM without any noise had a trajectory that matched the resting state evolution better than the those BNMs with noisy structural matrices. Therefore, we establish this technique as a valuable tool to compare how well two even slightly different BNMs are able to fit a given dataset.

In the fourth Chapter, we show that combining a RNN network with a BNM allows the generation of simulated brain data that closely approximates empirical rs-fMRI. The network is initialized by a single observation and is simulated for long periods of time, combining the prior information held in the RNN with the constraints of the particular BNM. We compared the data generated by the trained network to that obtained with traditional BNMs using the metrics that were determined in the second Chapter. We

showed that our new generative model is for the first time able to reproduce more complex transitions such as K-means trajectories and QPPs that were observed in empirical rs-fMRI but which were largely lacking in the BNM simulations.

Finally, in the last Chapter, we explore how to build on our work in the future to translate these models in applications that are in demand in neuroscience and clinical studies. We first explore how our initialization methods might translate into using BNMs in order to extract relevant behavioral parameters during task blocks. Lastly, we also explore how this could translate in Parkinson's Disease to model the effects of stimulation with a DBS electrode.

CHAPTER 2.

DYNAMIC PROPERTIES OF SIMULATED BRAIN NETWORK MODELS AND EMPIRICAL RESTING-STATE DATA

SUMMARY

Motivation: In order to explain spontaneous macroscale brain activity as measured via resting state functional magnetic resonance imaging (rs-fMRI), activity of populations of grey matter neurons have been simulated as interactions through long range white matter connectivity. These network models, known as Brain Network Models (BNM), have had some success in reproducing properties seen in resting state fMRI (rs-fMRI), including summary metrics such as functional connectivity. However, the simulations using different underlying equations to model the interaction between network elements result in similar static summary metrics, making it hard to distinguish which simulation best recapitulates measured rs-fMRI data. Therefore, it is important to identify metrics that are more informative in distinguishing between different brain simulations and aid in the process of finding a model to accurately represent the dynamics of macroscale brain activity.

Hypothesis: In order to better distinguish how well different BNMs recapitulate rs-fMRI, we hypothesize that dynamic metrics will exhibit greater differences when applied across different BNM simulations than static summary metrics.

Approach: We simulated two commonly used BNMs with different differential equations to model the activity at each node, the Kuramoto model and the simpler linear Firing Rate Model, and generated 30 simulations of 15 minute of timeseries data at the same sampling

rate as the measured rs-fMRI whole brain network data (Section 2.2.3). From previous literature, the more complex Kuramoto model is thought to recapitulate rs-fMRI dynamics better than the linear Firing rate Model. In addition, each model was simulated across three different parameter settings. One of the conditions was biased according to previous literature findings while the other two were set such that the network coupling was either higher or lower than the empirically determined regime from previous literature. The six different models and the empirical data were then subjected to six commonly used analysis techniques from fMRI, including summary statistics, such as functional connectivity, and other analysis techniques that examine the signal for structure in different spatial and temporal scales. A full list of the six techniques include 1) functional connectivity (FC), 2) power spectrum analysis, 3) point process or coactivation rates (section 2.2.6.1), 4) quasi periodic pattern analysis (QPP) (section 2.2.6.2), 5) K-means analysis of short windowed functional connectivity matrices (K-means) (section 2.2.6.3), and 6) recurrent quantification analysis (RQA) (section 2.2.6.4).

Metrics and Evaluation: Over the span of models and parameterizations, we tested each of the six metrics on its ability to distinguish between models and the empirical data.

To evaluate performance on summary metrics such FC analysis, we followed the approach outlined in previous literature and measured the correlation between FC matrices of the simulated data and the empirical data. Similarly, for the power spectrum analysis, we measured the 1/f slope and compared the models and the empirical data with each other (section 2.3.1). For the dynamic metrics, we based our evaluation on how they were applied and quantified on empirical data in previous literature. For the QPPs, we compared the distribution of the occurrence of the QPP over the course of a scan/simulation, duration

of the QPP, and which brain regions were involved during the pattern's evolution (section 2.3.3). After performing the K-means analysis and keeping the number of clusters constant across the different simulated and measured datasets, we measured the distance between cluster centers, the distance between the cluster centroids of the same dataset, the dwell time in each cluster, and the state transition matrix (section 2.3.4). For the RQA analysis, we calculated recurrence plot of the simulated and measured signals, and then compared their distribution of entropy length, the recurrence rate, and the total time length of trajectories across different simulations/ measured scans (section 2.3.5). For the point process coactivation rates, after generating the number of coactivation rates between regions we measured the distance between the empirical and simulated point process matrices using correlation (section 2.3.2).

Assumptions: Several assumptions were made in the simulations and analysis of the models (section 2.4.7). For the more complicated metrics there are numerous hyperparameters, like the number of states in the K-Means clustering, and instead of exploring parameter space we used established values from the rs-fMRI literature. For the models, there were also many parameters that we could have varied, i.e., time delays, simulation precision, Balloon Windkessel Model, but we largely used values that were established in literature and only varied parameters such as global coupling that were known to cause changes relating to the structural network and had been actively explored by the scientific community in order to match the empirical data.

Conclusion: The static summary metrics such as FC, power spectrum analysis, and point processes showed less variability between the two different models and three different parameterizations than the metrics that quantified transient changes to the signal such as

K-means, RQA analysis or QPP analysis. The K-means metric was perhaps the best metric, as the correctly parametrized Kuramoto model showed the most similar results to rs-fMRI data, and metrics such as the centroid distance increased linearly with the parameterization variable. Moreover, the simpler Firing Rate model could not produce the dynamics described by the resting state K-means analysis as shown in previous literature (Cabral et al., 2017). The RQA metric also showed linear variability to changes in the parameterization and showed that the Kuramoto model was closer to rs-fMRI data than the Firing Rate model due to the presence of more frequent varied trajectories compared to the Firing Rate model. The QPP analysis also showed linear variability to changes in the parameterization and but there was no difference between the Kuramoto and the Firing Rate in terms of the actual trajectory estimated. The summary metrics such as FC showed the Firing Rate Model outperforming the Kuramoto model, which is contrary to our assumption that Kuramoto is the more sophisticated model that recapitulates rs-fMRI. For simpler metrics, it is easier to bias the simpler simulation with fewer variables, although in terms of actual dynamics the Firing Rate model has none of the interesting transient features observed in the rs-fMRI time series. This suggests that in order to develop more sophisticated BNMs that recapitulate rs-fMRI, the focus should be on metrics that quantify transient changes observed in the rs-fMRI data.

2.1 Introduction

The complex activity patterns produced by the brain are critical for understanding behavior and the function of the central nervous system. To explain large scale coordination between different brain regions of interest (ROIs) during rest, task, and other behavioral paradigms, studies have used resting-state fMRI (rs-fMRI) scans and functional connectivity (FC) analysis (Smith et al., 2009). FC matrix is calculated by taking the correlation between every pair of ROI timeseries, usually over a long period of time (> 10 min). In recent years, analysis of FC data has moved beyond looking at average statistical relationships maintained over the course of a long scan (average FC) to dynamic analysis methods that assume the coordination of brain activity changes on a moment-to-moment basis (Hutchison et al., 2013; Keilholz, Caballero-Gaudes, Bandettini, Deco, & Calhoun, 2017; Shakil, Lee, & Keilholz, 2016). From a generative perspective, the anatomical connections of the brain are assumed to remain constant over for the duration of the scan so that the time-varying activity is thought to arise from the structural framework (SC) based on white matter connections (Cabral, Kringelbach, & Deco, 2017; Deco, Kringelbach, Jirsa, & Ritter, 2017; Shen, Hutchison, Bezgin, Everling, & McIntosh, 2015). Recent models have modeled the brain's activity as interactions of ROIs connected by a structural network, where the activity of each ROI is a function of the local state of processing plus the delayed activity of its network neighbors (Breakspear, 2017; Sanz-Leon, Knock, Spiegler, & Jirsa, 2015). The resulting set of differential equations form a dynamical system that can be used as a generative model to simulate activity across the whole brain.

Numerical simulations of this network of ROIs, known as the brain network model (BNM) (Sanz-Leon et al., 2015), simulate spontaneous neural activity in the absence of external stimuli. Without explicit external stimuli, as in rs-fMRI, there exists no time-locked measure or event that would allow for straightforward comparison across modalities. Instead, researchers have used measures that summarize activity throughout the brain, such as average FC, and then used the correlation between the simulated and the empirical FC matrices to quantify how similar the models are to the empirical data (Cabral, Hugues, Sporns, & Deco, 2011; Liang et al., 2015; Senden, Reuter, van den Heuvel, Goebel, & Deco, 2017). At least 12 different BNMs have successfully reproduced the most prominent features of average FC (Cabral, Hugues, Kringelbach, & Deco, 2012; Cabral et al., 2011; Cabral et al., 2017; Hansen, Battaglia, Spiegler, Deco, & Jirsa, 2015; Sanz-Leon et al., 2015; Senden et al., 2017). Since this doesn't provide any insight between the differences in models and their relationship with the empirical signal, newer studies have tried to develop more complex BNMs to reproduce certain transient features observed in resting state, such as spontaneous switching between two FC states during rest (Cabral et al., 2017; Deco et al., 2018; Hansen et al., 2015). However, many dynamic analysis methods have been applied to rs-fMRI and provide complementary views of the brain activity, so it is unclear which method should be used as a metric to show differences between our simulated BNM and the measured data (Hutchison et al., 2013; Keilholz et al., 2017, Cabral et al., 2017).

The following study compares the dynamics observed in rs-fMRI to the results of the same analysis methods applied to two BNMs. We simulate two different types of BNMs with delayed inputs, the Kuramoto oscillator model and the Firing Rate model, and then

apply six of the most common dynamic analysis and static summary techniques to compare features found in the simulated data with those found in rs-fMRI scans (Cabral et al., 2012; Cabral et al., 2011, section 2.1.3). We chose the Kuramoto and the Firing Rate models because they have been shown to be robust, have relatively few parameters to optimize, and exhibit different dynamical properties that we expect to lead to differences in analysis output (Cabral et al., 2017; Deco, Jirsa, & McIntosh, 2010). Moreover, the Firing Rate model is more simplistic in the dynamics it can reproduce, and we expect it to serve as a contrast to the more complex Kuramoto model (Cabral et al., 2017). To characterize the models and compare them with empirical data, we chose six analysis techniques that test the signal for states, repeating events, or trajectories that are representative of its higher order spatiotemporal structure. We chose these techniques in order to span the spectrum of analyzing the signal by the patterns in the spatial domain and patterns in the temporal domain. In addition, we simulated models at three different parameter settings, one in an empirical determined regime and the other two that would lead to spurious dynamics. This was conducted to test the metrics ability to distinguish correctly parameterized models from incorrectly parameterized models. This is particularly important because different parameterizations lead to vastly different set of dynamics and it is important to be able to tell which parameterizations best recapitulate the underlying biological processes.

Our approach evaluates how useful these analysis techniques in being able to distinguish between different models. We hypothesize that dynamic metrics will exhibit greater differences when applied across different BNM simulations than static summary metrics and will distinguish how well different BNMs recapitulate rs-fMRI.

Dynamic Analysis Techniques

1. **Point process or neural avalanche theory**, which models the fMRI signal as a combination of discrete neural events or avalanches. An event in an ROI is observed when the signal crosses a threshold, and then quantifies coactivation of these events between different ROIs (section 1.6.1,Caballero et al., 2010; Liu & Duyn, 2013; Natalia, Gaudes, Dryden, Francis, & Gowland, 2012; Tagliazucchi, Balenzuela, Fraiman, & Chialvo, 2012).

2. **Repeated or quasiperiodic spatiotemporal patterns (QPP),** which identifies a unique spatiotemporal pattern that is particularly prominent in the default mode network (DMN) and the task positive network (TPN). This pattern is extracted by iteratively using a spatiotemporal template of fixed length to correlate with the signal, finding the peaks in the correlation vector, and then averaging all the highest peaks to determine the next template (section 1.6.2, Belloy et al., 2018; Majeed et al., 2011; Majeed, Magnuson, & Keilholz, 2009; Thompson, Pan, Magnuson, Jaeger, & Keilholz, 2014; Yousefi, Shin, Schumacher, & Keilholz, 2018).

3. **K-means clustering on windowed functional connectivity,** which identifies discrete periods in time when the spatial patterns of correlated brain activity are relatively stable. Sliding window functional connectivity matrices are clustered using k-means in order to identify the clusters in dynamic functional connectivity (section 2.1.6.3, Allen et al., 2014).

4. **Recurrence quantification analysis (RQA),** which identifies repeated spatial signatures as a function of time (section 2.1.6.4 Webber & Marwan, 2015). In this method, the spatial pattern at each time point is correlated with the spatial pattern at all other time points, and the results are then quantified using information theory for repeated time signatures

Static Summary Metrics

1. **Functional Connectivity:** A correlation matrix between the time series activity of different brain regions. Requires a long period of time (~ 10 min) in order to converge to a stable value (Shakil & Keilholz 2014)

2. **Power Spectrum Analysis:** A power spectrum analysis of the timeseries of each individual region plotted as a distribution. Brain data even in these timescales exhibits a $\frac{1}{f}^n$ amplitude and drops of at higher frequencies.

Brain Network Models (Neural Mass Models)

1. **Kuramoto model:** A model where the trajectory of each neural mass is modeled as an oscillator and the phases of each oscillator are synchronized based on network input and perturbed by random noise (section 2.1.3). The global coupling term k is varied as a parameter.

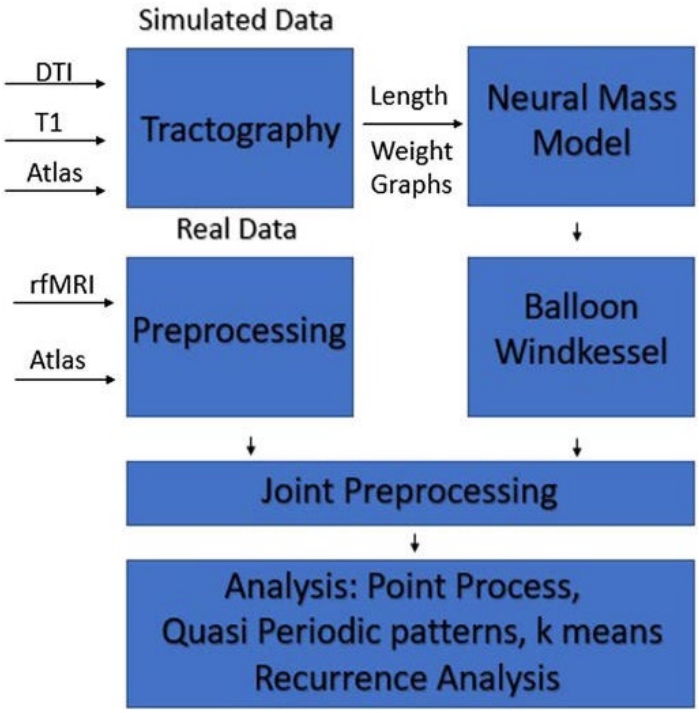
$$\frac{d\theta_n}{dt} = \omega_n + k\sum_{p=1}^{N} c_{np}\sin\left(\theta_p(t-\tau_{np}) - \theta_n(t)\right) + \sigma n(t)$$

2. **Firing Rate model:** Each neural mass is modeled by a single parameter that represents the aggregate firing rate of the population, and it decays with a certain time constant and increases its activity based on network input and random noise (section 2.1.3). The global coupling term k is varied as a parameter.

$$\tau_0 \frac{dr_n}{dt} = -r_n + k\sum_{p=1}^{N} c_{np} r_p\left(t-\tau_{np}\right) + \sigma n(t)$$

2.2 Methods

Figure 2.1 Methods General



We used DTI data to generate the length and weight matrix between ROIs of our specified atlas. Using this structural connectome, we generate data using the different neural mass models and transformed into the BOLD signal via the Balloon-Windkessel model. To compare with empirical fMRI data, scans from HCP were preprocessed and parcellated using the same atlas. The final preprocessing of filtering, global signal regression, and normalization was done jointly for all sets of data. The final data were then analyzed with each of the different dynamic analysis techniques.

2.2.1 Methods General

Our general methods are outlined in Figure 2.1 and follows the steps outlined in Cabral et al 2011. The generative arm starts with the Diffusion Tensor Imaging images, runs tractography, and estimates network graphs based on parcellation of a given atlas. The Neural Mass Models simulate activity based on this graph and the differential equations which are varied according to each model (Firing Rate or Kuromoto) and different

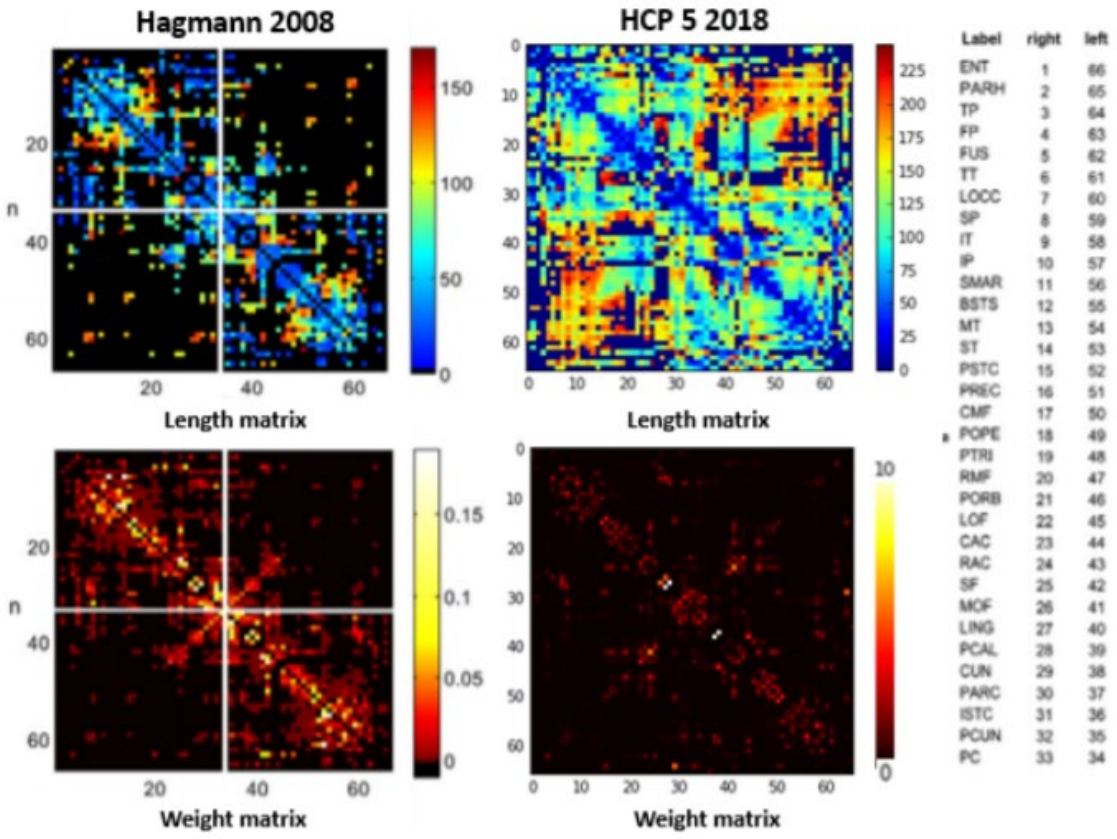
parameterizations. The simulated activity is converted to the hemodynamic response based on the Balloon Windkessel model. Separately, the resting state fMRI data is registered to standard space is then processed such that all the voxels in the brain regions is averaged according to the same atlas in order to estimate the response of the neural mass contained in that ROI. The simulated data then is collectively preprocessed according to steps outlined in Cabral et al 2011 and then analysed by the various analysis techniques.

2.2.2 Structural Connectome

The Structural Connectomes were generated from tractography, using Human Connectome Project's diffusion-weighted images (spin echo TR 5520 ms, TE 89.5 ms, flip angle 78, voxel 1.25 mm) from five random subjects and were prealigned to standard space (Van Essen et al., 2013). Together they were averaged in order to generate a mean structural connectome. Tractography was performed using the freely available software MRtrix with maximum fiber length set to 250 mm (Tournier, Calamante, & Connelly, 2012) and parcellated using the Desikan-Killiany atlas (Desikan et al., 2006). For each subject, their respective T1w images (TR 2,400 ms, TE 2.14, voxel size 0.7 mm) were aligned to the standard space; then the we transformed the diffusion-weighted images to standard space as well. Probabilistic tractography then was run between each ROI and then pruned to generate 10 million fibers. To generate the estimates for the length and weight matrices from the tractography, we used the same methodology as Hagmann et al. (2008). The comparison between our tractography and the one from Hagmann 2008 is shown below. The tractography that we are running is a newer algorithm from Mrtrix, and can estimate longer fiber connections more accurately. The length between two ROIs was defined as the average fiber length of all fibers that went between them, and the weight was the number

of fibers going between two ROIs normalized by the surface area of the receiving ROI. The atlas provides 84 cortical and subcortical ROIs, but we selected the same 66 cortical regions as in Cabral et al. (2011) for comparison to previous work. The resulting matrices are shown in Figure 2.2 (Kashyap & Keilholz, 2019). Our tractography is less sensitive to longer connections (Fornito, Zalesky, & Breakspear, 2013) and therefore the between-hemispheric connections were scaled by a factor of 4 to offset the known issue. The order of the ROIs are arranged so that the first half are from the left hemisphere and then mirrored for the right hemisphere. It gives the graphs a symmetric look, where the intrahemispheric connections are represented in the top left and bottom right, and the connections between hemispheres are in the bottom left and top right.

Figure 2.2 Comparison of Structural Connectome Data



Comparison of our structural connectome with the structural connectome published in Cabral et al 2011. Our tractography utilizes the newer tractography methods that were unavailable at that time and has set the max tract length to 250 mm which allows us to estimate the longer tracts that are between hemispheres (top right and lower left quadrants). The distances in the length matrix are mostly in agreement (corr > 0.9) for the connections that are in common between the two methods.

2.2.3 Brain Network Models

Brain network models describe the BOLD signal as the coupling of n distinct neural populations corresponding to different cortical regions. Each population is connected via a weight matrix obtained from structural connectivity that describes the strength of the connection between nodes. In general, each of these n areas are modeled by a differential equation for each node: $dn(t)/dt = f(N(t), W, L)$, where N(t) is the time series of all the nodes/ROIs, W is the weight matrix, L is the length matrix, and for given random initial conditions for n0, the time series n(t) can be solved for by using the Euler integration method (Sanz-Leon et al., 2015). The time series n(t) is the state variable and is representative of a measurable property of the neural mass such as firing rate. Some variants use more than one variable to represent the state of the neural mass, but in this paper we consider two models that only use one state variable, namely the Firing Rate and the Kuramoto models. Table 1 shows the mathematical description as well as the values of the parameters used in the simulations.

The Kuramoto model is derived from an assumption that each neural population is in a closed periodic trajectory in phase space that represents its computational processing (Cabral et al., 2011). It has been shown that it can then be modeled by a phasic oscillator that can be described by a single parameter, theta, that represents its location within a 2pi cycle. Inputs into these phasic oscillators perturb its trajectory, but it stays within its limit

cycle. Each of these oscillators couples via the network and is driven to the same angle and thus synchronizes the oscillators as a function of the difference between the angles of neighboring oscillators.

The Firing Rate model assumes that the mean firing rate of the neural populations is distributed in a Gaussian manner. This assertion is in accordance with the central limit theorem, which states that the sum of uncorrelated random processes converges to a Gaussian probability distribution, even if the individual processes are highly non-Gaussian. Inputs into this neural mass shift the mean firing rate to a higher firing rate. The mass shifted from its equilibrium tries to relax at the rate proportional to its own firing rate, keeping the system stable via negative feedback.

For each model, the differential equations were numerically integrated with a time step function of 0.1 ms for a duration of 15 min to match the length of an HCP rs-fMRI scan. The first 20 s are thrown away to avoid transient effects from initial conditions. The choices for the values for all the parameters given in Table 1 follow previous work by Cabral et al. (2012) and Cabral et al. (2011), except that the values for k are slightly different than the ones in the paper to account for differences in the structural connectivity matrix. The values were slightly smaller for the Kuramoto (13 instead of 18) because there were more numerous connections in the newer tractography. The low coupling models were simulated for the Firing Rate at k = 0.3. The Kuramoto model had a low coupling of 3. The high coupling models were simulated at k = 60 for Kuramoto and k = 0.999 for Firing Rate. We simulated 30 individual runs at parameterization values from previous studies, and 20 runs each at high and low global coupling levels. Simulations of functional connectivity and the intermediate steps with the original Hagmann matrices and the

comparisons with Cabral et al. (2011) are given in Supplementary Figure 1 (Kashyap & Keilholz, 2019).

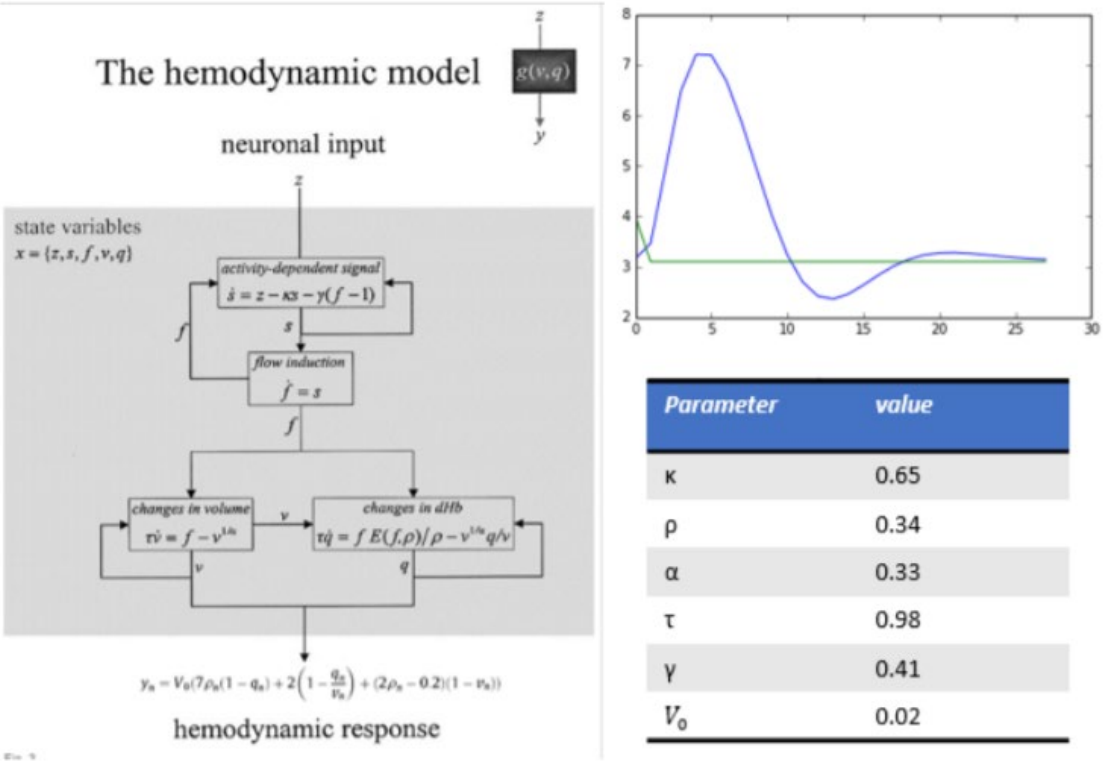
Table 2.1 Parameters of the Brain Network Models

Parameter name	Differential equations that were simulated via Euler method with a time step of 0.1 ms	State variables	C_{np} – structural network weight	T_{np} – structural network delays	σ_n – SD of Gaussian white noise	ω_n – oscillator frequency	c_1 – first eigenvalue of weight matrix	τ_0 – relaxation constant
Firing Rate	$\tau_0 \frac{dr_n}{dt} = -r_n(t) + \frac{k}{c_1} \sum_{p=1}^{N} C_{np} r_p(t - \tau_{np}) + \sigma n(t)$	r(t) – mean firing rate	K = 0.9 (scale of Cnp)	11 ms set for mean	2 rad/s	N/A	calculated	20 ms
Kuramoto	$\frac{d\theta_n}{dt} = \omega_n + k \sum_{p=1}^{N} c_{np} \sin(\theta_p(t - \tau_{np}) - \theta_n(t)) + \sigma n(t)$	θ(t) – oscillator phase	K = 13 (scale of Cnp)	11 ms set for mean	2 rad/s	Randomly initialized as N ~ (60 Hz, 2 Hz)	N/A	N/A

2.2.4 Converting to BOLD via Balloon Windkessel Model

In order to compare the neural simulated data with the hemodynamic response measured from fMRI, we have to convert the high-frequency activity down to the low-frequency hemodynamic response. This is performed with the Balloon-Windkessel model, which is a quadruple differential equation model that in a neuronal input and calculates the blood flow and blood volume and uses that to estimate the fraction of the oxygenated blood to the deoxygenated blood (Friston, Harrison, & Penny, 2003; Stephan, Weiskopf, Drysdale, Robinson, & Friston, 2007). Figure 2.3 (Kashyap & Keilholz, 2019) shows the impulse response of the Balloon-Windkessel model, which looks roughly like the canonical hemodynamic response function. We used the same constants for our Balloon model as those given in Friston et al. (2003). After passing the output of the BNMs through the Balloon-Windkessel model, it was then downsampled to the same sampling rate as the rs-fMRI data (0.72 s).

Figure 2.3 Hemodynamic Response Function



Parameter	value
κ	0.65
ρ	0.34
α	0.33
τ	0.98
γ	0.41
V_0	0.02

The equations are a reprint of (Stephan et al., 2007) used in Dynamic Causal Modeling to relate neuronal signal to the output. The model implemented is a set of five differential equations as shown in the diagram with the given variable values. An impulse response is plotted top right and is similar to the canonical Hemodynamic Response function in both its shape and duration.

2.2.5 Pre-processing rs-fMRI

For the rs-fMRI data we used 30 individual HCP scans (gradient echo EPI, TR 720 ms, TE 33.1 ms, flip angle 52, voxel 2 mm) that are each roughly 15 min long. The data came from the minimally processed pipeline and then were ICA denoised using the 300 ICA vectors that HCP provides. We then applied the same Desikan-Killiany atlas as used in the tractography onto the data and obtained the mean time series for each ROI. The ordering of the ROI is shown below in Table 2.2. From then on, the same processing pipeline was applied for the simulated data and the real data, in order to keep the processing as similar as possible. These steps in order were z-scoring

each time series, then band passing filtering the signal from 0.01 to 0.25 Hz, then global signal regression using a linear regression model, and then applying a final z-score step. These steps were selected in accordance with Cabral et al. (2011).

Table 2.2 Table of Regions of Interest

Abbreviation	ROI/ Brain Region
ENT	*Entorhinal cortex*
PARH	*Parahippocampal cortex*
TP	*Temporal pole*
FP	*Frontal pole*
FUS	*Fusiform gyrus*
TT	*Transverse temporal cortex*
LOCC	*Lateral occipital cortex*
SP	*Superior-Parietal cortex*
IT	*Inferior temporal cortex*
IP	*Inferior-Parietal cortex*
SMAR	*Supramarginal gyrus*
BTST	*Bank of the superior temporal sulcus*
MT	*Middle temporal cortex*
ST	*Superior temporal cortex*
PSTC	*Postcentral gyrus*
PREC	*Precentral gyrus*
CMF	*Caudal middle frontal cortex*
POPE	*Pars opercularis*
PTRI	*Pars triangularis*
RMF	*Rostral middle frontal cortex*
PORB	*Pars orbitalis*
LOF	*Lateral orbitofrontal cortex*
CAC	*Caudal anterior cingulate cortex*
RAC	*Rostral anterior cingulate cortex*
SF	*Superior frontal cortex*
MOF	*Medial orbitofrontal cortex*
LING	*Lingual gyrus*
PCAL	*Pericalcarine cortex*
CUN	*Cuneus*
PARC	*Paracentral lobule*
ISTC	*Isthmus of the cingulate cortex*
PCUN	*Precuneus*
PC	*Posterior cingulate cortex*

2.2.6 Dynamic Analysis Techniques

To compare the dynamics of the rs-fMRI signal and the BNMs, we selected analysis techniques that are commonly used and characterized the signal at different spatial and

temporal scales. Table 2.3 shows a quick comparison of the different techniques that were applied.

Table 2.3 Dynamic Analysis Techniques

Analysis technique	Description	State variable	Spatial scale	Temporal scale
Point process/ neural avalanche	FC is driven by discrete events in rs-fMRI when different ROIs coactivate together	Level of activity of a single ROI	Very similar to average FC and the SC input	Coordination is seen in the shortest scales (~ 1 s)
Quasiperiodic pattern	A specific spatiotemporal pattern observed in rs-fMRI that repeats itself over the length of the scan	Phase during the spatiotemporal pattern template	Specific nodes go through a sequence of activation and inactivation	Pattern (20 s) Occur (2/3 per min)
Recurrent quantification analysis	A sequence of similar ROI activity and/or trajectories that appear to repeat over time	Level of activity across the whole network at a single point in time	Repeating firing patterns over time	Medium (~ 10 – 20 s)
K-means clustering of FC matrices	Multiple stable FC states that transition between each other	Windowed FC matrices at ~ 45 s	Local networks embedded in the larger SC that coordinate processing for a stable period of time	Longest (45 s–1 min)

2.2.6.1 Point Process

The point process assumes that activity in an area triggers neural avalanches in regions that are involved in the information processing (Tagliazucchi et al., 2012). The signal is only interpretable at either its high levels of activation or very low levels of activation when it is coordinating information transfer with other elements in the network. Later models explicitly write out the mathematical formulation using impulse response and solve for a sparse representation of these coactivation patterns, which are thought to be unique computational trajectories across the brain (Karahanoglu & Van De Ville , 2015; Liu & Duyn, 2013). But in this analysis, we use Tagliazucchi's methodology by quantifying when different ROIs cross the same threshold over time. We implemented this approach by recording when the activity at a certain ROI crosses a certain threshold and then counting how many other ROIs cross the same threshold within three time steps (0.72 s) of the original crossing. We normalize the co-occurrence rates to get a fraction by dividing by the total number of crossings at each ROI. We applied this analysis with two different thresholds, one at the mean of the signal and one at 1 standard deviation away,

which for our normalized signals were at 0 and 1, respectively. Prior work that has shown that average functional connectivity is primarily driven by coactivation events (Tagliazucchi et al., 2012).

2.2.6.2 Quasiperiodic patterns

A second approach examines the quasiperiodic patterns of BOLD signal propagation over the course of the scan. The QPP algorithm identifies the most prominent repeating spatiotemporal pattern in the signal (Majeed et al., 2011; Majeed et al., 2009). In brief, the algorithm chooses a random chunk of the rs-fMRI data (20 s of data) and correlates it with the entire scan (Majeed et al., 2011). Time points with high correlation to the random chunk indicate repeated occurrences and are averaged together to form the new template. This process is iterated until the template converges. Since we use a random seed point as the original template, repeated runs of the algorithm produce QPPs with different phases. Therefore, in order to compare the patterns from the rs-fMRI and the simulated models, the QPP was circularly shifted to the point where maximum correlation occurred.

2.2.6.3 K-means analysis

Sliding window correlation followed by k-means clustering was applied to examine the brain states and transitions in each set of data (Allen et al., 2014). Using a sliding window length of 60 × 0.72 s, the Pearson correlation was calculated pairwise for all ROIs. The window was then advanced by one time point and the process was repeated until the window reached the end of the scan. This value is around the range used in previous work (Allen et al., 2014). Correlation values were Fisher-transformed to better approximate a normal distribution and the k-means algorithm was applied to cluster the data into seven

groups using Manhattan distance based on previous studies (Allen et al., 2014). Clustering was repeated 30 times, and the best resulting clustering was chosen based on minimizing the total distance from the cluster centroids and the feature vectors in order to mitigate the effects of randomly choosing the centroid locations.

2.2.6.4 Recurrent Quantification analysis

Recurrent analysis was performed by calculating correlation of the spatial pattern of activity pairwise across all time points. We then thresholded the values at 0.3, based on literature search, and created recurrent plots (Bassett, Nelson, Mueller, Camchong, & Lim, 2012; Cabral et al., 2014). These metrics were calculated using freely available MATLAB toolbox (Ouyang, Li, Dang, & Richards, 2008), and their distribution for each type of data was plotted. Recurrence rate, entropy rate, and average diagonal length were measured. The recurrence rate is the rate that similar states occur throughout the scan, as seen in Equation 1. Entropy rate quantifies the difference between repeated states, as seen in Equation 2. The average diagonal length, Equation 3, measures how long these trajectories occur. Collectively they give us an insight into how often similar states occur, how different they are from each other, and how long each of these spatiotemporal trajectories persists.

$$RR = \frac{1}{N^2}\sum_{i,j=1}^{N} R(i,j), \text{ where } R \text{ is the recurrent plot.} \quad (1)$$

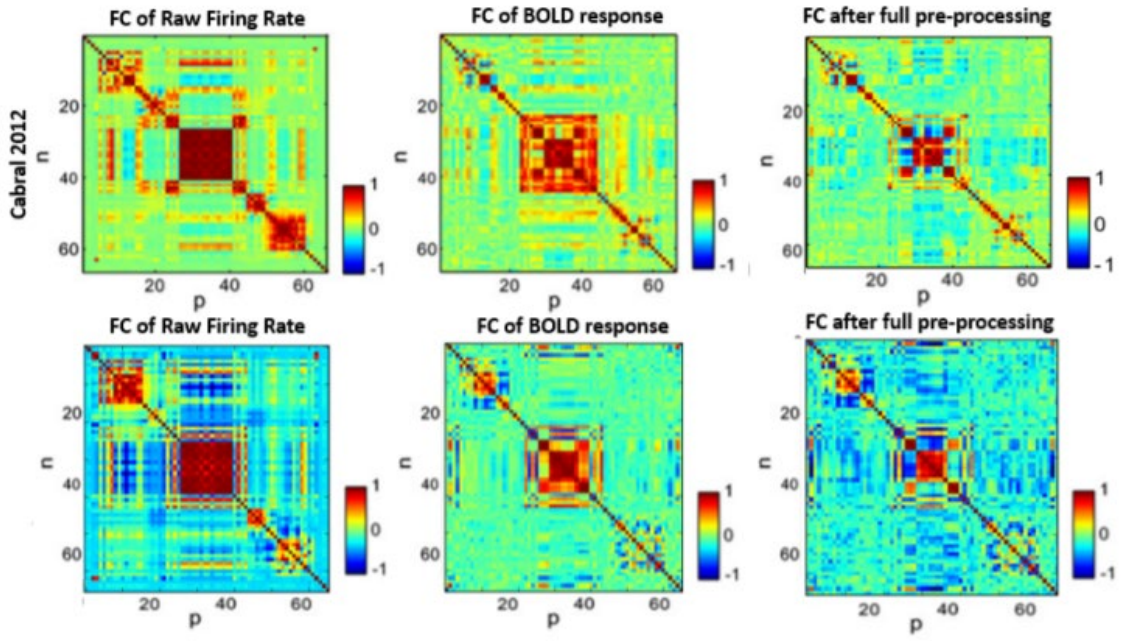
$$L = \frac{\sum_{l=lmin}^{N} lP(l)}{\sum_{l=lmin}^{N} P(l)}, \text{ where } P(l) \text{ is the frequency of diagonal length } l. \quad (2)$$

$$ENTROPY = -\sum_{l=lmin}^{N} P(l)\ln(P(l)), \text{ where } P(l) \text{ is the frequency of diagonal length } l. \quad (3)$$

2.3 Results

2.3.1 Reproduction of Previous Literature Results

Figure 2.4 Comparison of our methodology with the original BNM Kuramoto Model



A comparison between all the intermediate states of generating the simulated BOLD signal for the Kuramoto fast oscillator model. The top row is a screenshot from Cabral 2012 paper and the bottom is our reproduction of it using our own methodology. The left most panel represents the functional connectivity matrix calculated from the raw output of the Kuramoto. The middle panel represents the functional connectivity from the output of the Balloon Windkessel model. The rightmost panel represents the functional connectivity using the post processing steps cited in the paper. These models were generated using the 2008 Hagmann structural connectivity which was subsequently replaced by our tractography.

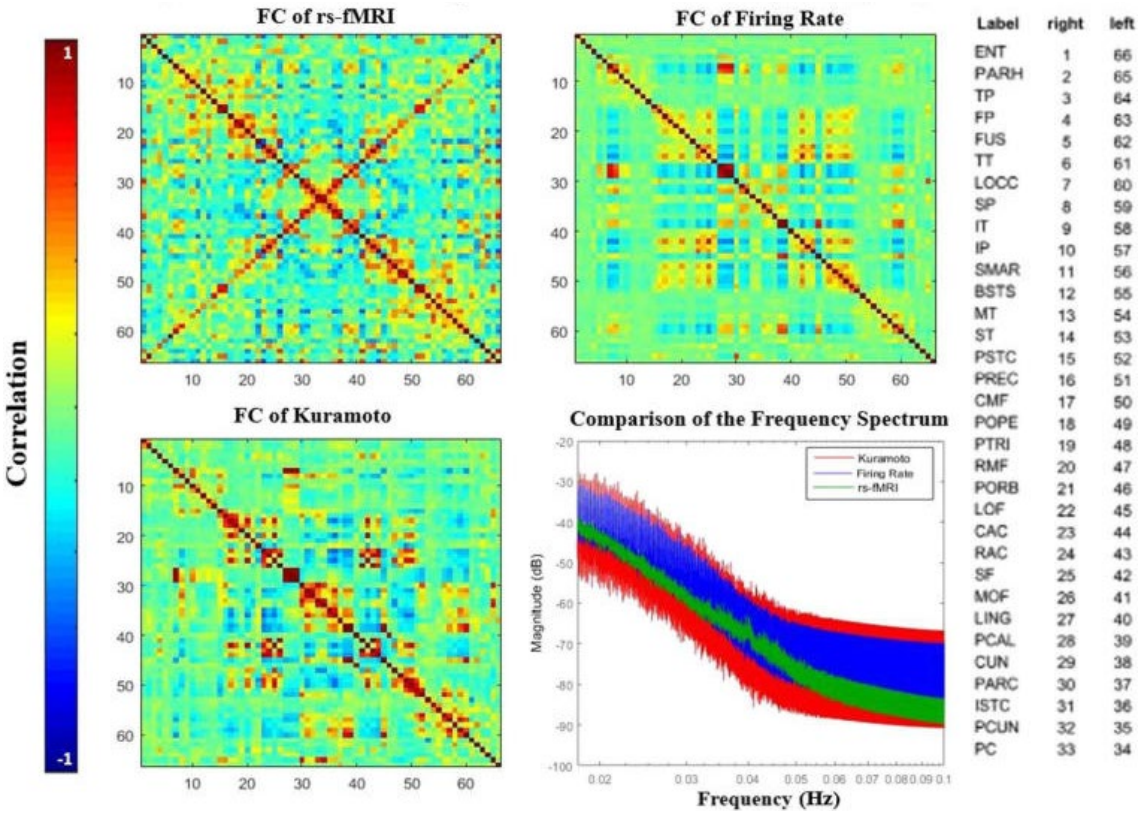
We first demonstrate that the simulated models reproduce common metrics in brain network modeling: average FC and power spectrum. Since our methodology and parameter values are similar to those described in Cabral et al. (2011) and Cabral et al. (2012), we first demonstrate the reproduction of their results using their tractography but using our

pipeline as shown in Figure 2.4. The ordering of the ROIs seen in the figure is shown in Table 2.2 (section 2.5) and is from Cabral et al. (2011). We debugged our methodology in order to match each of the figures at different stages of the pipeline (after the Brain Network Model section 2.2.3, the Hemodynamic response model section 2.2.4, and after the final preprocessing steps section 2.2.5).

The rest of the results use the tractography from 2018 (as in Figure 2.5) which show a much more coordinated response between the hemispheres consistent with the fact of the inclusion of longer tracts in the new tractography. To quantify the similarity between the simulated FC matrices and the empirical FC matrix, we calculated the correlation between the two, a method that is extensively used in previous studies (Cabral, Hugues, Sporns, & Deco, 2011; Cabral et al., 2017; Senden et al., 2017). Correlation was 0.37 between Kuramoto and rs-fMRI FC matrices, and 0.5 between Firing Rate and rs-fMRI FC matrices. These are in the range of values reported in earlier literature in other BNMs [0.3, 0.7] (Cabral, Hugues, Kringelbach, & Deco, 2012a; Senden et al., 2017). As expected, the correlation values between simulated and empirical FC are higher values at the correctly biased value than at very high and low global coupling levels (Figure 2.6 top row). Power spectra were calculated for each ROI independently, then averaged (Figure 2.5, bottom right). When plotted on a log-log plot, the BOLD signal has a characteristic 〖1/f〗 ^n linear slope. The power exponent n has been reported in literature as 0.88, comparable to the 0.9 measured here for our empirical rs-fMRI data (Bullmore et al., 2000). The empirical slope falls well within the distribution of the simulated power spectrums. The two simulated models had a slope of 0.74 (Kuramoto) and 0.7 (Firing Rate) before

preprocessing, comparable to a previous report of 0.78 using a different BNMs but not as good as the current best of 0.91 (Ritter, 2017; Ritter, Schirner, McIntosh, & Jirsa, 2013).

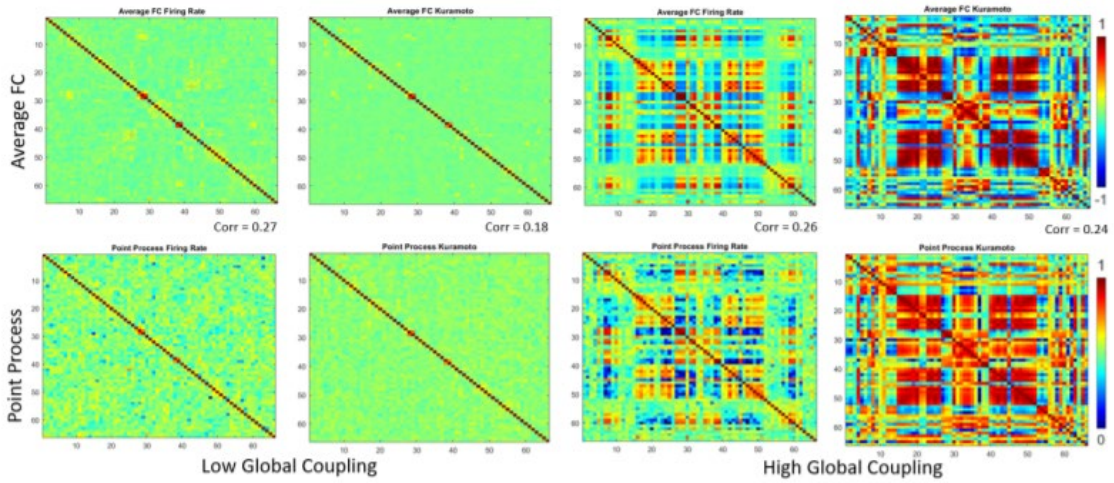
Figure 2.5 Comparison of Average FC and the Frequency Spectrum



Comparison of the average functional connectivity between the rs-fMRI signals and the two simulated models. Correlation between the matrix for empirical data and the Firing Rate simulation is 0.5; correlation between empirical and Kuramoto matrices is 0.37. Both modeled matrices and the empirical data exhibit similar structure such as the coordination between hemispheres, which can be seen in the symmetry of the matrix. The mean frequency spectrum of all ROIs is plotted (bottom right) and shows

that the real signal falls within range of both models. All power spectra exhibit a $\frac{1}{f}^n$ trend.

Figure 2.6 Average FC and Point Process across Parameters



Changes in average functional connectivity and point process for different global coupling parameters. The coupling changes the dynamics from purely noise driven to purely network driven processes. The functional connectivity were both much lower than in the correctly biased Figure 2.5 (0.15-0.3).

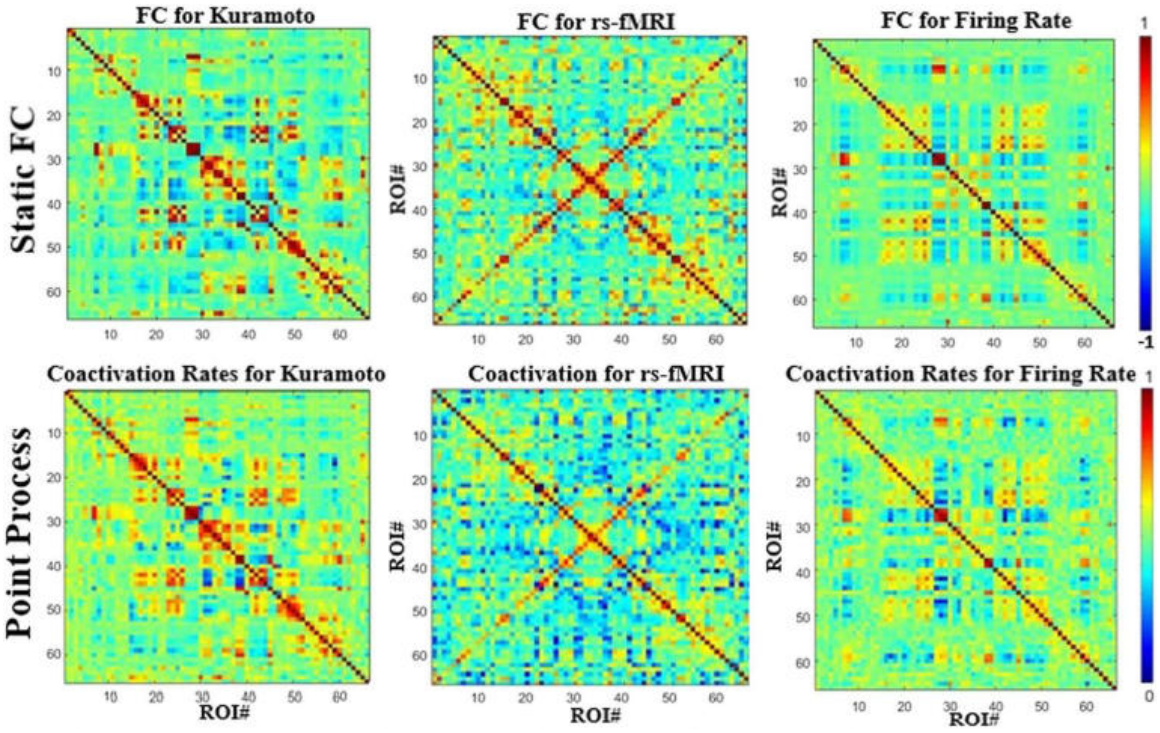
2.3.2 Point Process and Coactivation Rates

In coactivation analysis based on the point process approach, all ROIs that cross a certain activation threshold (see Methods; Tagliazucchi et al., 2012) are examined at each time point to identify coactivation patterns. The bottom row of Figure 2.7 shows the coactivation data obtained for the Kuramoto simulation, the Firing Rate simulation, and empirical rs-fMRI data. Each value in the matrix represents the fraction of co-occurrences between two ROIs. The matrices are compared with the respective average FC matrices (Figure 10, top row), and all three signals show a high degree of correlation (>0.9) between the two different analysis techniques. This is because average FC can be calculated by a handful of events, as shown by Tagliazucchi et al. (2012). Moreover, from Figure 2.6 point

process/coactivation rates are identical to average FC at different parameter settings, further suggesting they are measuring similar structure in the data. The Firing Rate

Figure 2.7 Point Process across Brain Network Models

Kuramoto model, which is similar to the results observed with the average FC analysis.



Coactivation rates (point process) between the different modalities compared with those from static or average functional connectivity. The resulting maps are almost identical between different modalities and have a correlation of over 0.9 between each respective dataset.

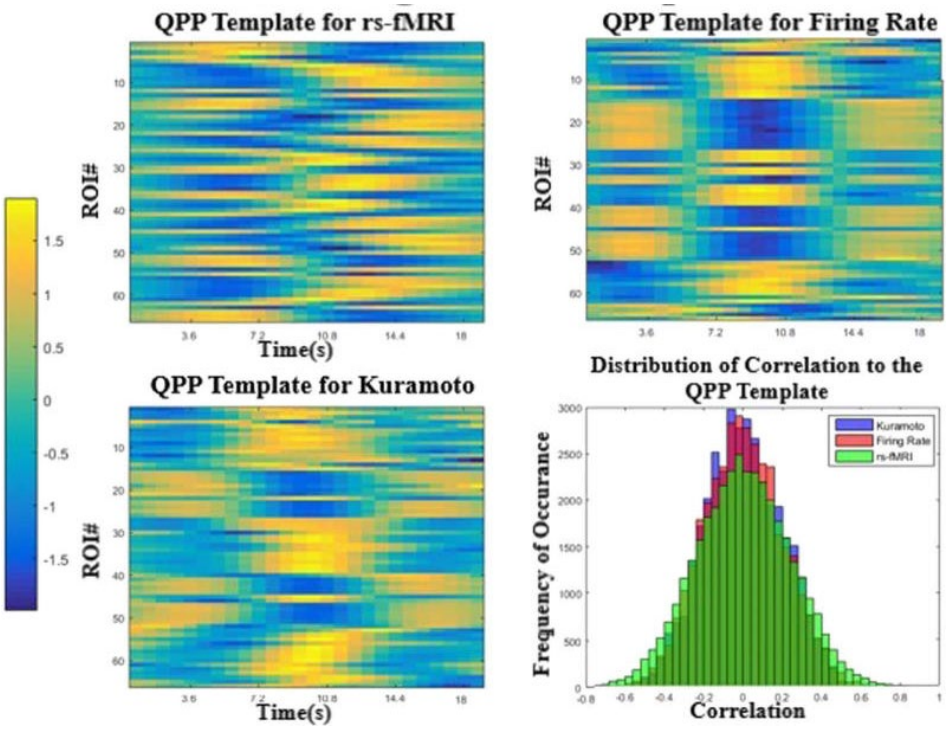
2.3.3 Quasiperiodic Pattern Algorithm Comparison

The quasiperiodic pattern (QPP) finding algorithm estimates a recurring spatiotemporal pattern that occurs throughout resting and task states. It consists of a characteristic pattern dominated by the activation and inhibition of the regions that

correspond to the default mode network (DMN) and task positive network (TPN) in a specific temporal sequence (Majeed et al., 2011; Majeed et al., 2009; Yousefi et al., 2018). The QPP templates obtained from the real data and from each simulation are shown in Figure 2.8 in a simplified format, where the color bar shows the level of activation or deactivation in each ROI as a function of time. For better visualization, please see the videos (Kashyap & Keilholz, 2019) that show the pattern as it evolves over a surface representation of the brain. The pattern in the rs-fMRI data is consistent with the QPP templates obtained previously (Majeed et al., 2011; Yousefi et al., 2018).

The QPP templates from the two models are very similar to each other (Figure 2.8 top right and bottom left, correlation of 0.81), but have important differences from the empirical QPP (Figure 2.8, top left, correlation of 0.34, 0.33). In fact, the pattern in the simulated models is a simple flip between two states, where a subset of ROIs is first active and then inactive. The boxy nature of the plot is due to the spatial ordering of the ROIs that was originally defined by their subnetwork connectivity, suggesting that these subcomponents are activating and deactivating together. The QPP obtained from the real data is more complex and demonstrates time lags between areas in addition to the alternation of states, suggesting that the power in the BOLD signal cyclically flows through a certain order of ROIs. The relative lengths of the simulated and the observed patterns are different as well. The QPP from the real data is approximately 20 s in length, in agreement with previous reports (Majeed et al., 2011; Yousefi et al., 2018). In contrast, both of the models give QPPs that are ∼12–13 s in length, despite the use of identical windows and the similar frequency content of the signals.

Figure 2.8 QPP Algorithm Comparison

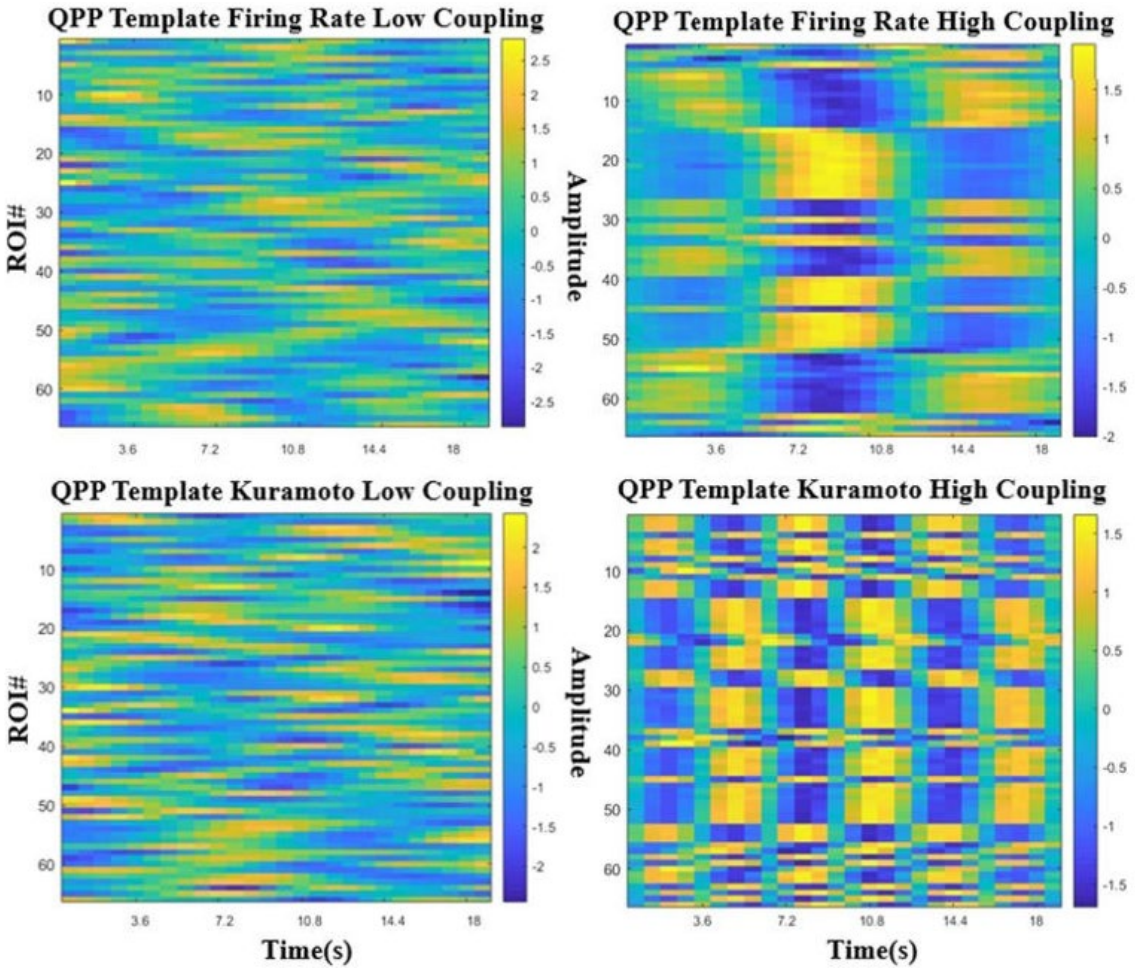


Comparison of the QPPs obtained for each model and the real data. The two simulated models (top right and bottom left) produce templates that are similar to each other but less similar to the template extracted from rs-fMRI. The correlation values to the template are plotted in a histogram (bottom right), which shows that the real signal has more extreme values than either model. All three are significantly different from each other ($p < 0.0001$) in a Komogorov-Smirnov test.

At low and high global coupling (Figure 2.9), the QPP pattern transitions from an unstructured noise-like template to a structured signal. The number of repeated patterns for a given window is dependent on the strength of the coupling parameter for the Kuramoto model, where at higher coupling the pattern is shorter and repeats more often. For the Firing Model, the template pattern emerges from unstructured noise after a certain coupling strength. The correlation vector represents how correlated the QPP template is with the scan at every time point. The distribution of the values are displayed using histograms (Figure 2.9, bottom right), in order to compare different modalities. The flatter distribution

of the resting-state fMRI shows that the template occurs more often significantly and at higher correlations in the real scan than either of the templates.

Figure 2.9 QPP Algorithm Comparison across Parameters



Comparison of the QPPs obtained for each model across different coupling strengths. The left column represents low global coupling and the right column, higher global coupling. The top row is the Firing Rate model and the bottom row is the Kuramoto model. As coupling strength increases, the pattern goes from a random unstructured signal to a highly boxlike structure.

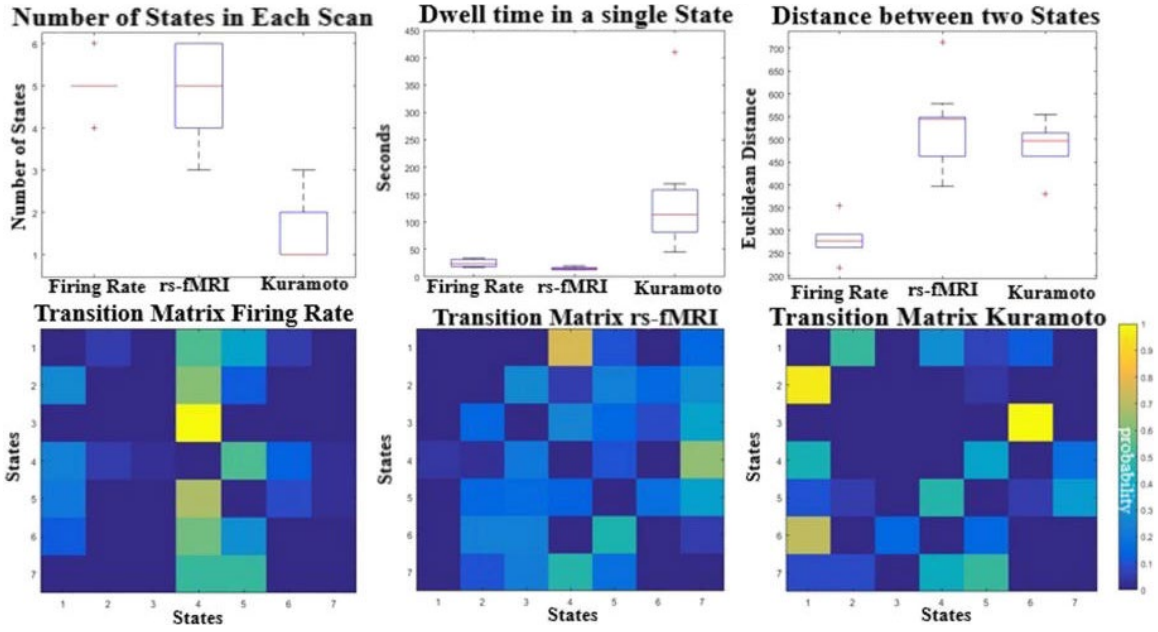
2.3.4 K-means on Sliding Windowed Matrices

To identify FC states that occur at different time points in the BOLD signal, we used k-means analysis to compare the sliding windowed FC of the real and simulated data. After k-means clustering (k = 7), we examined both the spatial composition of the resulting clusters (or states) and metrics that describe how the brain transitions between them in Figure 2.10 (Allen et al., 2014). We have also explored cluster numbers k = 8 and 9; these are shown in Figure 2.12. The results are similar across all metrics and all models. This is due to the number of actual clusters seen per individual simulation (Figure 10, right column; Kashyap & Keilholz, 2019), which is constant across cluster numbers, suggesting that the methodology is measuring the intrinsic dynamic structure seen within the data, rather than arbitrarily dividing up the segments.

The top row in Figure 2.10 quantifies for each individual scan or simulation, starting at different initial conditions (N = 30), how many of unique states are visited, how long they dwell in each state, and how far apart (L2 Norm distance between cluster centers) these visited states are on average. The Firing Rate and the rs-fMRI each have an average of five states per individual scan and transition at similar rates, but the average distance between centroids is almost twice as large in the rs-fMRI states compared with the simulated data. Visually the Firing Rate centroids look very similar (Figure 2.13), suggesting that the diversity of states encountered is still very low. The Kuramoto model produces states with similar distances between centroids as empirical data, but each instantiation has fewer states compared with rs-fMRI. Most runs (66%) result in only a single state, but under certain initial conditions the Kuramoto model can exhibit transitions between two to three different states. The model also dwells in these states longer than in the empirical rs-fMRI data and Firing Rate data. The transition matrix for the empirical

data shows that transitions are more evenly distributed between states than in the simulated data (Figure 2.10, bottom row). The empirical rs-fMRI data have more transitions between states than in either simulated model. The Kuramoto and the Firing Rate are roughly around the same complexity seen in the transition matrices but far less than seen in the empirical signal. We quantified this by measuring the sparsity fraction by counting the number of transitions and dividing by the total number of possible transitions (Firing Rate has 0.55 transitions, Kuramoto has 0.52, and rs-fMRI has 0.86).

Figure 2.10 K-Means Algorithm Comparison

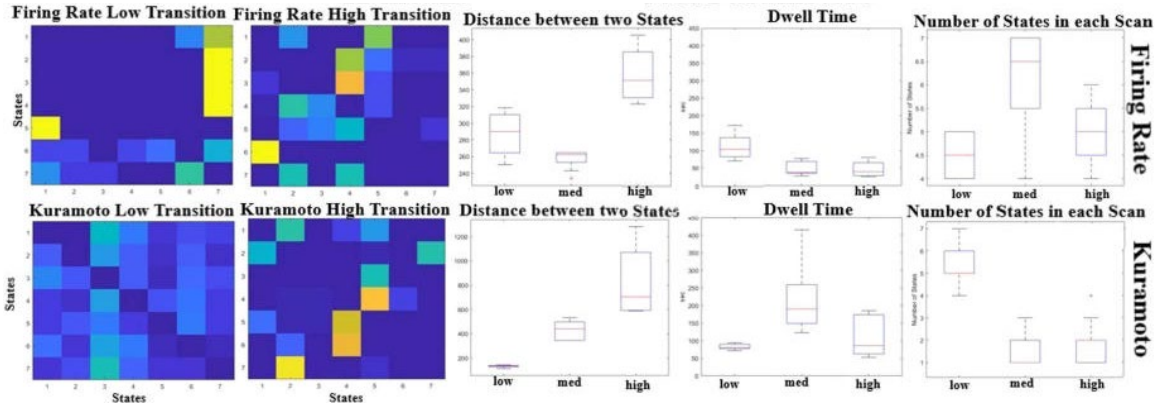


The top row shows the average number of states seen in an individual run (top left), the average dwell time in each state (top middle), and the mean distances between the centroids (top right).The transition matrices between the different k-means centroids are shown on the bottom row. The values reflect the number of raw occurrences divided by the total number of transitions giving the probability of transitioning from one state to another. Self-transitions are set to 0.

Varying the coupling strength also affected the state dynamics seen in the models. Figure 2.10 shows that the distance between the states increases as a function of coupling

strength, especially for the Kuramoto model, suggesting that stable dynamical states are moving apart from each other. The other parameters have a nonlinear relationship with the coupling parameters as the dynamics of the system changes. None of the models have the complexity seen in rs-fMRI in terms of number of distinct states and large distances between their centers, but the one that is the closest to the real signal is the Kuramoto model at medium coupling levels. Even though the BNMs produces some state transition properties, these BNMs clearly have much simpler dynamics as compared with the rs-fMRI and it cannot just be rectified by simply varying the parameters.

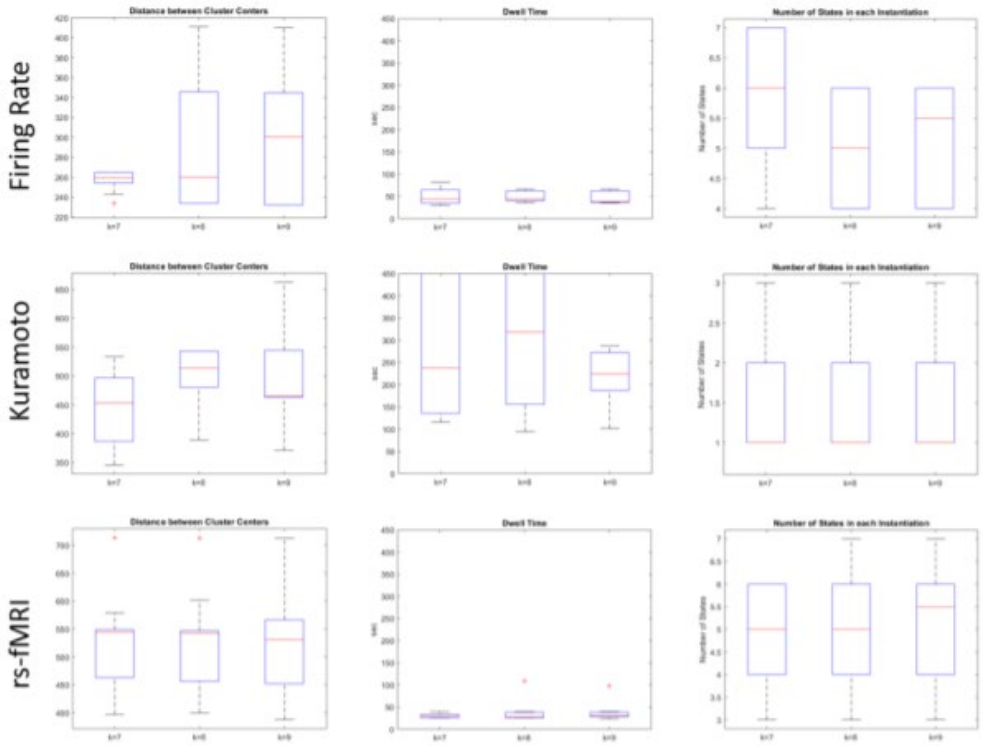
Figure 2.11 K-means Analysis Across Parameters



K-means analysis across parameters. The transition matrices for the Firing Rate (top) and the Kuramoto (bottom) are given at different coupling parameters. The distance between cluster centers (middle) is increasing as a function of coupling strength, suggesting that the states are diverging and are having more state-like dynamics for higher coupling strengths. The number of states is relatively the same even at higher

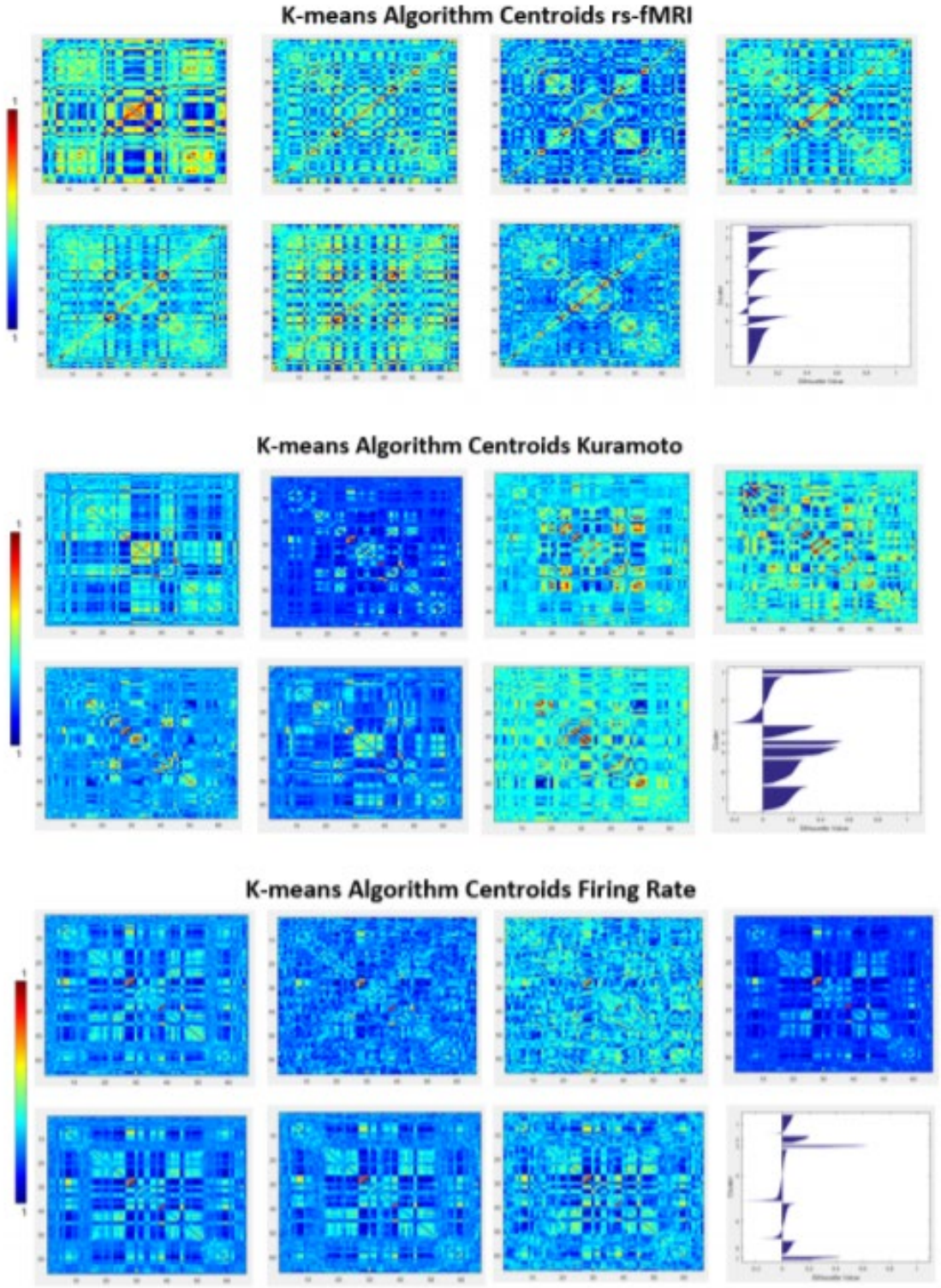
coupling strengths, suggesting that the models are limited to how much variety they can produce.

Figure 2.12 K-means Analysis with different number of Clusters



The effect of different K-means centers on the metrics used to evaluate the K-means model, such as average dwell time and the number of clusters. With sufficiently many clusters the metrics used is irrespective on the number of clusters used. We used 7 clusters in the main analysis, which is about how many are used in literature (5-9).

Figure 2.13 K-means Centroids



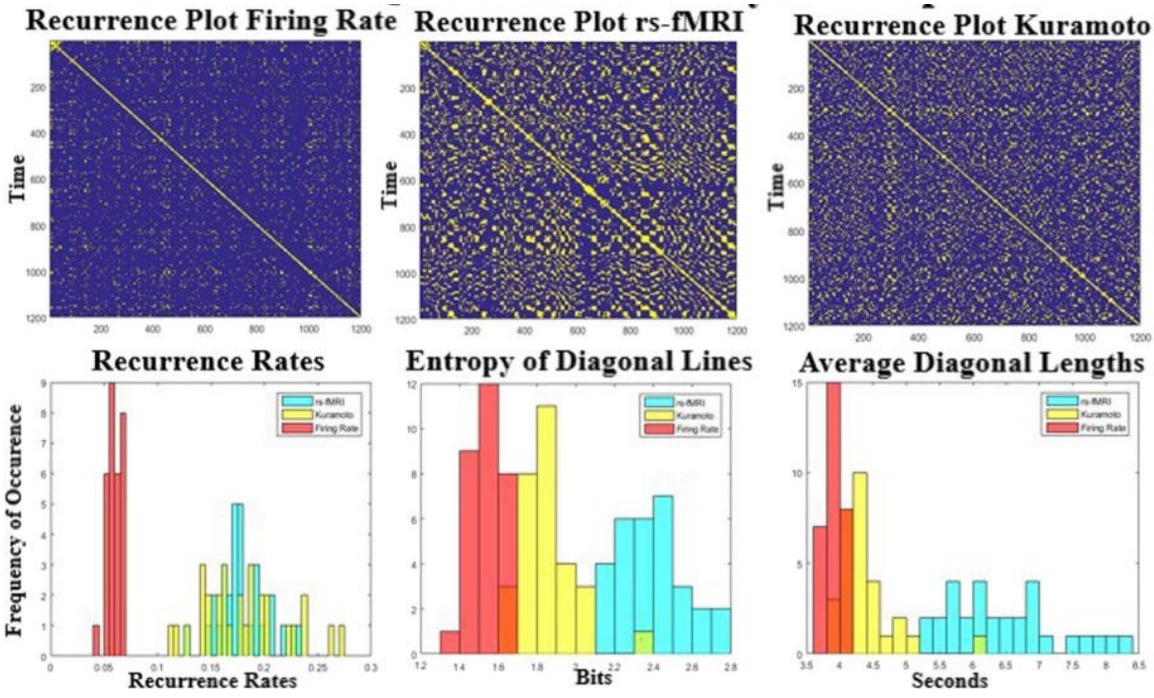
A comparison of all the cluster centroids identified by the K-means algorithm from each of the respective data sets. The spatial states don't vary that much in the two

simulated models as much as in the real data. The silhouette (bottom right of each data type) is a measure of how similar the values in each cluster are too each other.

2.3.5 Recurrence Quantification Analysis

Figure 2.14 shows the recurrence plots for the empirical and simulated BOLD signal. These plots are calculated by correlating the pattern of activity at each time point with the pattern from every other time point. Diagonal lines that are parallel to the main diagonal represent repeating transitions that are seen throughout the scan, whereas vertical or horizontal blocks represent dwell periods during the scan. A cursory inspection of the three recurrence plots (Figure 2.14, top row) shows that the two models have far less repeating structure than seen in rs-fMRI. This relation is quantified by the bottom three plots that show the recurrence rate (left), entropy of diagonal lines (middle), and average length of diagonal lines (right). The recurrence rate are much higher in the Kuramoto and empirical signal than in the Firing Rate model. However, the entropy and length of the lines (related to how different the states are and how long they linger) clearly separate the three datasets (bottom and right). Entropy and line length are highest in the real data and lowest in the Firing Rate simulation, with the Kuramoto simulation residing in between. The low entropy values for the Firing Rate data signify that the model does not have as many repeated trajectories as compared with the other modalities.

Figure 2.14 Recurrence Quantification Analysis Comparison



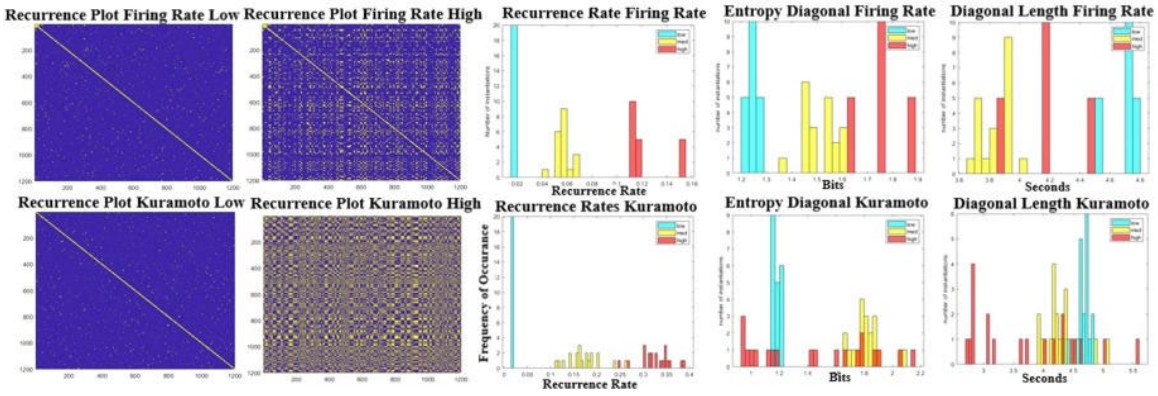
Comparison of the recurrent quantification analysis (RQA) between the simulated and real datasets. The top row shows three scans of recurrent plots thresholded at 0.3 for the three datasets. The bottom row shows the distribution of three different RQA techniques over all scans. The bottom left shows the recurrence rate, which is much higher for the Kuramoto and rs-fMRI model than for the Firing Rate simulation. The recurrence rate is a measure of repeated states seen in the dynamics of the rs-fMRI signal. The middle and the right plots quantify how much similar trajectories occur during the scan. The measured rs-fMRI signal shows much more variance between different trajectories and is of much longer duration.

Changing the global parameters has some very linear effects, as can be seen in Figure 2.15. At low coupling parameters the recurrence plot shows almost no structure, and at high coupling more and more structure emerges. The recurrence rate that is most related to the number of events seen in recurrence plots is almost linear (Figure 2.15, middle column)

for both models. At higher levels of coupling there is a larger spread in the models (entropy rate, Figure 2.15, right from middle column), as they are more a function of initial condition than at low levels of coupling. The entropy levels and the average diagonal length are much higher in rest than all of the models, suggesting longer, slower repeated trajectories in the real signal. Overall the technique is able to separate the empirical data and the models

Figure 2.15 Recurrence Quantification Analysis Across Parameters

and the more complex Kuramoto model, and at least one of the measures varies linearly with the parameter selection.



Recurrence quantification analysis across different parameters. Left are the recurrence plots at high and low global coupling for Kuramoto (bottom row) and Firing Rate (top row). The recurrence rate that quantifies the number of events increases monotonically (center column) as a function of coupling strength. The entropy and the lengths of the diagonal lines have a nonlinear relationship. Neither of the right two metrics are as high, as seen in the resting dataset (Figure 2.14).

2.4 Discussion

2.4.1 Analysis of Average Functional Connectivity and Power Spectra

From previous studies using multiple models and parameterizations, it appears that certain properties of the simulated signal are most reliant upon the underlying structural connectivity rather than the model of activity used at each node (Bullmore & Sporns, 2009;

Stam et al., 2016). In our study, these properties should be similar across models (which share identical structural connectivity) and in the real data. Average functional connectivity analysis is one of these properties. The structural connectivity matrices derived from diffusion tensor imaging and the respective functional connectivity estimates derived from resting-state fMRI have a correlation value of 0.45 as measured through our methodology, which is similar to what has been described before (Bullmore & Sporns, 2009). In fact, all three dynamical systems produce signals where functional connectivity is highly correlated with the structural input, leading to suggest that average functional connectivity is closely related to the underlying connectome. There are known differences between SC and FC; for example, an edge in FC between two nodes can be the result of a third process that drives the two structurally disconnected regions. However, most of the edges between the ROIs can be described as a function of how many white matter tracks run between them (Stam et al., 2016). Moreover, the frequency spectrum and the characteristic (1/f)^n distribution are similar for both BNMs and the empirical rs-fMRI, suggesting that the spectrum is a property of the underlying structure of the network. The Firing Rate model reproduces the average FC better than the Kuramoto model, which might be due to overfitting since it has fewer parameters to optimize compared with the Kuramoto model. This matches well with previous reported literature where the Firing Rate model has produced FC matrices that have a large correlation (corr. = 0.8) by tweaking the input SC to maximize similarity to the output FC (Senden et al., 2017).

2.4.2 Coactivation Patterns/ Point Process

The coactivation analysis also showed a shared feature across the empirical and simulated data. For all three datasets, coactivation patterns were strongly correlated with

average functional connectivity. This is likely because functional connectivity is driven by processes that activate certain subnetworks of the SC together (Smith et al., 2009). This further strengthens the notion that SC and average FC are closely related. The global coupling parameter affects the coactivation rates in a similar way as for average FC, where low levels of coupling result in a very uncoordinated system, and high levels result in global networks that are highly active. Since the model parameters were chosen to fit average FC best, and it is easier to fit the Firing Rate model, it once again reproduces measures that are more faithful to empirical rs-fMRI compared with the Kuramoto model, probably for the same reason that it can better match the average FC of the empirical data. Regardless of the coupling parameters used, the coactivation patterns are similar to the average FC for both models.

2.4.3 Quasiperiodic Patterns analysis

The successful detection of quasiperiodic patterns in both BNMs indicates that these network models capture at least some of the dynamical features of the brain's activity. Unlike the previous sections of spatial metrics where the naïve Firing Rate model performed better than the Kuramoto model, the QPP templates for the two models are indistinguishable. On the other hand, there are substantial differences compared with the real QPP. The real QPP is more complex, with gradual switching at different time lags in different areas. The real QPP template is also longer in length than the ones from the BNMs, as the simulated model starts repeating itself before the end of the template. The length of the spatiotemporal pattern is a function of the coupling parameter, especially in the Kuramoto model, where increased coupling leads to faster repetitions of the patterns. The overall spatial shape of the pattern (i.e., the areas involved in activation and

deactivation) are quite similar across models and parameterizations, though none are as complex as the patterns in empirical data. We believe the incorporation of aspects of neural field models and Connectome Harmonics (Atasoy, Donnelly, & Pearson, 2016; Sanz-Leon et al., 2015) into the existing BNMs may result in a more accurate reproduction of the spatial propagation because it would take into account surface propagation was well as network propagation. It could also be possible that the difference in QPP templates results from the unidirectionality of certain white matter connections or other properties that cannot be captured using standard tractography.

2.4.4 K-means Analysis

Some dynamic properties appear to arise more from the complex interactions linked to the unique temporal description of activity in each ROI than from the underlying structural connectivity. These properties are likely to be different for each BNM, and from previous literature, the Kuramoto model has outperformed the Firing Rate model (Cabral et al., 2017). The k-means algorithm on the windowed FC matrices revealed a complex network of states in the rs-fMRI data that each demonstrated distinct spatial patterns of connectivity between ROIs along with a complex web of transitions between them. In the Firing Rate model there are a similar number of states compared with rs-fMRI, but the distances between the states are much smaller compared with rs-fMRI, suggesting that it is more appropriate to consider the Firing Rate states as representing a single state artificially divided into multiple components. The state description in the Firing Rate model echoes previous findings because it is known that stable attractor states cannot be produced with only a linear set of differential equations (Cabral et al., 2017). The Kuramoto model at very low levels of coupling has similarly many states that are close together. But at

higher values the Kuramoto model, for some initial conditions, can produce transitions between states that are as spatially distinct as the rs-fMRI but limited to fewer states and a simpler transition matrix than the empirical signal. This suggests that BNMs can reproduce at least some of the dynamic states observed in rs-fMRI, although current models do not recapitulate the rich variety observed in empirical data. The Kuramoto model under certain parameters does better than the Firing Rate model and can produce complex state-like behaviour.

2.4.5 Recurrence Analysis

Recurrence analysis quantifies elements in the temporal structure of the data similar to clustering analysis on short FC matrices. Therefore, we predicted it to depend more on the dynamic description of the data rather than the shared spatial connectivity input. The Firing Rate model again exhibited the least complexity, the rs-fMRI data exhibited the most, and the Kuramoto model fell in between the two for the normal parameters. The empirical data have repeated trajectories that occur more often and are longer than either observed in the simulated BNMs. Out of the dynamic analysis metrics that were examined, the RQA metrics separated the three datasets the most effectively, whereas the average functional connectivity analysis exhibited the fewest differences. Moreover, the recurrence rate that quantifies the number of repeated temporal events seems to linearly depend on the coupling parameters for a certain range. This relationship is similar to average FC, which at low levels of coupling shows no structure and at higher levels of coupling shows increased network structure (Figure 2.6). Average FC and recurrence analysis both use correlation, except that one uses the space across rows that spans the ROIs, whereas the other uses the space of the columns that represent single time points in the BOLD data.

Average FC, which examines coordination between ROIs, reveals a static network related to the input SC. Recurrence analysis that examines the time domain reveals properties that seem to be most unique to the formulation of each BNM. Recurrence analysis also is quick to compute and therefore could be a good addition to average FC as a metric for model selection, which is a very computationally intensive process. Together they can ensure that the model has roughly similar network component structure compared with rs-fMRI based on average FC, and similar temporal structure recurrence rates.

2.4.6 Overall Discussion

The goal of our exploratory study was to find better dynamic metrics to compare empirical rs-fMRI and the brain network models. We have chosen two different BNMs at three different parameterizations to provide an axis of contrast between the simpler Firing Rate model and the more complex Kuramoto model, which has been shown to reproduce more complex dynamic trajectories (Cabral et al., 2017). The dynamic analysis techniques can be ordered in how much they analyze the structure in the spatiotemporal BOLD signal as a function of spatial coordination between regions or repeated temporal trajectories. The ordering in Table 2.4 is not strict, but loosely goes from techniques that observe spatial patterns to those that observe temporal patterns. These two components seem to correspond to the two main components in the formulation of BNM: the structural network that provides input from connected ROIs and a description of the evolution of the state variables.

The Firing Rate model outperforms the Kuramoto model on metrics that are more closely linked with spatial patterns, whereas the Kuramoto performs better on metrics that are linked with the temporal structure observed in rs-fMRI. We believe that the

performance on spatial metrics, such as average FC and point process, is due to the Firing Rate model being easier to fit to the rs-fMRI because of its fewer parameters. Moreover, since average FC and the SC are similar, it is probably an easier task to match the FC output that is very closely related to the SC input. The temporal metrics reveal that the Kuramoto model has much richer dynamics than the Firing Rate model and is closer in reproducing features seen in rs-fMRI. Moreover, these differences are more likely due to the differences in the differential equation formulation of the BNM since it defines the network evolution.

Table 2.4 Comparison between models across different analysis Techniques
suggesting that it might be an invariant property of all BNMs.

Analysis technique	Firing rate	Kuramoto	Metric
Static FC	Better	Worse	Correlation between average FC of rs-fMRI and models (Figure 2)
Point process	Better	Worse	Correlation between point process matrices of rs-fMRI and models (Figure 3)
QPP	Tied	Tied	Correlation between the shape of extracted spatiotemporal signal between rs-fMRI and models and their rate of occurrence (Figure 4)
K-means	Worse	Better	Number of distinct states and diversity of state transitions (Figure 6)
RQA	Worse	Better	Average number of repeated temporal transitions and the entropy of those transitions (Figure 8)

2.4.7 Limitations

Our modeling approach makes many simplifying assumptions that do not capture the true complexity of the brain. In the construction of the structural connectome, we assumed that all connections were bidirectional. This is a limitation of using tractography to build the structural network, since tractography cannot distinguish unidirectional connections. Moreover, estimates of fiber density for connections between regions that

have very sharp angles or between regions that are spatially far apart are far lower than the true connectivity between these regions (Bullmore & Sporns, 2009). In our generative models we also assumed a homogeneity in the response of ROIs, in both their neural description, as well as their transformation using the hemodynamic Balloon-Windkessel model. Moreover, we did not simulate subcortical structures that are known to play a crucial role in the operation of the central nervous system. All these factors might change the association between dynamic metrics and the simulated BNM signal.

We also examined only a single parameterization for only two BNMs. There are a variety of BNMs, some of which are likely to exhibit more complex dynamics than either the Kuramoto or the Firing Rate model (Sanz-Leon et al., 2015). Even different parameterizations of a single model can give rise to vastly different behavior (Hansen et al., 2015). We chose to focus on the Kuramoto and Firing Rate models because of their relative simplicity, their thorough characterization, and the expectation that they would have dissimilar dynamic properties.

There are also numerous dynamic analysis methods available for rs-fMRI (Keilholz et al., 2017). We chose to focus on a few of the most common ones, but future work should certainly examine the use of other types of analysis to produce even more sensitive metrics. Moreover, our study does not look at methods to test these metrics and use the established correlation as distance function. We have also not explored the entire space of parameterization, so it is possible that these models can produce more realistic signals; however, based on previous results establishing these as close to optimum, the results are probably a realistic representation of their capabilities.

2.5 Conclusion

We believe that the more-temporal metrics, namely RQA or k-means, would be the most appropriate in evaluating BNMs in the future because they show that the more complicated Kuramoto model recapitulates rs-fMRI dynamics better than the Firing Rate. The k-means approach is a stronger criterion to evaluate on, because the cluster centers as well as the state transitions between the model and the empirical signal would have to match in order to reproduce resting-state dynamics. However, the RQA approach is less computationally intensive and can used to quickly check the diversity in the temporal structure of the simulation and to assist the selection of parameters in the model. The QPP algorithm would also be an interesting method to test future models as it is strongly dependent to changes in the global coupling but seems to be less dependent on differences between Brain Network Models. Average FC and the point process do not reveal processes that are much more complex than the SC input.

From the dynamic analysis perspective, the most distinguishable metric in rs-fMRI seems to be predicting the temporal structure of the signal. Although this chapter establishes these dynamic metrics as more useful in comparing between models and empirical data, they don't allow for a direct comparison of the evolution of the model and the empirical data. We speculate that the most transient features are the most characteristic of rs-fMRI and the hardest to reproduce using non-linear equations and if there was a way to synchronize the models and the data, we could train on the difference directly.

CHAPTER 3.

MODELING RESTING STATE FMRI EVOLUTION USING NEURAL ORDINARY DIFFERENTIAL EQUATIONS

SUMMARY

Motivation: Brain Network Models (BNMs) represent a family of dynamical systems that simulate whole brain activity. The simulations have been compared to empirical measurements such as functional magnetic resonance imaging (fMRI) using metrics that are averaged over long periods of time and have shown some degree of similarity. However, for shorter periods of time the BNMs have not been compared with the empirical data because the initial conditions of the BNM with respect to the data are not known. To address this challenge, we use Deep Learning to solve for initial conditions based on empirical rs-fMRI data. From these initial conditions, we use an Ordinary Differential Equation (ODE) solver to integrate the BNM model and compare the resulting timeseries directly with either the empirical timeseries or from another BNM with a different dynamical system (Chen et al., 2018). The approach represents a scientific tool that allows us to test how well a specific dynamical system is representative of the observed signal over a short period of time.

Hypothesis: From the estimated initial conditions, the BNMs that are known to reproduce more complex non-linear properties of rs-fMRI, will evolve more closely to the empirical

signal as measured by r-squared over multiple future timepoints, than the simpler linear BNM models.

Approach: The approach is based on the Neural ODE paper (Chen et al., 2018) which uses a recurrent neural network (RNN) to estimate the initial condition for a given BNM and an ODE solver to generate subsequent estimates for time series prediction based on the BNM equations (Section 3.2.1). During training, we estimate for each given datapoint at every timestep the initial conditions. From the initial conditions, we integrate over the same time interval the data is collected for and generate our estimate for the next timestep. The RNN is then trained using the difference between the predicted value and the actual measured value. We first validate that our model predicts the correct initial conditions by validating on a constructed spiral dynamical system, where the initial conditions are known (section 3.2.4, section 3.3.1, section 3.3.2). To fit the neural data, we use the linear Firing Rate model and the nonlinear Sigmoidal Firing Rate model as our BNMs (section 3.2.3). For each specific model, we also vary the parameters of the system which change the dynamics of the system (section 3.2.3). During testing, we use these BNMs on their own without the RNNs to generate multiple timepoints from a single set of initial conditions. The timepoints that are further out in the future are less dependent on the RNN's initial conditions and more of a function of the candidate BNM. The trajectory of the BNM is then compared with the measured timeseries. For our empirical dataset, we used rs-fMRI scans from 500 subjects that were registered to standard space and collected at 0.72 sec intervals (section 3.2.5.2). We utilize three null models to test against the Neural ODE approaches (section 3.2.7). The first one is created by setting the BNM derivative to zero, and the next predicted timestep is the output of the RNN. The second one is a Neural ODE using a purely

exponential function to fit to the fMRI data. The third model is a non-machine learning Autoregressive model that is traditionally used as time series predictions in fMRI data (Smith et al., 2009).

Metrics and Evaluation: We use the accuracy at the future timesteps to evaluate the differences between BNM models. In order to test our hypothesis that this technique is able to differentiate between dynamical systems that are either closer to or further away from empirical measurements, we test on three different scenarios. First, we test how the differences on parameterizations for a single BNM model changes the short-term prediction, by varying the weight of the global coupling term (section 3.4.2). For each parameterization, we train separate RNN networks to solve for initial conditions for the respective models. From previous literature, we can set this global coupling term to lead to dynamics that is closer to rs-fMRI dynamics or to a different set of dynamics that are thought to be further from rs-fMRI dynamics. In the second test, we test differences in short term predictions due to changes in the structural matrix of the BNM by introducing noise at varying levels into the structural matrix that is estimated via imaging (section 3.4.3). We anticipate that the model using the measured structural matrix would have the highest r-squared compared to those models with a noisier version of the structural matrix. For the final test, we test the differences in prediction between a Firing Rate Model and adding a sigmoid function to a Firing Rate model with different parameterizations. We expect the introduction of a sigmoid to increase our accuracy as it limits the derivative to a finite value and is therefore more biophysically plausible (section 3.4.4).

Conclusion: The Neural ODE framework allowed us to differentiate how well BNMs with more and less accurate structural matrices fit the rs-fMRI data. It was also successful in

showing that the non-linearity increased the accuracy of predicting short term rs-fMRI signal. The parameterization problem did not yield a simple solution, due to the unknown quantity of noise and the global coupling, and the relationship on which model performed the best seemed to be a function of the noise added into the system. Accuracy moreover decreased with the addition of noise which made it difficult to evaluate the parameters using only the r-squared metric. However, using the structural connectivity difference as a ground truth, we believe that our method can be used in order to solve for both of these parameters. In short Neural ODE framework shows promise in differentiating between BNMs that best recapitulate the measured rs-fMRI data.

3.1 Introduction

Resting state functional magnetic resonance imaging (rs-fMRI) allows researchers to estimate the macroscale activity of the cortex over the slow frequency range (0.01- 0.1 Hz) (Biswas et al., 1995). At these frequencies, the brain activity is thought to originate from modular neural populations interacting with each other through a macroscale structural network (Deco et al., 2008). During resting state, the human participant is not presented any sensory stimulus or asked to do a particular task, so the activity is thought to be originating mostly from intrinsic cortical loops present in the structural network (Honey et al., 2007). Supporting this hypothesis, generative models based on the structural network, known as Brain Network Models (BNMs), constructed from a given cortical parcellation and estimating the number of the white matter fibers between two regions of interest, are able to recapitulate aspects of the temporal dynamics of neural populations measured by rs-fMRI (Cabral, Hugues, Sporns, & Deco, 2011, Sanz Leon et al., 2015). Exploring these network models, would allow us to gain insight into how activity in a modular region, i.e., precuneus, interacts with the rest of the cortex. The ability of BNMs to translate changes in local activity to changes in network activity, has also made the models a target for task studies, where they are being used to model the inputs to the cortex and mimic observed fMRI responses (Ritter et al., 2019). Moreover, BNMs also provide a natural framework for clinical interventions, since they use latent variables that represent properties of a neural population that can be measured and manipulated such as the mean firing rate of a neural population with the means of a deep brain stimulator (Saenger et al., 2017). Therefore, current research has been focusing on improving our understanding of these network

models with respect to reproducing features of rs-fMRI in order to analyze, predict and manipulate whole brain signals.

One of the challenges in building a generative model for rs-fMRI, is that it has been difficult to compare the generative model directly with the data (Cabral et al., 2017). Since rs-fMRI is defined without an explicit stimulus, there is no clear starting point to start generating the signal. Therefore, researchers have been using metrics that quantify properties in rs-fMRI over a relatively long period of time, to compare the data and the simulations of the models (Cabral et al., 2011). This is somewhat unusual, since the generative models provide an exact description of how the signal should evolve and for a given observation should be able to predict a trajectory. The lack of a direct comparison between the predicted trajectory and the actual trajectory has made the field rely on time averaged metrics such as functional connectivity (FC), which is computed as the correlation between the timeseries representing the activity of regions over a long period of time. However, the FC metric is not very effective in differentiating between BNMs that are simulated using different equations and parameterizations and result in vastly different dynamics (Cabral et al., 2017, Kashyap et al., 2019). In chapter 3 we showed that dynamic metrics that quantify fast transient processes in rs-fMRI are better at differentiating between models. Therefore, in this work we hypothesize that if we can compare the simulated short-term trajectories directly with the observed timeseries, we can differentiate between models more efficiently.

This chapter describes using a novel method developed in the Machine Learning community that can be used to compare the short-term trajectory of the BNMs with the measured timeseries. We use a technique, known as Neural Ordinary Differential

Equations (ODE), that uses a recurrent neural network that keeps track of information from previous timepoints, in order to predict the initial conditions of a given BNM from an observed datapoint (Chen et al., 2018). From the initial conditions we can then synthesize the trajectory based on the equations and compare it to future datapoints. The RNN system is trained such that the integral, implemented via an ODE solver, of the predicted initial conditions with respect to a given dynamical system matches the next measured timestep. Chen et al., demonstrated that the ODE solver can be differentiated and be used in conjunction with current RNN, in order to train the output of the system to match the initial conditions for a known ODE system. They show that minimizing the loss function distance of the next predicted datapoint and the measured datapoint, allows the RNN to converge on the initial conditions of the current timepoint. Before testing the system on rs-fMRI data, we validate that our implementation is able to solve for the correct initial conditions on a constructed spiral dataset.

Our goal with this method is to use it as a tool to differentiate between candidate BNMs on how well they can recapitulate rs-fMRI. We hypothesize the BNMs that are known to reproduce more complex non-linear properties of rs-fMRI, will evolve more closely to the empirical signal as measured by r-squared over multiple future timepoints, than the simpler linear BNM models. We define this future trajectory by using a trained model to estimate the initial conditions and then generate a set of future discrete timepoints via integrating the BNM equations at the same sampling rate as the measured fMRI dataset. Comparing the resulting generated trajectory with the observations will then allow us to see which model fits the data. We validate that our method can perform this notion of system identification, on our constructed spiral data. We show that the dynamical system

that was used to generate the noisy data, fits closer to the observations, than if we try to fit a perturbed version of the original dynamical system to the observed data.

To quantify how well the approach does on differentiating between BNM with respect to rs-fMRI data, we designed experiments where we changed aspects of the BNM that would result in known differences to the simulated timeseries. The structural matrix was perturbed in a controlled manner starting with the original structural matrix, and adding noise to generate increasingly noisier versions of the original. We expect the models simulated with the original structural matrix to outperform the models with noisy structural matrices. Moreover, we introduced a simple nonlinearity in our base BNM to see whether the method can differentiate between simpler and more complex models. For that effect we chose to compare the linear Firing Rate Model and with a non-linear Sigmoidal Firing Rate Model, where we expected the non-linear to outperform the linear model (Cabral et al., 2011). The Sigmoidal Firing Rate Model is used here as it is a simple non-linear extension with relatively few parameters to bias and is used to differentiate it from the linear Firing Rate Model and not intended as a gold standard biophysical model. The models were also parameterized at different levels, where one set of parameterizations reduce the contribution from the network to zero, resulting in an exponential decay system. Another null model was constructed by setting the differential equation to zero, and directly using the output of the RNN to predict the next timestep. Furthermore, we fit a linear autoregressive model, and use these two basic functions as a null model to compare against the performance of Neural ODE BNMs.

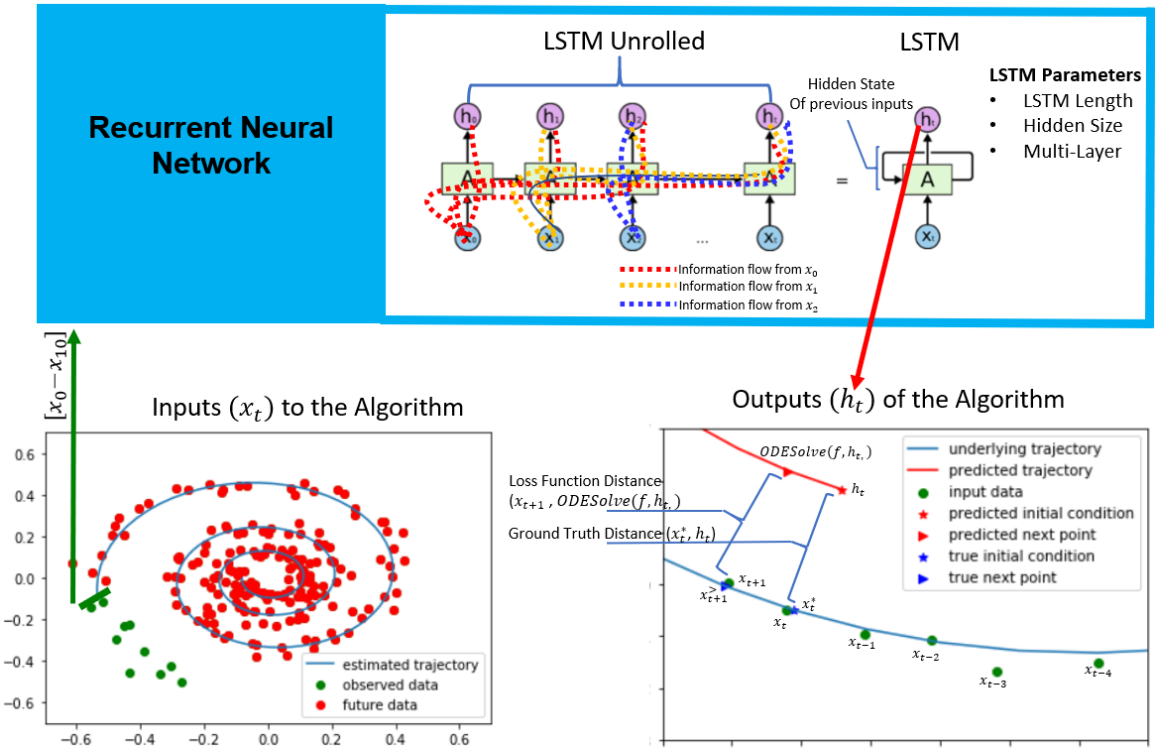
This novel approach, allows us to directly compare the predicted trajectories of a given BNM with the observed timeseries. If validated successfully in its ability to

distinguish BNMs with known variations, it would potentially be a new useful tool in order to help find better generative models for rs-fMRI.

3.2 Methods

3.2.1 Neural Ordinary Differential Equations

Figure 3.1 Schematic for Neural ODE Algorithm



Schematic for the Neural ODE algorithm. An example spiral is shown on bottom left. Sequences of the data are fed into the LSTM which updates its hidden state with more input values. The LSTM sees one data point at a time and updates its hidden state as well as outputs its prediction for where it believes the initial condition to be. Information from previous observations is kept track using the hidden state and is passed through feedback to future timesteps. An illustration of this concept is shown using the LSTM unrolled diagram. The output of the RNN represents the initial condition of the dynamical system at that timestep. For the spiral dataset, the true ground truth initial condition x_t^* is known and is illustrated in bottom right. The ground truth distance is used to determine if we have trained our RNN network successfully to predict initial conditions. The next timepoint is predicted by

integrating the ODE system based on the given dynamical system. The loss function is defined as the difference between the next predicted timepoint and the next observed time point, x_{t+1}. We predict that minimizing the loss function distance should minimize the ground truth distance.

An overview of the algorithm is shown in Figure 3.1, based on using Neural ODE to fit a spiral based on noisy observations. The task of the machine learning network is to predict an initial condition on the underlying trajectory shown in blue starting from the sequence of observed measurements shown in green. We chose an RNN implementation known as Long Short Term Memory (LSTM) in order to implement this architecture, as it allows us to keep the information of past data observations [x_0 to x_(t-1)] in its hidden state p_(t-1) (Graves & Schmidhuber,2009). Thus, when the timeseries is fed into the system one by one, the current information is incorporated into the hidden state and is passed forward as shown in the LSTM unrolled version, in order to aid in the prediction of future observations. As the LSTM observes more datapoints, its predictions become more accurate up to a certain limit, after which newer data doesn't add anymore information to what is already contained in the hidden state (Graves & Schmidhuber, 2009). Thus, after ignoring the effects of initialization, the LSTM implementation of RNN has become a popular tool in modeling timeseries as it is able to infer the relationship between past inputs in order to predict the next timepoint. For our particular task, we train the LSTM to predict the initial conditions of a given dynamical system based on the observed timeseries. Since the initial conditions are not known and thus an effective gradient cannot be computed based on initial conditions alone, we assume that the next observation is the integral of the predicted initial conditions and the given dynamical system with some noise added to it (Chen et al., 2018). We thus calculate our loss function with respect to the next

measurement in order for the output of the LSTM to converge onto the correct initial conditions for the given timepoint.

We outline the steps in the sequence they are executed in order to train our system.

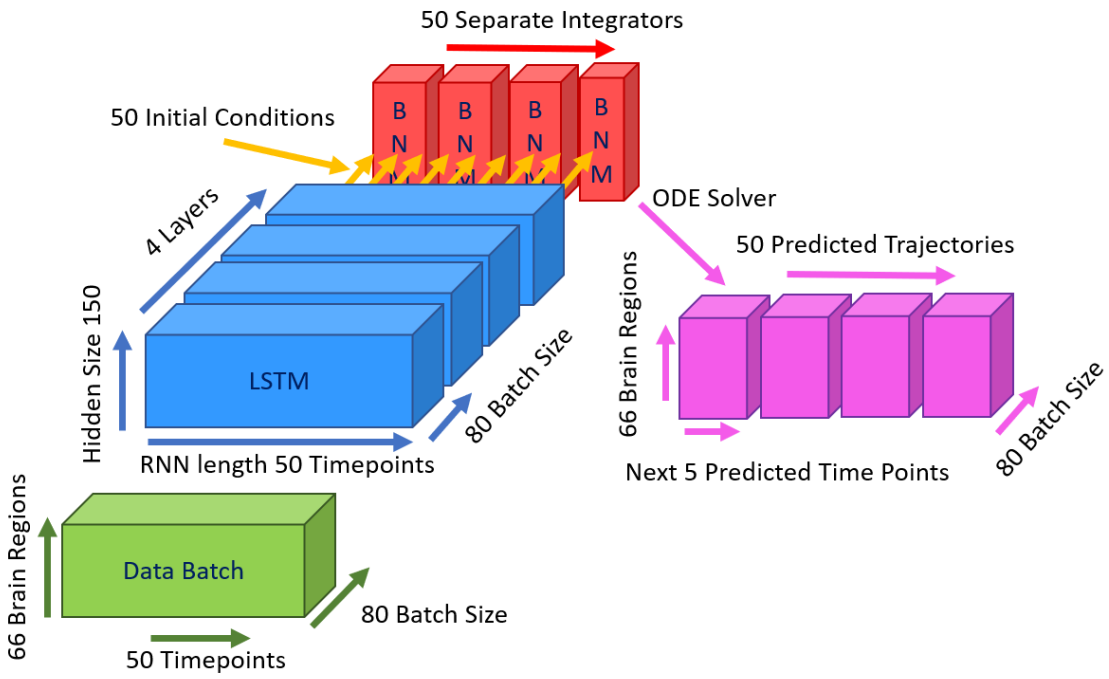
1. The input data is shown in the bottom left corner Figure 3.1, where it consists of sequence of timeseries x_0, x_1,x_2… to x_t. They are fed into the LSTM network one at a time. The Tensorflow LSTM implementation of the RNN lets us to remember up to x_(t-N) values where N is the length of the LSTM. N is set sufficiently large such that the performance of the LSTMs converges to its optimal value in M timesteps and N>M.
2. For the current input x_t and the LSTM with its hidden state p_(t-1), the LSTM estimates the initial conditions as a gaussian distribution with a mean and standard (μ_t,σ_t). We then sample from the distribution in order to estimate for the initial condition h_t. This is not explicitly shown in Figure 3.1, but is done in order to avoid overfitting.
3. Calculate the loss function for our estimate at point t (Figure 3.1 bottom right): We use a 4th order Runge-Kutta to compute ODESolve(f,h_t), where ODEsolve represents the differential equation solver to integrate over the interval t to t+1. We assume that this point is close to the next measured datapoint x_(t+1). The Euclidean Distance between these points is used as our loss function. A graphic of this distance is shown in Figure 3.1 bottom right.
4. The gradient is then calculated with the average of our loss function across all timepoints as well as across all batches and is back-propagated through the Neural ODE using Tensorflow, in order to train the network.

The algorithm trains until the loss function distance approaches zero. The Neural ODE algorithm assumes that the ground truth distance between the predicted initial conditions and the true initial conditions also converges to zero when the loss function distance is minimized. The algorithm is initialized with a null hidden state, and slowly updates its hidden state as it sees more observations over time.

3.2.2 Tensorflow Implementation

The schematic above is implemented as the architecture shown in the Figure 3.2 below, which utilizes parallelization in order to train efficiently with large amounts of data. The network trains using multiple subjects at the same time (batching of 80 subjects) and multiple time steps at the same time (50 consecutive time steps). However, in its simplest essence, we can consider a single time step and a single subject. The input would be a vector from the activity of all brain regions (66 ROIs at one time point) and hidden state vector calculated from the previous time step by the LSTM (4 layers of hidden layer size of the LSTM 150) (for now we will assume we have a correctly initialized hidden state and will explain later how its initialized). The hidden state represents the feedback arm of the LSTM. The final output of the LSTM is then transformed from the size of the LSTM (150) using a feed forward Neural Net to the size of the initial conditions before it is fed to the BNM (66). We represent the activity at each region by its mean firing rate of the neural population at that timestep (66 for the different brain regions). The model is then integrated for a time interval and produces an output representing the activity of the measured signal at the next time signal.

Figure 3.2 Neural ODE Architecture



Neural ODE architecture as represented by tensors with their appropriate sizes. The Data Batch consists of 50 consecutive timepoints from 66 Brain Regions of 80 subjects (batch size). The batch is fed into a 4-layer LSTM network, with a hidden size of 150. The initial conditions of the LSTM are set first to zero, and then to the output of the previous batch that contained the last 50 timepoints. The output of the LSTM is transformed via a feedforward layer to represent the mean and standard deviation of the initial conditions. An initial condition for each timepoint is then sampled and fed into the BNM in parallel and integrated by 50 separate integrators for the next 5 consecutive timepoints. The prediction at the next timestep is used to train the system, while the other 4 timepoints are evaluated during testing.

The algorithm is trained in parallel using a 3-dimensional tensor (in my code 66 ROI regions, 80 batches, 50 timepoints). The batches are treated independently across the entire pipeline. The LSTM is first initialized with a hidden state of all zeros and computes the data sequentially starting with the first timepoint of the 50 timepoints. The hidden state of the first input timepoint is then passed to the next timepoint. It therefore produces 50 outputs the size of (66, 80) after the forward neural network and represents the initial conditions of the BNM at each timepoint. The BNM then integrates in parallel for a given time interval from each of the 50 initial conditions to give 50 predictions of the signal at

the next time step (49 are used as the last one does not have a corresponding next timestep) The mean square error between the final tensor (66, 80, 49) and input tensor without the first timestep (66, 80, 49) is used to calculate the gradient to update the weights. The algorithm also outputs the hidden state of the LSTM which is fed as the input to the next sequential batch. For example, HCP has 1200 timepoints and is cut up to segments of length 50, and the first 50 are processed in the first batch and then the next 50, in order for the hidden state to align. The accuracy on the first batch is usually much lower than the subsequent batches and is not used in calculating the results.

Training represents when batches are used to calculate the loss function and the weights are adjusted according to Tensorflow gradient descent. During testing, we do not calculate the loss function. For inference or testing, we first generate the hidden state with a batch of data the same size of the testing data in the same dimensions as described in the first paragraph describing the simplest case. After the hidden state is initialized, we run the system as single timepoint at a time using the previous output and hidden state as our next input. This process has no external information from the fMRI data except for the first initial state of the LSTM and the first initial measurement. We use this process to generate our estimates from the first initial timepoint X_0 and hidden state, the predictions of $X_1, X_2, \ldots X_k$.

3.2.3 Brain Network Model

3.2.3.1 Firing Rate Model

We start by using the simplest BNM in order to fit the rs-fMRI data. The Firing Rate Model represents the activity of a brain region as the mean firing rate. The change in

firing rate of a region is a weighted sum of all its neighbors' activity (Eq 1). The weight matrix is a normalized representation of the structural network, and represents the number of fibers between two brain regions. The Firing rate model has two parameters k, which controls the influence between a decay to the mean firing value, and the level of noise σ that simulates random activations of brain regions due to unknown neuronal activity (Cabral et al., 2012). It is fairly easy to parameterize and is used because it is the simplest model that utilizes the structural network and is usually parameterized with k=0.9 and σ=0.3. While training the networks, the noise is reduced to zero, but after the initial conditions are generated by the RNN, we vary the level of σ when integrating Eq 1 to generate future timeseries.

$$\dot{x}_l = -x_l + k \sum_{j \in Neighbors\ of\ l} w_{lj} * x_j + n(0, \sigma)\ Eq.1$$

3.2.3.2 Non-linear Model

To provide a non-linear model to contrast the Firing Rate model, we add in a sigmoid function on the network term, and refer this to as the sigmoidal Firing Rate Model:

$$\dot{x}_l = -x_l + \alpha S(k \sum_{j \in Neighbors\ of\ l} w_{lj} * x_j) + n(0, \sigma)\ Eq.2$$

$$S(y) = \frac{1}{1 + e^{-y}}\ Eq.3$$

The purpose of the sigmoid is to limit the magnitude of the network component and it adds biophysical plausibility by limiting the max firing rate. Similar justifications are used to

extend the Wilson Cowan model, and indeed the sigmoidal Firing Rate Model is just the excitatory component of the Wilson Cowan Model (Sanz Leon et al., 2015). We test a model that is identical to the Firing Rate Model with the same parameterization, vs the same model containing the sigmoid. In this manner, we evaluate the effectiveness of a adding a known non-linear function to reproduce rs-fMRI dynamics.

3.2.3.3 Parameter and Structural Perturbations

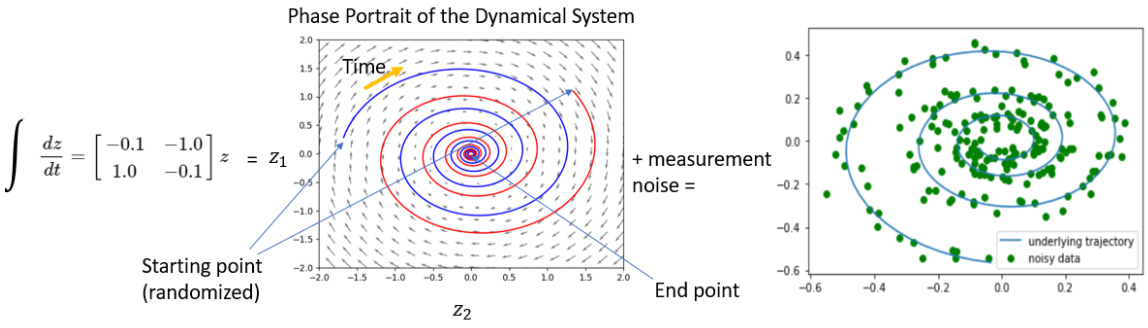
To generate the structural perturbations, we add noise to the structural matrix W_lj by swapping edges randomly, but keeping the graph symmetric. In this manner we produce random perturbations in a controlled manner starting with the original structural matrix and randomizing different number of the edges. Each of these graphs would result in different dynamics, but the trajectories from the model containing the original structural connectivity should be the closest to the measured rs-fMRI data.

For our parameter variation, the global coupling parameter k in Eq 1, is varied from 0.9 to 0. At values higher than k=1, the matrix has positive eigenvalues making the dynamics unstable, and at k=0 the system reduces to an exponential decay system with no network interaction at all. In the original Firing Rate Model paper, k=0.9 was thought to be most representative of brain dynamics (Cabral et al., 2012). We expect the parameterization based on previous literature to perform the best in matching measured rs-fMRI.

3.2.4 Simulated Spiral Data

In order to validate our approach, we tested our algorithm/architecture that we use on fMRI data, on a toy spiral data set. This spiral data has been used in the original Neural ODE paper (Chen et al., 2018), as well as in subsequent papers in order to test the validity of solving for the underlying dynamical system from noisy data. Our approach is to establish that our algorithm can reproduce the performance of the Neural ODE algorithm (Chen et al., 2018) on the spiral data and this therefore justifies its use on fMRI data. In Figure 3.3, we show how we generate our spiral dataset, namely by integrating a set of coupled differential equations with two state variables (equation shown in the left panel). The phase portrait of the dynamical system is shown, where all the trajectories from any initial condition spiral inwards towards the origin (Figure 3.3 middle). Time is not shown in the graph but is implied, where the first timepoint is on the edge of the spiral and the last one is the one closest to the origin. The derivative is large on the outside of the spiral and then decreases as it approaches the origin. Gaussian noise is added to the integrated trajectory to simulate measurement noise set at sigma = 0.03 (Figure 3.3 right). The goal of the Neural ODE algorithm is to estimate the underlying trajectory from the noisy observations that generated the data. For our experiments, we use the trajectories shown in the middle panel of Figure 3.3 as the ground truth and take the Euclidean distance between this trajectory and the predicted initial conditions as a measure on how well our algorithm performs.

Figure 3.3 Generation of Spiral Dummy Data



The equation on the far left is integrated and produces the dynamical system shown in the phase portrait (middle). From two random initial conditions the red and blue trajectories spiral towards the origin, getting slower and slower as they approach the origin. The arrows indicate the magnitude of the gradient. We generated 1000 spirals from different initial conditions and added noise to simulate a noisy measurement. This results in the plot on the far right where the green dots represent the data that is fed into the algorithm. The goal of the algorithm is to be able to estimate the underlying trajectory shown in blue (far right) from the noisy observations. This will serve as our ground truth to check our predictions against.

3.2.5 Experimental Data

3.2.5.1 Structural Network for Brain Network Model

To estimate the structural network we ran tractography on 5 HCP Diffusion Weighted Images using the freely available software Mrtrix (Kashyap & Keilholz, 2019; Van Essen et al., 2013). From the tractography we estimated the number of fibers that intersected two ROIs in the Desikan-Killiany atlas and normalized the power by dividing by the surface area of the receiving region (Cabral et al., 2011; Desikan et al., 2006). The matrix is finally normalized by dividing by the largest eigenvalue such that the graph Laplacian (k*SN-I) has only negative eigenvalues (Cabral et al., 2012). This normalizes the dynamics so that the feedback decays over time, and does not exponentially increase the signal over time. The value of k is a hyperparameter and modulates the strength of the

network and is varied as listed in the Section above. We started with 0.9 for all implementations which is in line with previous work (Cabral et al., 2012), but tested a range of values between 0 and 1.

3.2.5.2 fMRI Data

We acquired the training fMRI data from the 447 subjects from the Human Connectome Project (HCP) (Van Essen et al., 2013). We took the minimally processed scans (MSMAII) that were registered to standard space and in CIFTI format and denoised using 300 Independent Component Analysis that are provided from HCP. We transform from the surface-voxel time series to the ROI time series by averaging all voxels according to the parcellations established by the Desikan-Killiany atlas. This was done on an individual level since the surface parcellations are provided to by HCP and Freesurfer for each individual subject (aparc and aprac2009 files). The signal is then bandpass filtered from 0.0008 Hz to 0.125 Hz and then the global signal regressed using a general linear model with the mean timeseries of all cortical parcellations. The final signal is subsequently z-scored (Kashyap & Keilholz, 2019). For the task data, each dataset was processed separately (language, working memory, motor, social, emotional, gambling, relational) and then concatenated together. Each task dataset was rounded to the closest multiple of 50 and the autoencoder fed alternating segments of task and the rest data. We trained our algorithm using task data scans, because the algorithm was able to able to perform better on our short and long term metrics when trained with more varied data such as task. Moreover, it is thought that even during specified tasks, the resting state networks dominate most of the activity (Smith et al., 2009). However, during evaluation we have

only shown our results on predicting future resting state fMRI, and are planning to address task in our future work.

3.2.6 Metrics and Evaluation

The dynamical models are evaluated on how well they fit with the empirical observations from the given initial conditions. In order to quantify how well they fit, we calculate the r-squared and the mean squared error at each timestep between the vector representing the activity of the 66 brain regions of the predicted and the observed data. Since the loss function trains the one step error close to zero, this metric tends to be most similar across models. Therefore, in order to differentiate between the models, we need to look at timesteps past the first timestep to gauge how well the trajectory matches future measurements. For the spiral dataset, the initial conditions are already known and therefore we can compare the output of the RNN directly via Euclidean distance.

3.2.7 Null Models

In order to compare the effects of fitting to the Neural ODE, we utilize three different null models. The first and most naive is the autoregressive model which in essence predicts that the next timestep will be the same as the current timestep. Since the rs-fMRI evolves relatively slowly, a large part of the variance can be explained by a flat line. A similar autoregressive model is the current standard model used in modeling resting state during task activation blocks. The second null model is constructed by setting the BNM function to zero and the output of the system is the initial condition predicted by the RNN. Therefore, when integrating over time it's just a constant value and is used to control how well the network activity can be directly predicted by the RNN network. The third null

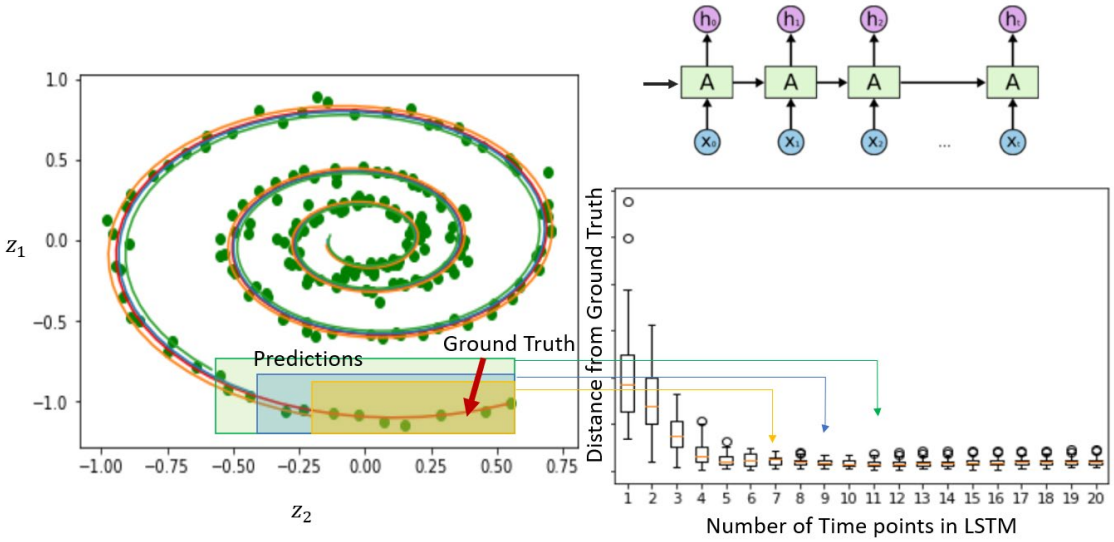
model is obtained when the global coupling parameter is set to zero in the Firing Rate Model and the differential equations become an exponential decay.

3.3 Results

3.3.1 Validation of the Neural ODE Algorithm on Spiral Data

The spiral data allowed us to test how well our algorithm estimates the initial conditions, in a simplified situation where the initial conditions are known. In Figure 3.4, we show that the predictions of a trained network converge towards the ground truth of the initial conditions for the sample spiral dataset. Starting from the first observation, at the bottom right corner of the spiral, the LSTM makes a prediction based on all the observations it has seen so far, and then a trajectory is produced via integration of the dynamical system of the spiral dataset. The trajectories after seeing 7, 9, and 11 datapoints are shown on the spiral itself. The distance between ground truth initial conditions is also quantified to the right as a function of number of points the LSTM observes. The accuracy during the first couple timesteps is low because the hidden state of the LSTM is not initialized properly, but after enough datapoints are included, it is able to predict the initial conditions within a reasonable margin of error.

Figure 3.4 Infer Initial condition for each Timestep on Spiral Data



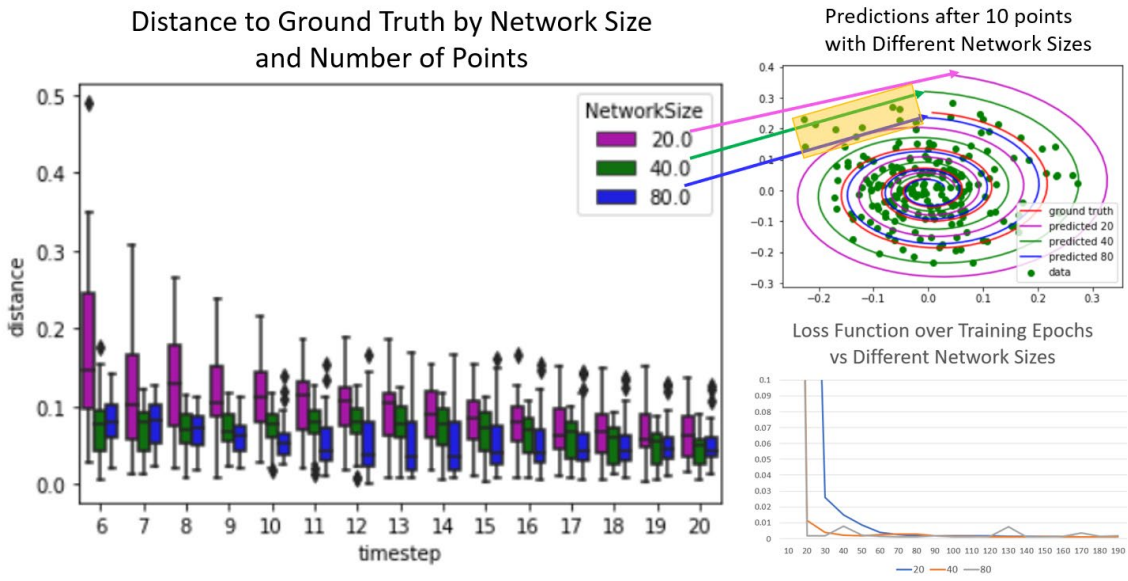
Estimating the initial conditions for the spiral data. The figure on the left shows an example of a trained network estimating the initial conditions using 7, 9, and 11 timepoints. Each timepoint is fed into the LSTM sequentially (shown top right) one datapoint at a time and outputs the initial condition for that timepoint. The trajectories are then integrated and then compared with the ground truth (shown in red). The Euclidean distance between the predicted and the true trajectory is shown in right for a distribution of 40 unseen spirals. The accuracy converges after the first few timepoints during which the LSTM has not yet been initialized properly. After the LSTM has seen enough examples, the estimates converge slowly towards the true spiral trajectory.

3.3.2 RNN Parameter Estimations on Spiral Data

Next, we test to see how our ground truth distance changes as a function of network size and parameters (Figure 3.5). For our given architecture, we can vary the number of layers (network depth), the size of the hidden layers, and the length of the LSTM network. The length of the LSTM network is a parameter due to Tensorflow implementation of LSTM and limits the number of previous seen observations. We followed Tensorflow's guidelines and biased it until the error converges during the time period contained in the

hidden size of the RNN (in Figure 3.5 after 7 or 8 timepoints where the length of the LSTM was set to 15 previous timepoints). We instead chose to test the effect of the hidden size. The depth of the network was kept to three layers, as the spiral dataset is small, and is varied for the neural dataset. The hidden size represents the feedback arm of the RNN, and larger hidden size allows for more complex relationship with previous data observations. We show in Figure 3.5 bottom right, that larger the hidden size (such as 80) the loss function tends to converge in fewer training epochs. Moreover, after they are trained, these networks are more accurate with fewer datapoints (Figure 3.5, left), and are more sample efficient in extracting information from previous data. There are also observable differences in their accuracy, as illustrated with a sample spiral predicted after each network has observed 10 datapoints. However, the difference in prediction between 20, 40 and 80 sized networks is less pronounced especially after observing many datapoints, suggesting that regardless after many data observations the system converges roughly to the same error. This points to a certain robustness of convergence of the loss function with regard to parameter variations for long time sequences.

Figure 3.5 Effect of Network Size on Initial Condition Predictions



The effect of the network size in its ability to converge to the true initial conditions with fewer samples. Three network sizes of the hidden state (20, 40, and 80) are compared while keeping the depth of the network and the number of batches constant. The plot on the bottom right shows that in all three architectures the loss function approaches zero and plateaus at roughly the same value. However, the larger networks are more sample efficient, meaning that with fewer samples, they are able to estimate better initial conditions. This is shown on the left, where the largest network (80) has the smallest distance to the ground truth. After the LSTM has viewed enough samples the differences between the networks vanishes. An example on a single spiral is shown (top right), where the predictions after seeing 10 datapoints are 0.11, 0.08, and 0.04 apart from the ground truth for the three differently sized networks at 20, 40, and 80 respectively. The actual pink and green data do not cross any of the initial data, because the algorithm only estimates the initial conditions which are not precise for smaller networks, and since we are integrating from the initial conditions, the resulting trajectories are far away from the future datapoints which have not yet been seen by the Machine Learning Network.

3.3.3 Metric Choices and RNN Parameter Estimations on fMRI data

For the neural dataset we have a much larger state space having at least 66 state variables. Therefore, we tested a number of different parameters as listed below in the Table 3.1. For a given BNM, the r-square accuracy was calculated for different parameters

of the machine learning network. The base parameters were chosen from Kashyap et. al 2020 with 600 iterations, 50 for length of RNN, 150 for hidden size, and 4 layers and across each row in the table we change one of these parameters at a time, to see how it effects the performance of the algorithm. The accuracy doesn't change a lot when the parameters are varied as shown in Table 3.1 suggesting that the network is maximizing the information transferred from previous states like in the spiral example.

Table 3.1 R-squared Accuracy at the 3rd timestep across different RNN Parameters

Parameter Evaluated				
Number of Iterations	200	600	800	1000
	0.729 (0.02)	0.743 (0.021)	0.739 (0.022)	0.744(0.02)
Length of RNN	30	40	50	60
	0.741(0.03)	0.744 (0.028)	0.743 (0.021)	0.741(0.024)
Hidden Size	100	120	150	180
	0.733(0.02)	0.734 (0.02)	0.743 (0.021)	0.741(0.022)
Number of Layers	3	4	5	6
	0.738(0.02)	0.743 (0.021)	0.739(0.02)	0.730(0.02)

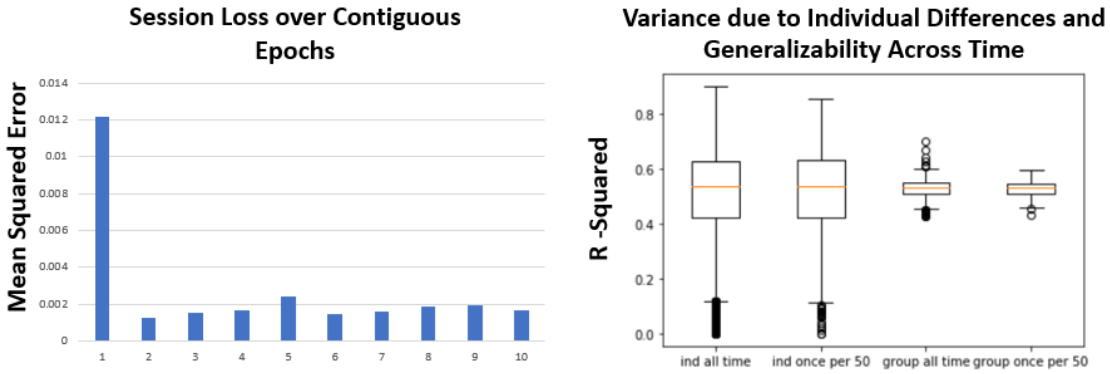
Evaluation of network parameters of the RNN. The mean r-squared (standard deviation) are given for different sized networks on the Firing Rate Model (k=0.6) on the third timestep prediction for 2500 short resting state fMRI trajectories.

The average loss over 10 epochs each with 50 timepoints is plotted in Figure 3.6 left. The epochs are contiguous over time, where the hidden state of the first epoch is the input for the second epoch. The loss is the mean squared error between the predicted and the next timepoint. The first batch has a large error when the RNN is not properly initialized but then converges to a minimum, similar to the spiral dataset after observing enough timepoints. The first epoch is ignored, and all the calculations are made after the first 50 timepoints.

Once trained, instead of using the mean squared error as used in the training, we use the more general r-squared metric in order to test how well the trajectories originating at

the predicted initial condition fit the future datapoints. Unlike the mean squared error metric, the r-squared metric would generalize even when the number of brain regions are changed or under different normalizations allowing it to be more compatible with future algorithms that test short term predictability. In Figure 3.6 right, we also tested how well the measure generalizes from testing once every epoch (50 timepoints) to testing every timepoint as well as the effects of computing the r-squared over a batch of data consisting of 60 individuals vs testing each individual at a time. The accuracy at the 3rd timepoint from the initial condition is plotted for these four conditions (i.e the permutations of individual vs group and one timepoint vs all timepoints). There is no difference in the mean or the variance in testing at every timepoint vs testing on all timepoints. This is not surprising as the algorithm was developed to predict the correct the initial conditions at every timepoint of the timeseries and the experiment shows that it generalizes and performs relatively similarly on all timepoints. On the other hand, there is a difference in the variance but not the mean when comparing the group r-squared values vs the individual r-squared values. This can be explained as individual differences in fMRI are averaged out in the group metric. This shows that our approach might be sensitive for individual differences, but we use the group metric for the subsequent results as they are more robust and allow us to test our hypothesis on the differences between different BNM. Moreover, since we utilize only a group averaged structural matrix, we are more interested how well the BNM fit to the group than to any particular individual.

Figure 3.6 Individual Variability and Generalization Across Time



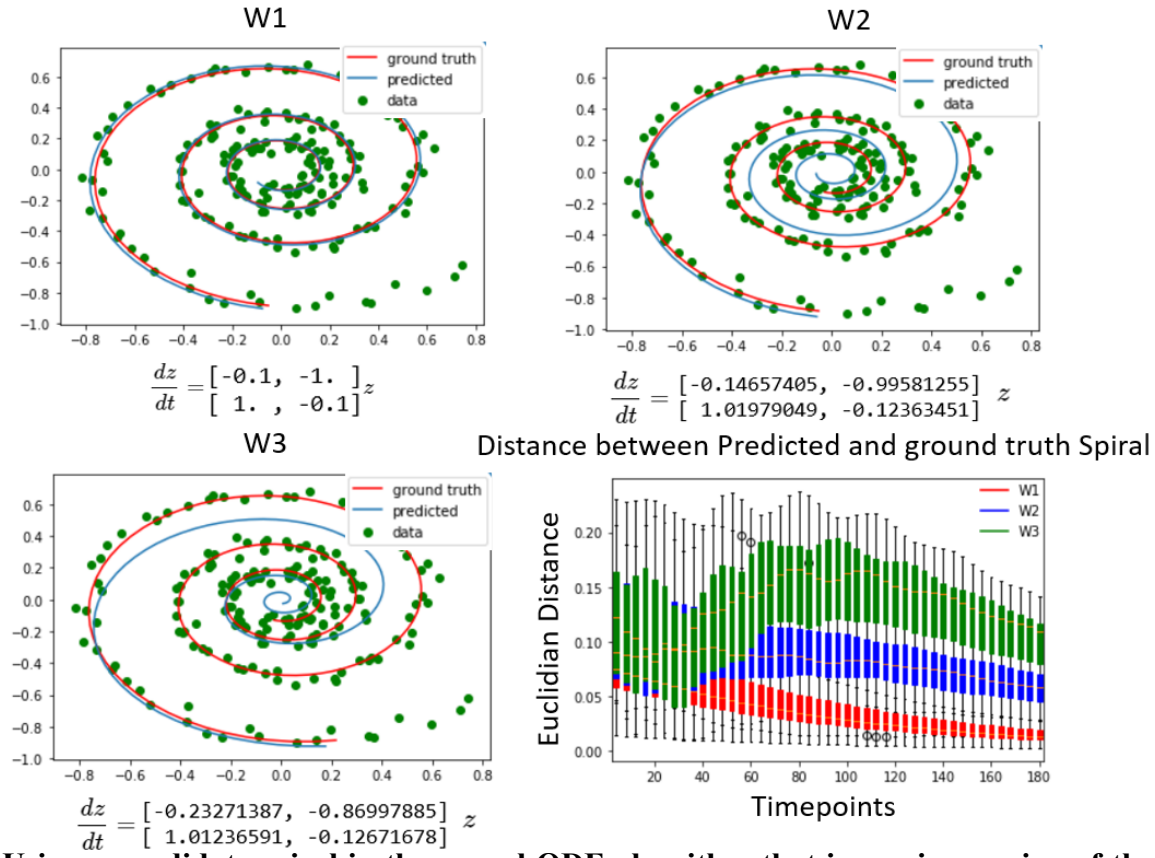
Left: The mean squared error calculated over each epoch of 50 timepoints from a continuous 1200 timepoints of fMRI data. The loss is constant after the first epoch which is much higher since the RNN hidden state is not properly initialized. The constant loss also suggests that after the first epoch the estimation of the initial condition is constant and has converged. The first epoch is not used in any of the subsequent estimates of evaluating the dynamical system. Right: The R-squared of a FRM (k = 0.9, σ = 0.3) is computed using 4 different methods. The first one (ind all time) evaluates the most number of tests, where the r-squared of each individual is calculated on every timepoint after the first epoch. The second one evaluates the r-squared of each individual fMRI once per epoch. The last two averages the r-squared across a batch of individuals at every timepoint and once per epoch. There is no difference in evaluating once per epoch or at every timepoint. There is a difference between the individual and group measures, which is expected as the group measure averages out the effect of individual variance. In our subsequent results we use the group measure, as our model does not take into individual differences in the structural matrix, and the group measure is more robust in evaluating the differences in the ODE which is what we are interested in.

3.3.4 Differentiating Between Dynamical systems on Spiral Data

We conduct an experiment to determine whether we can replicate the task of differentiating between dynamical systems with the spiral data. Using the same spiral dataset as before (see methods section 3.2.4), we try to fit different candidate dynamical systems, starting with the original dynamical system W1 and then perturbing the structural

matrix with noise, W2 and then even more noise W3. This can be seen in Figure 3.7, where the matrix for the dynamical system is shown below. The Neural ODE algorithm fits each of the spirals to the data, and our hypothesis is that the spiral with the original matrix should fit the data the best. Unlike in previous sections, we look at the difference of the spiral trajectories generated by integrating the initial conditions of the spiral dynamical system. The initial conditions after observing 10 datapoints is predicted and then integrated for 200 timepoints. The difference between the ground truth trajectory and the spiral trajectory is then plotted on the bottom right for 100 different instantiations. The spiral matrix W1, is the closest to the data when observed over long periods of time, but the difference is less pronounced at short time periods where the distributions overlap with W2 and W3. Therefore, since the Neural ODE fits any dynamical system tangentially in time, it is important to observe the long-time dynamics to differentiate between the models. At very long intervals, the distance starts to decrease as all trajectories converge to the origin. However, we do not expect to see this when fitting dynamical systems to brain data, as the real signal is constantly changing over time.

Figure 3.7 Differentiating between Dynamical Systems on Spiral Data



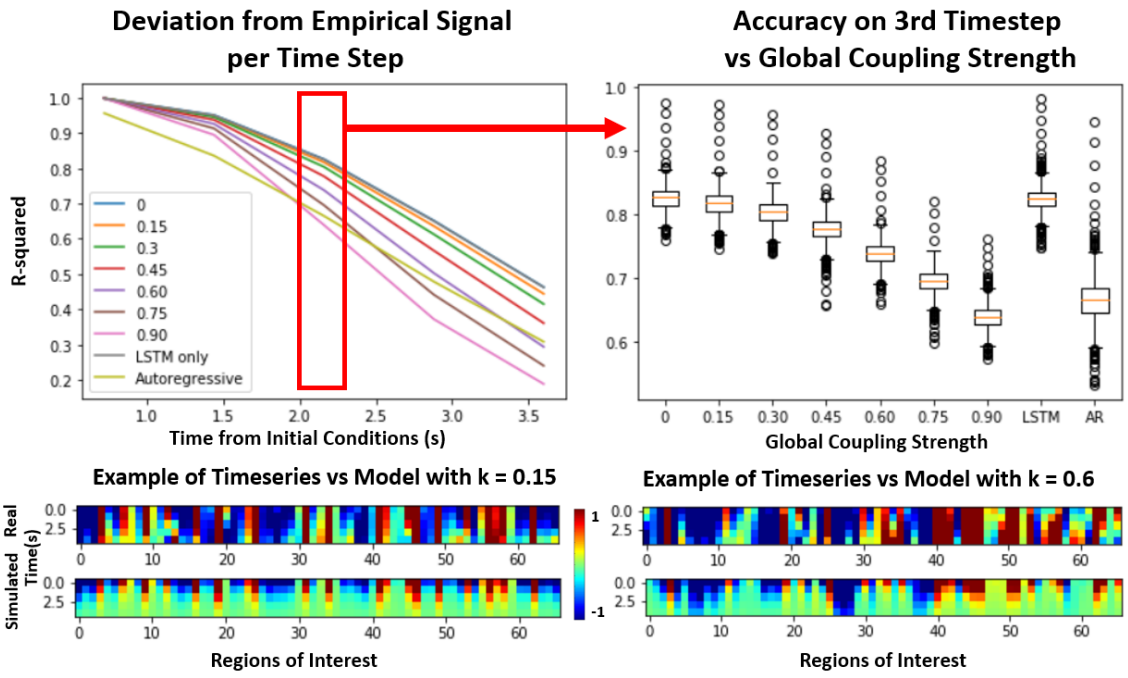
Using a candidate spiral in the neural ODE algorithm that is a noisy version of the ground truth spiral. This experiment is similar to how the algorithm is intended to be used with fMRI data. For the fMRI data, we utilize a measured structural connectivity estimated from white matter connectivity. Our hypothesis states that the measured white matter connectivity will follow the brain trajectories closer than using noisier versions of the white matter connectivity. The top left shows the alignment of the data, while fitting the spiral with a weight matrix identical to what was used to generate the data. The top right and bottom left figures show the alignment with spirals increasingly further away from the ground truth. The neural ODE algorithm is able to fit any of the spirals to the data for a given set of observations, but over time the candidate spirals that are further away from the ground truth diverge much faster from the future data points. This is quantified by the plot on the bottom right where the distribution of the three different spirals distance to the data is plotted from the predicted initial conditions. The distance between the spirals of different weights and the observed data points is initially close but diverges further away when compared to future timepoints. Note due to all trajectories going towards the origin, the distance at very large timescales converges

to zero, but this is not expected in the brain data, where the signal does not approach a single attractor.

3.3.5 Differentiating Between BNMs due to differences in Parameterization in the Firing Rate Model

Most BNM have many parameters and therefore it is essential to test whether this method allows us to parameterize BNM correctly. Here we examine an experiment with fitting multiple Firing Rate Models to resting state fMRI, with different global coupling values from 0 to 0.9 in 0.15 intervals. The global coupling parameter affects the amount of network interaction, whose value can vary from a maximum of 0.9 to a minimum value of zero when the interaction is non-existent (exponential decay, see Methods).

Figure 3.8 Effects of Global Coupling without Noise in the Firing Rate Model



Evaluating the Firing Rate model (Eq 1) with different global coupling parameters. Top Left: Error per timestep from the estimated initial conditions for various different parameters compared. The y axis represents the r-squared value, between the simulation and the model. Examples of the model timeseries vs the resting state timeseries are given on the bottom at two different parameterizations but the

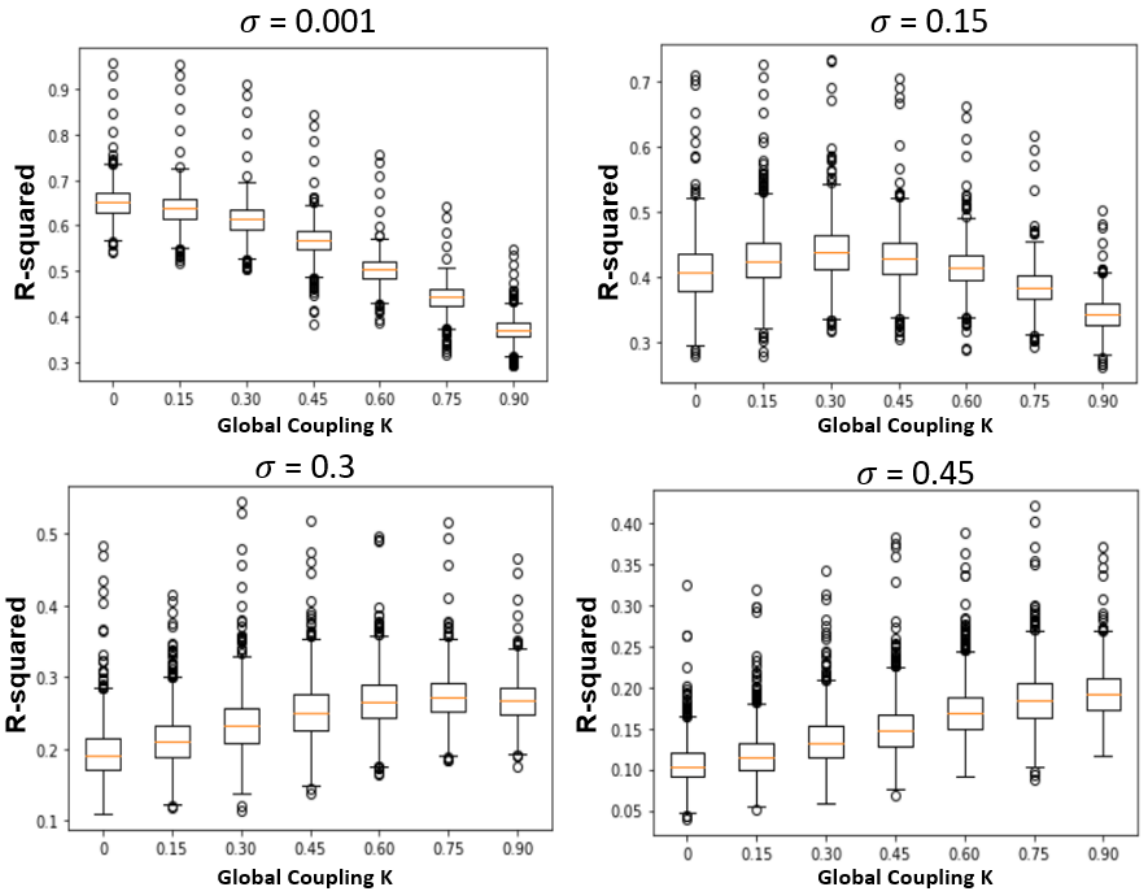
distribution shown in the top panels quantify their performance across 2500 trials. Top Right: At the third timestep, we plot the histogram of the r-squared across all the models. The model with zero global coupling performs the best, and as the global coupling strength increases the accuracy goes down. Most models perform better than the Autoregressive Model (AR).

In Figure 3.8, we compare the different parameterized models as well as the Machine Learning Null model. At the first timestep (Figure 3.8, top left), all the models perform relatively similar as a result of minimizing the loss function to zero. The models perform differently as we move forward in time. At the snapshot of three timesteps, we compare the different models (top right) and it is clear that the model with a zero global coupling performs better than all other models. At this coupling (k=0), it is nearly equivalent to the Machine Learning Null Model (see Methods). This suggests that the introduction of any BNM, decreases the accuracy of the model and the model performs best using just the LSTM predictions.

However, interestingly this trend completely reveres when noise is introduced into the models. In Figure 3.9, we vary both the standard deviation of the noise as well as the global coupling parameter. At low noise levels, we see that low global coupling outperforms the models with high global coupling. However, with increasing levels of noise, the BNMs with stronger global coupling outperform the more naïve models as well as the Machine Learning Null models. This result shows both the important of noise in differentiating between the parameters of the Firing Rate Model. The overall r-squared of the models also decrease with the introduction of noise, but the rate at that they diverge seems to be a function of the global coupling parameters. The Firing Rate models are traditionally simulated with $k = 0.9$ and $\sigma = 0.3$, and noise is seen as essential in simulating the BNMs. However, since both are unknown and the overall r-squared decreases with the

introduction of noise, just based on these two parameters it is difficult to conclude which parameterization yields the best result, other than the very trivial conclusion of an exponential decay only model with no noise.

Figure 3.9 Effect of varying Global Coupling and Noise on 4th Timestep Accuracy of the Firing Rate Model

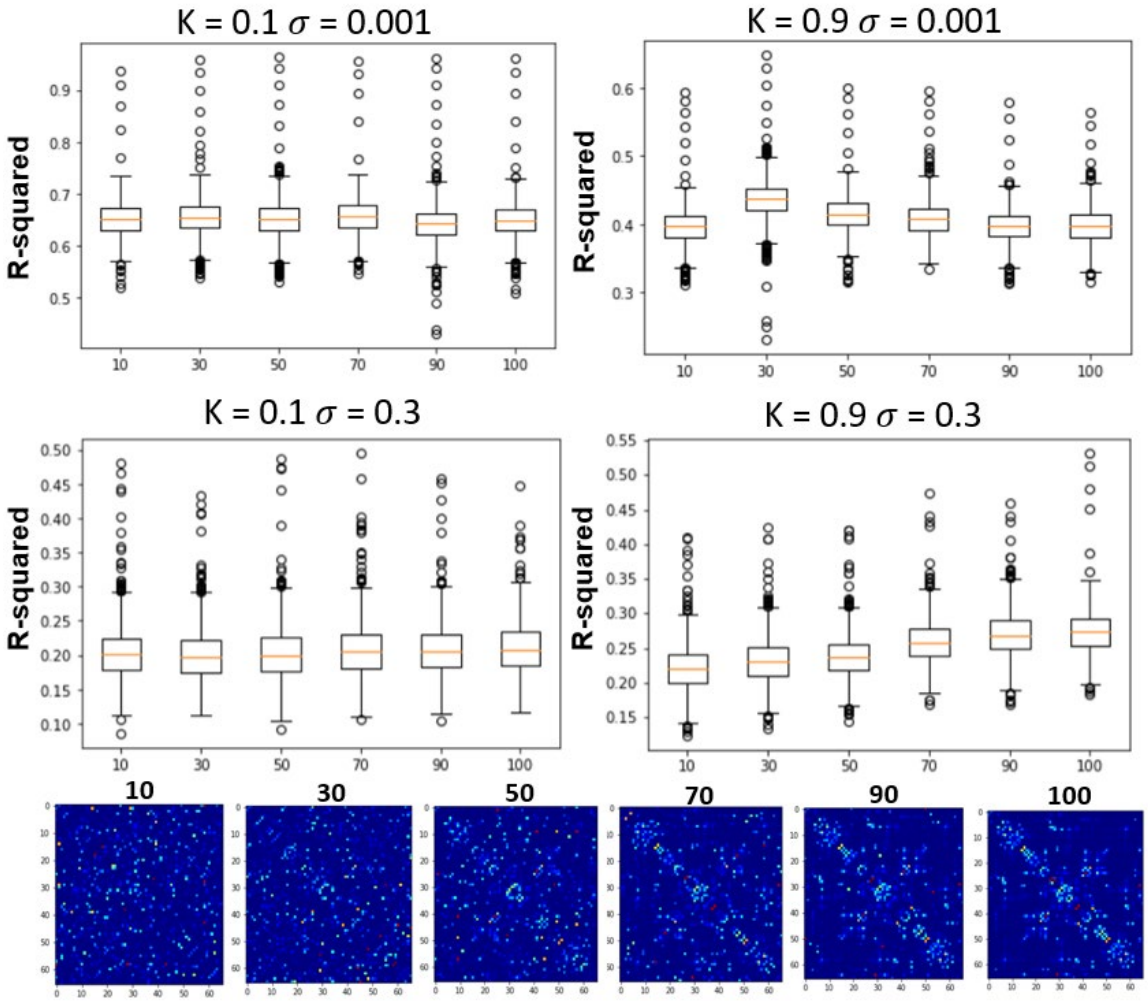


The effect of varying the two parameters in the Firing Rate Model, the global coupling K and the standard deviation of the noise σ. The introduction of noise lowers the accuracy of all models but does so in an uneven fashion. At low levels of noise, the exponential only model (k=0) outperforms the BNMs with structural connectivity matrices. This trends reverses with the introduction of increasing noise power, where the BNM with stronger global connectivity matrices seem to outperform the naïve exponential models.

3.3.6 Differentiating Between BNM due to differences in Structural Connectivity

In the previous section, we only changed the parameters of the Firing Rate model, the global coupling strength as well as the magnitude of the noise. Here we observe the effects of simulating 6 different structural connectivity matrices at two different global coupling strengths, one close to no network effect (k=0.1) and one close to high network connectivity (k=0.9) and at low and high noise levels (σ=0.001,σ=0.3). The SC is varied from the measured SC, and then slowly corrupted by flipping edges, and the number of edges in common with the original SC is varied from 10 to 90 percent. We examine the r-squared value at the fourth timestep, in Figure 3.10.

Figure 3.10 Effects of Global Coupling, Noise, and Structural Connectivity on 4th Timestep Accuracy of the Firing Rate Model



Examining the effects of parameterization and estimating the correct SC. At top left, we show the results of changing the structural connectivity for a low global coupling model. It does not vary as a function of the structure and performs relatively similarly to the LSTM only null model. At high global coupling, the models show that they are more of a function of the correct structural network, although they perform worse than at high global coupling.

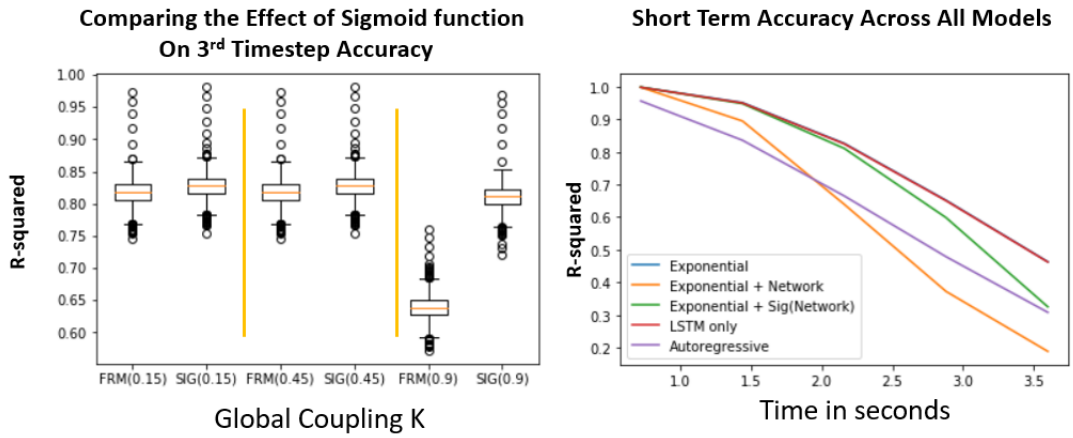
In the low noise levels (σ =0.001), the model shows no relation to changing the structural connectivity for either of the coupling strengths. At the high noise levels (σ =0.3), although the model has a lower r-squared than at the low noise levels, we see the effects of the network, where the original SC outperforms the corrupted versions for both low and high coupling strengths. The trend is once again more pronounced for the high global coupling (k=0.9) when compared to low global coupling (k=0.1).

3.3.7 Nonlinear vs Linear BNM

The Firing Rate model is thought to be the simplest BNM, as it is a linear function of the structural matrix (Cabral et al., 2012). Here we extend the model by simply adding a sigmoid, which is a common feature in other BNMs such as the Wilson Cowan, limiting the effect of the network at high levels of activity (Sanz Leon et al., 2015). Our goal is to test whether the Neural ODE algorithm can determine the difference between the simpler linear and the more complex non-linear model. We simulated the Firing Rate Models and the Sigmoid Firing Rate Model at three different parameterizations (k=0.15, k=0.45, k=0.9) at low noise levels, and plotted the accuracy at the third timestep shown in Figure 3.11. The addition of the sigmoid term increases the accuracy of all baseline Firing Rate Models but has the largest affect when the global coupling is large, and the network component is active. The sigmoid model when compared to all other models does about as well as the exponential decay only and the LSTM only models, but utilizes the BNM equations and

the structural network. Moreover, the model does slightly better at k=0.45 than at k=0.15 or k=0.9, suggesting that maximizing the global coupling term yields a non-trivial solution.

Figure 3.11 Performance of Nonlinear sigmoidal Firing Rate Model without Noise



Left: Performance of the sigmoidal Firing Rate Model vs Firing Rate Model over different parameterizations (k=0.15, 0.45, 0.9) on the third timestep. Adding the sigmoid only increases the performance from the base Firing Rate Model, but has the largest affect when k=0.9 when the network effect is the strongest. Right: A comparison between, the Firing Rate Model, sigmoid Firing Rate Model and the three null models over time. The exponential and LSTM only curves are on top of each other, so only 4 traces are visible.

3.4 Discussion

We proposed using the Neural ODE methodology to fit different candidate BNMs to resting state data and then subsequently evaluate them on how well they evolve compared to the real data. In order to test this methodology, we first try to use the method on a well studied spiral dataset to determine if our algorithm solved for the initial conditions. Then we tested the method to see if it can identify by simulating different differential equations, which system was used to simulate the toy spiral dataset. Then moving to neural rs-fMRI data, we fit different BNMs using the Neural ODE algorithm by varying their 1) parameterizations, 2) structural connectivity, and 3) by examining the effect of introducing

non-linearity. For the BNMs, although the ground truth is not known, we construct our experiment in a manner such that changing these variables has a known and well documented effect.

3.4.1 Validation on the Spiral Dataset and Comparison to other Architectures

From the spiral example, we first validated that our algorithm implements a form of Neural ODE by showing that the RNN were accurately able to predict the unknown initial conditions using the Neural ODE loss function framework. We chose to do this because our own implementation of the Neural ODE algorithm, differs from the original in the following manner. The original implementation uses a backward time architecture, where the timeseries is inverted and fed into the RNN network, such that the first timepoint is fed into the RNN last and the final prediction is used to infer the initial condition of the whole timeseries and then integrated forward in order to compute the loss function (Chen et al., 2018). They do not evaluate the RNN prediction at every timepoint like in our implementation, but explicitly state that such an architecture would speed the training process. This fact is also stated on the Tensorflow RNN implementation page as their recommended use of the RNN (tensorflow.org/guide/keras/rnn). The innovative backwards time architectural method gets rid of the initialization problem that exists in our forward time implementation but runs into a causality problem where future inputs influence the predictions of previous initial condition. Because BNMs are defined as a function of previous network activity, we use a time forward architecture in order to solve for the initial conditions. The other significant difference, is that the Github implementation of the Neural ODE also uses a LSTM after the ODE integration (Chen et al., 2018). This methodology is extensively evaluated in Chapter 4 but confounds our goal of comparing

the fit of different dynamic systems, so it is not used in the implemented methodology in Chapter 3. Despite these architectural differences, we show that our Neural ODE implementation solves for the correct initial conditions, and after the hidden state of the LSTM is initialized is able to solve for trajectories that are similar to those reproduced in the original paper.

The other main result from using the spiral data, is to show a working example of how we intend to use the algorithm in order to differentiate between dynamical systems that might have generated the data. We show that by observing the long term trajectories of several candidate models, we can infer which system produced the data by measuring the r-squared of future timepoints. However, when looking at short time differences, since all the models were all fit to reproduce the next time step, it was less clear how to differentiate between the models as they gave similar estimates of each other. The results show that it is possible to use this as a system identification, but the systems need to be simulated long enough for the differences to manifest, since they are all trained to minimize the prediction error at the first timestep.

3.4.2 Parameterization of the Firing Rate Model

The Firing Rate Model, has two parameters in the model (Eq. 1), the magnitude of global coupling and the level of noise. From previous literature and experiments in Chapter 2, the global coupling value of a traditional Firing Rate Model is set slightly less than 1, around 0.9, which is just before the system becomes unstable. Closer towards k=0, the dynamics become an exponential decay according the Eq 1. The noise is usually set around 0.3 (Kashyap et al., 2019, Cabral et al., 2012), where closer towards zero it turns the system

to the well-known consensus problem (Mesbahi & Egerstedt, 2010), and at higher degrees of noise the system becomes completely chaotic and non-deterministic. Searching for the correct parameterization of the Firing Rate Model between the global coupling and the magnitude of the noise is therefore an important problem, because in the in between regime the resulting dynamics is closest to rs-fMRI data. Using the Neural ODE, we fit many different Firing Rate Model with different values of global coupling and noise. We expected that the Firing Rate Model ($k = 0.9$, $\sigma = 0.3$) or around those values would fit the data the best from previous studies. This seems to be true only at high values of noise, and at low values of noise the models that were closer to exponential decay performed better. We hypothesize that at high values of noise, the models without strong structural connectivity, the noise caused the signal to deviate largely from the original predicted trajectory. The models with strong structural connectivity, the noise is averaged out by the matrix and the trajectory follows the original trajectory more closely. However, since the exact value of noise is not known, it is difficult to conclude what parameterization fits the data the best. No noise is unrealistic, as it leads simplistic deterministic dynamics not seen in rs-fMRI. However, since both the coupling strength and the noise is unknown just varying these two does not give a clear answer that would solve the parameterization problem, especially when trivial solutions such as the exponential decay and the autoregressive model perform relatively well.

3.4.3 Parameterization of the Firing Rate Model while changing the Structural Connectivity

Solving the parameterization problem looks more promising when observing different parameterizations of the BNM while also varying the structural matrix at the same

time. According to our scientific paradigm, we expect the BNM with the original structural matrix to outperform the other models with increasingly corrupted structural matrices. Therefore, under the right parameterizations values we expect to see the greatest difference when we change the structural matrix. When observing under low values of noise, there is no difference in the performance of the FRM as a function of the structural matrix. At higher values of noise, the trend clearly emerges that accuracy is a function of how many edges the structural matrix has in common with the original structural matrix. Moreover, it still seems to be a function of the coupling strength, where at higher coupling strengths we see a larger difference between BNMs with the original structural connectivity and the randomized structural connectivity. Rather than looking at the r-squared, which changes drastically with the amount of noise introduced, by looking at the slope as a function of the structural connectivity we can establish which parameterization would fit the data the best, and therefore solve the parameterization problem using short term metrics.

3.4.4 Comparison between Non-linear and Linear Models

In order to test the effect of a sigmoid function, we fit two different BNMs, the Firing Rate Model and the Sigmoidal Firing Rate Model. The two models are nearly identical and are parameterized at the same value, except for the presence of a sigmoid in the Sigmoidal Firing Rate Model. We show that the sigmoid increases the accuracy of the model, especially at high global coupling values when the network component is most active. At lower global coupling values, when the network is not very active, the difference is much smaller as the dynamical system reduces to an exponential decay. Similar to the parameterization problem of the Firing Rate Model, the best Sigmoidal Firing Rate Model is at the limit when the dynamics reduces to an exponential decay. However, the sigmoid

component improves the baseline Firing Rate Model at realistic coupling levels, suggesting we can utilize this tool to differentiate between models that are more and less representative of rs-fMRI. Further research with more complex nonlinear models, might outperform the naïve null models with simple functions in terms of their short-term predictions.

3.4.5 Assumptions and Limitations

The error from the model's prediction comes from two different sources, 1) the mismatch between the differential equations and the actual dynamics, and 2) from the error in predicting the initial conditions. We assume that for any assumed dynamical system the error from the RNN is uniform no matter what the function is, and the subsequent error calculated from the trajectories is due to the mismatch between the data and the dynamical system. However, this might not be true, and more complex models might have a larger errors in estimating initial conditions and therefore is a potential confounder in our analysis.

A major limitation of this approach is that the algorithm needs a good estimation of the underlying dynamical system that represents the data. This requires vast knowledge of what model including the specific parameterization might fit the dataset. For the simple Firing Rate model, maximizing the parameterization utilizing Neural ODE, led to a trivial result. Although this was not true in the sigmoidal model, the approach needs to be applied cautiously when used to fit functions to the fMRI data.

Moreover, another major assumption and limitation of the approach is our choice of metric, r-squared used to compare the distance between two high dimensional vectors. It assumes that better models have a higher r-squared value, although they might be

explaining trivial components of the signal. Other metrics such as derivative, or the relative phase between different regions of interest might prove as a much more useful metric to compare the predictions against the empirical signal.

3.5 Conclusion

We attempted to investigate whether we could solve for the initial conditions of a Brain Network Model for a given observation of rs-fMRI data using a novel Machine Learning tool known as Neural ODE. Using these initial conditions, the goal was to be able to differentiate between different BNMs that best recapitulate the data. We showed that we are able to differentiate between more and less realistic structural networks and between linear and simple nonlinear functions. Although, the approach shows promise to differentiate between models in certain cases, it has to be used with caution as trivial functions such as a line or an exponential can be fit very well to the slow changing fMRI data.

CHAPTER 4.

MODELING RESTING STATE FMRI EVOLUTION USING RNN INFERENCE AND BRAIN NETWORK MODELS

SUMMARY

Motivation: Brain Network Models (BNMs) are becoming more sophisticated in capturing aspects of the temporal dynamics of neural populations measured by functional magnetic resonance imaging (fMRI). However, as was shown in Chapter 2, the BNM models showed varied results and performed poorly in matching empirical rs-fMRI data on short term metrics and transient dynamics. In Chapter 3, we showed that we can directly evaluate short term dynamics by utilizing a trained RNN model to infer the initial conditions from the data and then generate the future timeseries purely based on the BNM. In this chapter, we use the trained RNN model in conjunction with the original BNM to generate the future timeseries, by piping the output of the BNM back into the RNN model. We test whether this combined system improves our ability to generate long term resting state data. The Neural ODE paper also utilizes a similar approach of using both the RNN and a dynamical system in order to generate a timeseries and has shown that it generates more realistic trajectories as long as the dynamical system fits the given the data (Chen et al., 2018). We use a traditional BNM and a RNN only null model to compare the output of our Neural ODE system using the metrics established in Chapter 2, on long term simulations of timeseries data. Most of this work has been published in Network Neuroscience except for Figure 4.9 (Kashyap & Keilholz 2020).

Hypothesis: Utilizing a trained Neural ODE model that uses both the RNN and the BNM, improves the model's overall ability to reproduce rs-fMRI dynamics as defined by the dynamic metrics identified from Chapter 2 compared to the traditional BNM.

Approach: Train a Deep Learning technique (Neural ODE) to solve for the initial conditions based on the given BNM and then integrate to predict the next observation. The system is trained in the exact same manner as in Chapter 3 (Section 3.2.1). After training, we generate our future time sequence by taking the output of the Neural ODE algorithm and feed it as our next input. The process, known as inference, is then repeated to generate long time sequences of data and utilizes both the RNN and BNM components at each timestep. We simulated two distinct Brain Network Neural ODE systems using different latent BNMs, the Firing Rate and the Wilson Cowan model, and determined the parameters within the models using machine learning. After training the Neural ODE models, we simulated 30 instantiations that were 15 min long and compared it with 30 instantiations from traditional Brain Network Models and 30 scans from measured rs-fMRI in humans. We also simulated a Machine Learning null model trained without a BNM and compared its ability to generate signals compared to a Neural ODE system utilizing a BNM (section 4.3.4).

Metrics and Evaluation: In order to evaluate the new generative models compared to the traditional BNMs and rs-fMRI data, we used the dynamic metrics identified in Chapter 2 to compare synthesized rs-fMRI data with empirical measurements. These metrics were shown to be more effective in differentiating between models, and some of these metrics like Quasi Periodic Patterns relate to physical processes that a network based rs-fMRI should be able to emulate. We started by using functional connectivity to compare the three

datasets as it is a standard metric in evaluating BNM (section 4.3.3). For our more complex dynamic metrics, we chose Quasi Periodic Pattern analysis (section 2.2.6.2) and K-means sliding window analysis (section 2.2.6.3), because they are particularly difficult to reproduce as they have complex spatial and temporal relationships between the brain regions. The QPP metric was evaluated by estimating the mean squared error (MSE) between the patterns generated from the models and the patterns determined from empirical data (section 4.3.3). For the K-means centers we used correlation to compare between all the cluster centers of the generated data and all the cluster centers of the simulated data (section 4.3.3). Since the centers are not ordered, we took the maximum correlation between one center of the simulated data and all the centers of the empirical data.

Conclusion: This paper shows that the Neural ODE inference approach using both the RNN and BNM can be utilized effectively in order to simulate rs-fMRI. On all metrics that compare BNMs with rs-fMRI, both on static metrics such as FC as well as dynamic metrics it performs far better than the traditional BNMs and is almost comparable to the original data. Most of these metrics can also be reproduced by a Machine Learning null model utilizing RNN only inference. However, on the most complex metrics such as QPP, the Neural ODE system with a BNMs outperforms the Machine Learning model in reproducing complex spatial temporal patterns that occur in measured rs-fMRI data. Although this approach is a step away from the interpretability of BNMs and the activity of its modular components due to the RNN involvement in the timeseries production, we demonstrate that we can learn a generative model that is able to reproduce features in rs-fMRI that previous traditional BNMs have been unable to reproduce. Moreover, utilizing this approach we can train on fMRI data collected on different groups of people that have

different sets of dynamics, and use it to study how they diverge when synchronized to the same initial conditions.

4.1 Introduction

Over the past decade, our understanding of spontaneous whole-brain activity and coordination between brain regions has largely been obtained through noninvasive resting-state functional magnetic resonance imaging (rs-fMRI) studies (Biswal, Yetkin, Haughton, & Hyde, 1995; Margulies et al., 2016; Smith et al., 2009; Zalesky, Fornito, Cocchi, Gollo, & Breakspear, 2014). Resting state, a state without an explicit task or stimulus, has surprisingly complex whole-brain trajectories that are well structured and highly dependent on the previous brain activity (Allen et al., 2012; Billings et al., 2017; Shakil, Lee, & Keilholz, 2016; Zalesky et al., 2014). Current generative models such as brain network models (BNM) attempt to characterize-whole brain activity as the interaction between a single neural population and the activity of its network neighbors defined by its structural fiber connections as measured through diffusion tensor imaging (Cabral, Hugues, Sporns, & Deco, 2011; Honey et al., 2007). Although there are many variants of the model that use different sets of differential equations to describe the activity at each node, all brain network models heavily rely on the description of the structural network through which they interact (Sanz-Leon, Knock, Spiegler, & Jirsa, 2015).

Long simulations of brain network models, starting from random initial conditions, are able reproduce time-averaged properties of rs-fMRI. These properties such as average functional connectivity (FC) are defined as the correlation between brain regions over long periods of simulated time greater than 10 min (Cabral et al., 2017). The time-averaged properties are thought to be more related to the structural network and are thus able to be reproduced by many different BNMs since they all share the structural network as an input (Cabral et al., 2017; Kashyap & Keilholz, 2019; Skudlarski et al., 2008). However, the

BNMs are worse and more variable at reproducing transient dynamic features that occur at shorter timescales, on the order of seconds and minutes, which are much more dependent on the exact description of the differential equations (Cabral et al., 2017; Kashyap & Keilholz, 2019). Since BNMs are not synchronized with actual measurements, there exists a gap in understanding how much these models are able to capture the actual changes to fMRI signal between measurements.

We propose a novel method synchronizing the BNMs to empirical data using recurrent neural networks (RNNs) in order to learn the initial state of the BNM from measured rs-fMRI data. Then using a Euler integration scheme, we can use the differential equations from the BNM to predict the next rs-fMRI data point and then evaluate directly how well it compares against the next measured time point. By applying this technique, brain network autoencoder (BNA), we can quantify how much of the variance of future resting-state activity can be accounted for from previous brain activity using RNNs with BNM constraints as opposed to other sources that influence large brain activity such as external stimuli. This approach of using RNNs with constraints in order to model biological systems has been recently gaining attention as a efficient tool in solving for and modeling unknown systems of differential equations. The approach combines the power of machine learning and allows for the incorporation of known biological variables that allow for interpretation on how the signal evolves (Chen, Rubanova, Bettencourt, & Duvenaud, 2019; Pandarinath et al., 2018). Moreover, the approach has an advantage over traditional methods fitting parameters of the BNM, which simulate over a large parameter space and then use time-averaged measures such as FC for model selection. Rather, the mismatch between the model empirical signal at every time step is fed into the machine learning

system to fit model to the data. This approach might also help distinguish between different variants of BNMs as it provides a useful measure in evaluating their performance on short time predictions, as they are all able to produce time-averaged measures as average FC (Cabral et al., 2017).

We evaluate the effectiveness of our model in its accuracy of short-term predictions (< 5 s) that are synchronized to the empirical data and the dynamic properties of the simulated signal over long time intervals (>10 min). In order to train our model, we used fMRI scans from 407 Human Connectome subjects (Van Essen et al., 2013) reduced to 66 regions of interest (ROI) according to the Desikan-Killiany atlas (Desikan et al., 2006). The corresponding structural connectivity to the Desikan-Killiany atlas was estimated using tractrography on five HCP subjects (Kashyap & Keilholz, 2019). After we trained the model, we then evaluate our model on a set of 40 unseen subjects and over 1,000 different initializations to see how well the system generalizes from the training set in order to produce correct predictions on unseen brain activity. We test two variants of this model that have different latent states, the firing rate and the Wilson-Cowan model, in order to see whether this method can distinguish the performance between different variants of brain network models (Cabral, Hugues, Kringelbach, & Deco, 2012a; Deco, Jirsa, McIntosh, Sporns, & Kötter, 2009). We utilize an autoregressive model as a null model to compare our effectiveness on short-term predictions. A similar linear variant (general linear model) is currently being used to distinguish the activity between rest and task blocks and regress out resting-state activity in order to infer task networks (Smith et al., 2009).

We test long periods of simulations of our generative model using dynamic analysis techniques in a similar manner currently used for evaluating traditional BNM (Kashyap &

Keilholz, 2019), in order to see whether it can reproduce dynamic properties observed in rs-fMRI that repeat over minutes. We utilize the k-means analysis on short-windowed FCs that looks for structure in the signal in the timescale of around a minute, and a quasiperiodic pattern (QPP) technique that searches for a 20-s repeating pattern. We use a traditional firing rate model as a null model to compare against the long instantiations of BNAs.

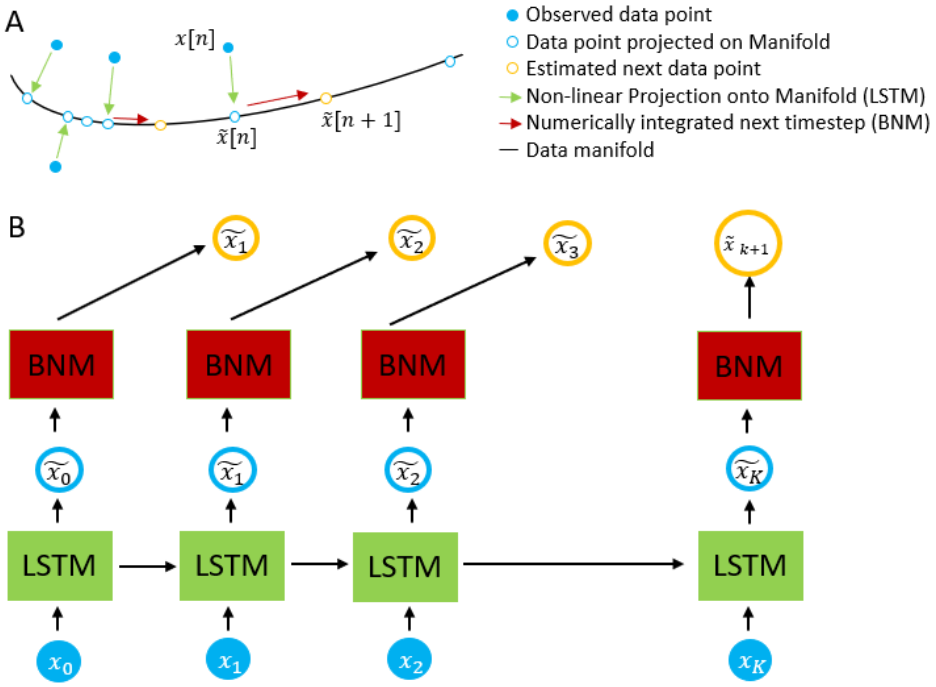
The BNA method offers three main strengths in comparison with other methods that are currently used to simulate whole-brain signals:

1. It solves the problem of comparing simulated and empirical data without using time-averaged metrics such as average FC, by directly using real data to initialize the model and by measuring differences in the predicted transient dynamics on a moment-to-moment basis.
2. It allows us to use black-box machine learning techniques while simultaneously estimating interpretable latent variables such as firing rate or excitatory and inhibitory currents that can be verified using multimodal recordings.
3. In long simulations of the BNA, the simulated signal exhibits dynamic properties seen in empirical rs-fMRI that occur over a timescale of minutes, which are not reproducible using traditional BNM techniques.

Therefore, we believe that the brain network autoencoder will be a useful tool to help us understand brain dynamics at the macroscale level.

4.2 Methods

Figure 4.1 Schematic of Neural ODE algorithm



The measurement x(n) is passed into the LSTM in order to estimate x˜(n)which lies in the data manifold. Using the BNM forward equations and x˜(n) as our initial conditions, we estimate x˜(n+1). The system is trained by difference in our predicted vs actual measurement at x(n+1).

4.2.1 Mathematical Background of the Whole Architecture

The Brain Network Autoencoder is constructed using the constraints from the Brain Network model, in conjunction with a Recurrent Neural Network variant known as Long Short Term Memory (LSTM). The overall design is shown in Figure 4.1 and implemented using Python Tensor Flow. The architecture is a sequential Autoencoder, as it is trained with the previous time point to predict the next consecutive time point and uses a latent space where the dynamics are constrained to a smaller space defined by BNM equations to reconstruct the next time point.

Formally, in order to predict the next time point, for each neural measured time point x(n) we map it to the space F(x(n)). F is the transformation performed by the RNN and lives in R M×M×T, where M represents the M distinct ROIs being modeled and T is the length of previous time-points the LSTM depends on. The next time point is computed as x(n + 1) = BNM(F(x(n))). In essence, the LSTM does a non-linear coordinate transform of the vector x(n) into the Brain Network Space where the dynamics are well defined and we can predict the next time point. This process is shown pictorially in Figure 4.1a, where we show the projection of each data point, shown in filled blue circle, into the manifold represented by the BNM shown in a hollow blue circle. On the manifold, we can use the BNM equations to update it to the next time step shown in orange. Fig 1B shows the actual architecture used to update the timesteps.

Another way to understand the system, is to imagine a stacked RNN network where each layer is represented as h(n +1) = h(n) + g(x(n)), where h(n) is the previous output of the Recurrent Neural Network and x(n) is the input and the function g is a non-linear function learned from the data. The last layer in the Brain Network Autoencoder is fixed such that g(x(n)) = BNM (x(n)) where BNM represents the ordinary differential equations from the Brain Network Model.

4.2.2 Brain Network Model

The Brain Network Model is a discrete neural population model of the whole brain and is constructed by specifying a parcellation or atlas of the brain and dividing it into regions of interest (ROI). Each ROI becomes the node and the edges between these nodes are represented as the number of fibers between regions which is usually calculated using

tractography applied on Diffusion Weighted Images (DWI). A Brain Network Model describes the change in neural activity in the lth region of interest $\dot{x}_l$ in two terms. The first term represents the network component which is described by a function F that depends on its own activity, its neighbor x_j activity, and the physical properties of the fiber represented by the vector ρ (i.e., the number of fibers between regions, the delay in propagation). The second term consists of a function G that represents external input, whose activity is represented by a k-dimensional vector u representing all sub-cortical and sensory inputs, and the vector π representing again the physical properties that project these inputs (i.e., thalamic tracts into cortex). The last term represents noise from the neuronal populations and from higher order terms not included.

$$\dot{x}_l = \sum_{j \in Neighbors\ of\ l} F(x_l, x_j, \rho_{ij}) + \sum_{k \in \text{Task inputs}} G(u_k, \pi_{lk}) + N(0, \sigma)\ Eq.1$$

For resting state activity, the assumption is that u_k (t) = 0 ∀ t and the first term dominates the change in activity. The function F for example can be as simple as the linear Firing Rate model:

$$\dot{x}_l = -x_l + k \sum_{j \in Neighbors\ of\ l} w_{lj} * x_j\ Eq.2$$

where wij represents the number of fibers between i and j, and k represents the global coupling parameter. The firing rate model, is a linear model and can be represented as A ∗ x. The matrix A is the graph Laplacian (A = k(SN – I)), where k is chosen to be less than 1/largest eigen value of A and SN is the structural matrix as measured through tractography using diffusion tensor imaging (Methods section X) (Hagmann et al., 2008). The dynamical

system does not add in any unstable dynamics due to all of its eigenvalues being less than zero because k was set to be less than 1/largest eigen value of SN (Mesbahi & Egerstedt, 2010). Therefore, under zero input the network propagation dies out over subsequent timesteps and the dynamics collapses into the well-studied consensus equation.

In a more complex model, the state variable x can also be represented by multiple variables such as the Wilson Cowan model shown in the equation below, which uses excitatory and inhibitory currents to describe the change in activity at every ROI. While in the Firing Rate Model the output is assumed to be the rs-fMRI signal, in the Wilson Cowan the fMRI signal is assumed to be just the excitatory signal since it dominates metabolically. This assumption for the Wilson Cowan has been done in previous works as well (Sanz Leon 2015, Ritter 2017). The update for the Wilson Cowan is the following three equations:

$$\dot{E}_i = -E_i + \alpha * S[c_{ee}E_i - c_{ei}I_i + k \sum_{j \in Neighbors\ of\ i} w_{ij}(E_j)] \quad Eq\ 3.$$

$$\dot{I}_i = -I_i + \alpha * S[c_{ie}E_i - c_{ii}I_i + k \sum_{j \in Neighbors\ of\ i} w_{ij}(E_j)] \quad Eq\ 4.$$

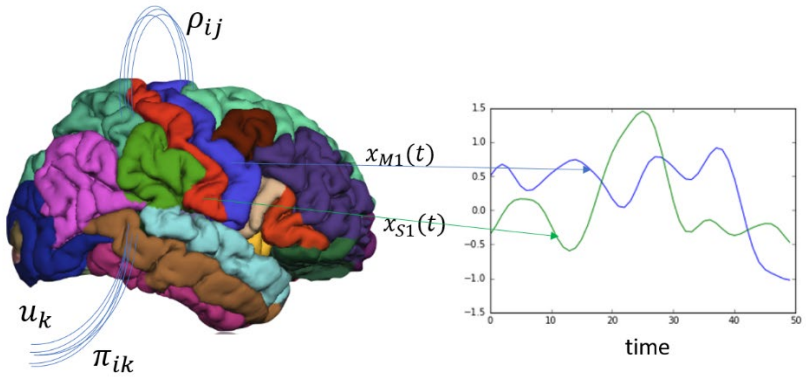
$$S(x) = \frac{1}{1 + e^{-x}} \quad Eq\ 5.$$

We choose the Firing Rate Model and the Wilson Cowan model as our BNM to implement in order to test the method with a simpler linear Firing Rate model and the more complex latent state representation in the Wilson Cowan model. To account for noise in the BNM model, we chose to define the output of the LSTM as two variables: one representing the mean and the other the standard deviation. We then sample from this gaussian distribution in order to estimate our initial conditions. This is similar to the

sampling done in Variational Autoencoders and is also used in the Neural ODE architecture (Chen et al., 2018). By representing the mapping as a non-deterministic process with some variation due to the sampling, the algorithm generalizes to perform better on test data sets and gives more robust results between instantiations.

The output of the BNM is taken to be the next fMRI predicted timestep. The loss function is taken as the difference between the predicted and the empirical next time points and the Autoencoder is trained based on this gradient. By forcing the output of the BNM to be the next predicted fMRI signal, the output of the LSTM is forced to become the closest initial time point and the LSTM solves for the non-linear transformation.

Figure 4.2 Brain Network Model



The Brain Network Model state space is constructed by averaging together the time courses of each parcellated region. The change of one of those areas x_i's a function of its own activity and its neighbors activity that it is connected withρ_{ij}, and the projection of external cortical input u_k to the brain via π_{ik}.

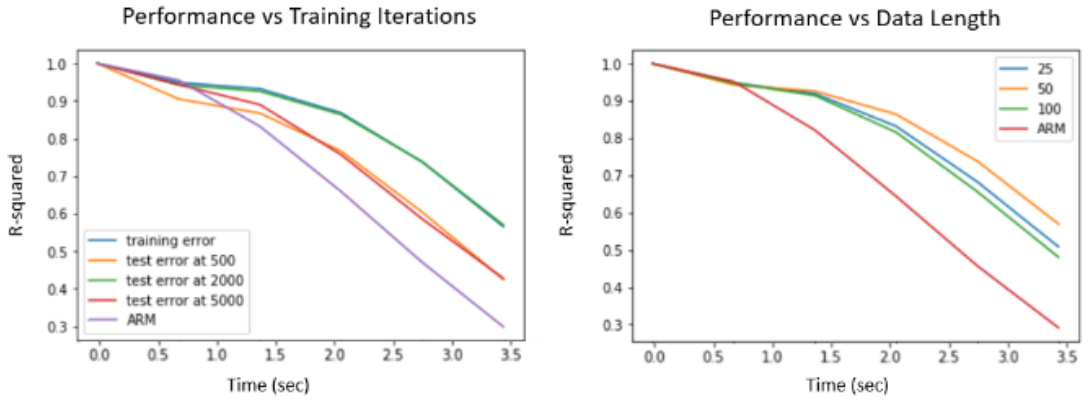
4.2.3 Long Short-Term Memory and Implementation

The LSTM units take in a series of consecutive time points and for each timepoint outputs the corresponding initial condition (see Figure 4.1b). The LSTM units are a form

of recurrent neural networks and have memory of previous time points by using a hidden state vector which it uses as an input to itself for the next consecutive time point (Figure 1B). Hence, LSTMs have become popular in the Machine Learning community because of their success in using this architecture in modeling time series such as speech and natural language processing, in self driving cars, and even in neural Turing computers thought to emulate biological intelligence (Graves, Mohamed, & Hinton, 2013; Graves & Schmidhuber, 2009; Graves, Wayne, & Danihelka, 2014). Moreover, they solve the problem of learning structure across infinite sequences of consecutive time points by using a forget gate to truncate inputs seen from a long time ago. In practice this means that they need to be trained with a finite sequence length of data.

The preprocessed data (Methods Section 4.6.2) is first cut into contiguous segments of length k. This whole segment is then passed into the Long Short Term Memory unit as shown in Fig 1B. The units are built using the Tensor Flow API, specifically the GPU boosted version to improve speed and performance.

Figure 4.3 Tuning Parameters



Left: The effect of over and undertraining the network. The performance on the test data compared to the training data at 500 and 5000 is much worse. For our network size it performed best at around 2000 iterations. It is compared to the

Autoregressive model (ARM) as baseline Right: The effect of picking different length segments and the performance accuracy. Again, the maximum is closer to the middle which was in our case 50. Too small or too large networks perform slightly worse in terms of predicting future rs-fMRI timepoints.

For our implementation, we tested data of length 25, 50 and 100 time points (18, 36, 72 sec) as seen in Figure 4.3 left, and used the accuracy of future predictions as a metric to choose the best time length to train on. The model performed best on 50 length segments, and slightly worse for shorter and longer segments. The LSTM network was also stacked into several layers in a manner similar to the way that convolutional neural networks are stacked together in a series. We used 5 identical layers to model the fMRI timeseries. We tested with more layers (up to 7) which didn't improve accuracy and took a lot longer to learn. Fewer layers also considerably reduced the time of training the network but around 3 layers the accuracy was dropping considerably. Using the inference error as a metric we also swept the number of training iterations until the performance on unseen crossvalidated testing data was about the same as the training data as shown in Figure 4.3 right. For the crossvalidation we split the data of 447 individual scans for 40 test and 407 training samples randomly. At the right amount of training steps, the system does relatively equally in test and training sets. An overtrained or undertrained network on the other hand, resulted in large differences in test and training, although all three models do equally well on the training dataset. To additionally control for overfitting, we also used the inbuilt tensor flow Dropout function that prunes a large number of the weaker weights used in the LSTMs. This has been shown in neural networks to better generalize to unseen test data (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014).

To speed up the training process, we utilized mini batches, where multiple instances of the training data are used simultaneously to train the network (Ioffe & Szegedy, 2015). The number of instances that the network can be trained on simultaneously, depends on the size of the training data, and with 400 subjects we used 60 instances to simultaneously train the algorithm. The LSTM network in our model is initialized to a random point, and the first segment is passed through the algorithm to determine the initial state for the subsequent segment. The performance on the very first block is very poor due to the unknown hidden state and is not included in our evaluation of the algorithm in the results section. This is a limitation with our implementation, and more complex architectures that solve for the initial state might circumvent this problem.

We used the Adam optimizer for the gradient descent with a learning rate of 0.0001. Smaller values took longer to train, and larger values ended up not converging.

4.2.4 Experimental Data

4.2.4.1 Structural Network

To estimate the structural network we ran tractography on 5 HCP Diffusion Weighted Images using the freely available software Mrtrix (Kashyap & Keilholz, 2019; Van Essen et al., 2013). From the tractography we estimated the number of fibers that intersected two ROIs in the Desikan-Killiany atlas and normalized the power by dividing by the surface area of the receiving region (Cabral et al., 2011; Desikan et al., 2006). The matrix is finally normalized by dividing by the largest eigenvalue such that the graph Laplacian (k*SN-I) has only negative eigenvalues (Cabral et al., 2011). This normalizes the dynamics so that the feedback decays over time, and does not exponentially increase

the signal over time. The value of k is a hyperparameter and modulates the strength of the network. We chose 0.9 for all implementations which is in line with previous work (Cabral 2011). We also tested a few values around 0.9 and showed that they had little effect since the LSTM would just change the mapping. For the Wilson Cowan we set both the network coupling parameters k to 0.9 as well but learned the other parameters (Methods Section 1.2).

4.2.4.2 fMRI Data

We acquired the training fMRI data from the 447 subjects from the Human Connectome Project (HCP) (Van Essen et al., 2013). We took the minimally processed scans (MSMAII) that were registered to standard space and in CIFTI format and ica-denoised them utilizing the 300 melodic ICA vectors that are provided from HCP. We transform from the surface-voxel time series to the ROI time series by averaging all voxels according to the parcellations established by the Desikan-Killiany atlas. This was done on an individual level since the surface parcellations are provided to by HCP and Freesurfer for each individual subject (aparc and aprac2009 files). The signal is then band passed filtered from 0.0008 Hz to 0.125 Hz and then the global signal regressed using a general linear model with the mean time course of all cortical parcellations. The final signal is subsequently z-scored (Kashyap & Keilholz, 2019). For the task data, each dataset was processed separately (language, working memory, motor, social, emotional, gambling, relational) and then concatenated together. Each task dataset was rounded to the closest multiple of 50 and the autoencoder fed alternating segments of task and the rest data. We trained our algorithm using task data scans, because the algorithm was able to able to perform better on our short and long term metrics when trained with more varied data such

as task. Moreover, it is thought that even during specified tasks, the resting state networks dominate most of the activity (Smith et al., 2009). However, during evaluation we have only shown our results on predicting future resting state fMRI, and are planning to address task in our future work.

4.2.5 Dynamical analysis techniques

The BNA timeseries were first filtered (0.008 - 0.125Hz) before analyzing the properties using dynamical analysis techniques. The dynamical analysis techniques such as QPP and the K-Means analysis are described in detail in Chapter 2, that outlines metrics in order to compare the simulated whole brain signal and the rs-fMRI signals (Kashyap & Keilholz, 2019). The QPP algorithm randomly picks a twenty second, segment of data and correlates it with the whole signal. At the regions of peak correlation, the algorithm sums up all segments and creates a new template and iteratively converges to a repeating pattern (Majeed et al., 2011). The K-means analysis takes in sliding windowed (36 sec) functional connectivity matrices that are Fisher transformed and clusters them into seven different clusters (Allen et al., 2012). We used an L1 distance to calculate the distance between matrices (Allen et al., 2012). The resulting transitions between clusters were then quantified.

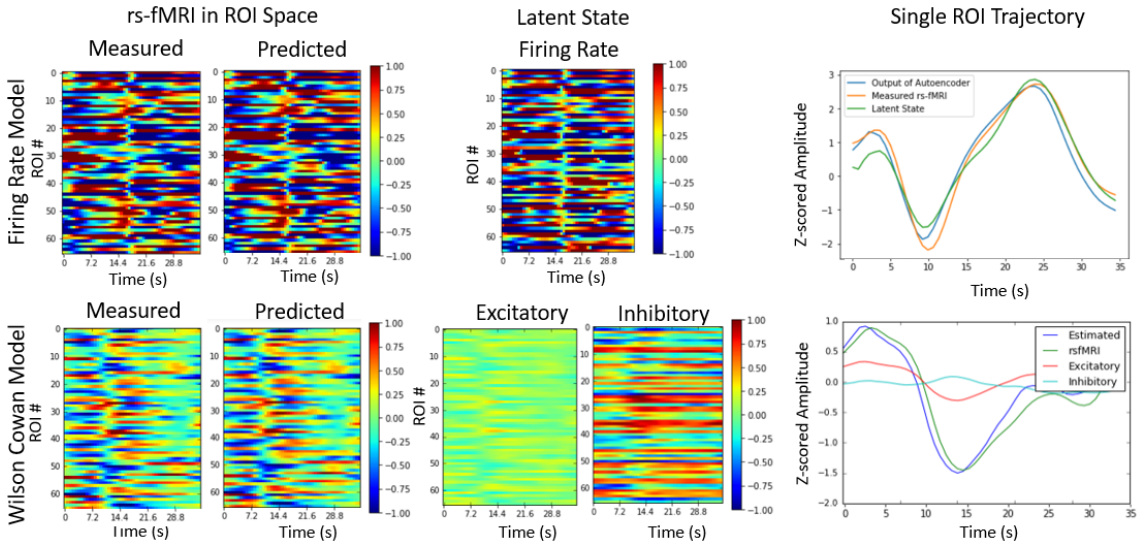
4.3 Results

4.3.1 One Time Step prediction

The sequential Autoencoder is trained to predict one time step in advance. In this section we show how the signal is reproduced across 66 regions starting from the input,

then projected onto the latent space representing the initial conditions of the BNM and finally are integrated to predict next time step. In Figure 4.4, we present the results of predicting the next time-step from the previous time-step for the two different variants of BNA, the Firing Rate Model and the Wilson-Cowan model. Although both are able to reproduce the spatial temporal signal as shown in Figure 4.4 (middle top) and Figure 4.4 (middle bottom), they differ in the latent or hidden variables used to represent the transitions. For the Firing Rate BNA, the measured data is projected into a space with firing rate as the hidden variable for each region, as can be seen in Figure 4.4 (right top). The latent variable time series has a high degree of similarity to the original signal (correlation >0.9) shown in Figure 4.4 top right. The latent state is then passed onto the BNM which integrates it according to the Firing Rate model to predict the next time-step. The traces of the input, output and latent state for a single ROI are shown in Figure 4.4 (left top). For the Wilson-Cowan model Figure 4.4 (bottom row), the latent state is represented by two variables, the excitatory and the inhibitory currents, and their interaction through the Wilson-Cowan model produces the next rs-fMRI time step. The excitatory current is positively correlated with the measured signal and the inhibitory current is negatively correlated with the signal although both to lesser degrees than the Firing Rate model. The models perform relatively similarly in predicting one time step in front and are able to reproduce the input signal with an r-squared of 0.95 averaged across all areas.

Figure 4.4 One Time Step Prediction



Two different Brain Network Autoencoders are able to reproduce the next time step from the previous time step. The Autoencoder takes as input the measured signal (left most) at time step t and outputs the predicted (second from Left) signal at t+1. The Autoencoder projects the input into a space constrained by the Brain Network Model Equations (middle panel) which are represented as the state variables in the Firing Rate model (Firing Rate) or the Wilson Cowan model (Excitatory and Inhibitory Currents), and then are integrated to produce the predicted output. The plots represent a 3D contour plot where the x axis represents time, and y axis represents the different 66 ROIs and the color represents the intensity which has been z-scored (see preprocessing). The right most panel shows the timeseries of a single ROI for the input (rs-fMRI), output (Estimated) and latent state (Firing rate or Excitatory/Inhibitory currents). At one time step the accuracy in terms of r-squared across all ROIs is on average 0.95.

4.3.2 Multiple Time Step prediction

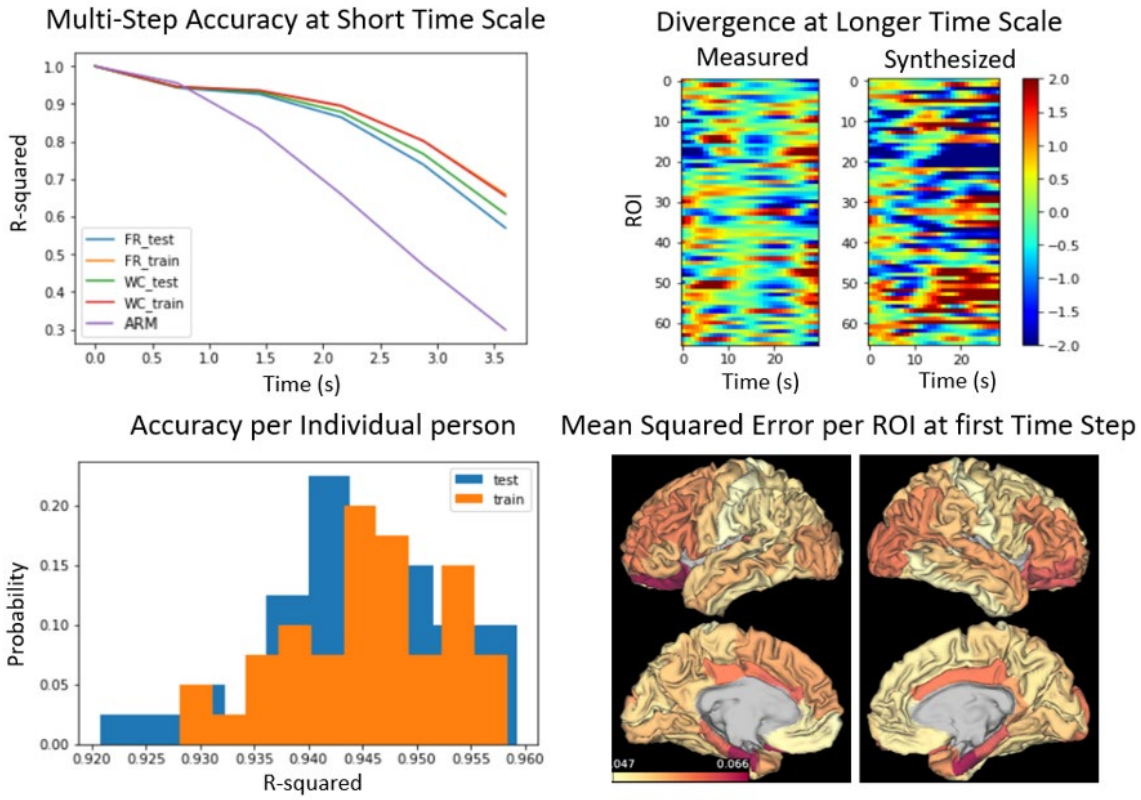
The sequential Autoencoder architecture can also predict multiple steps into the future by recursively feeding the predicted output in as the next input. The performance of multi time step forecasting is shown in Figure 4.5 (top left), where the averaged r-squared across a test and a subset of the training data of the same size for both BNA variants are compared with a naive variant of the Autoregressive model (ARM) that assumes the next time point is the previous time point (see methods). The ARM is similar to the current

approach used to differentiate task from rest signals, namely the generalized linear model (GLM), which uses the time steps before task activation as a regressor to remove the resting state activity from task responses (Smith et al., 2009). The GLM and the ARM model resting state as a constant baseline that does not change over time. Although the ARM performs as well as the BNA for the first time point, the BNA is able to reproduce the first three time steps with an r-squared of around 0.9 or higher as opposed to ARM model which is only greater than 0.9 for the first time step. The test and training performance are relatively similar for the Autoencoders, only when all the parameters are set correctly and the network is not over or under trained (see Methods Figure 4.2 for more detail).

Characteristic of Autoencoders, the error compounds at every timestep, because the previous errors are propagated to the next time step. This causes the model to completely diverge around 10 seconds from the measured signal as shown in Figure 4.5 top right. the bottom left panel in Figure 4.5 also shows that the BNA generalizes across individuals, as the histogram of the errors is roughly the same for all the individuals in the training or the testing data set. The two different BNA variants, the Firing Rate and the Wilson Cowan are similar in performance as seen in top right of Figure 4.5, with the Wilson Cowan having on average a higher r-squared on the test data set. The BNA does not perform equally in predicting each of the ROI timeseries. It predicts certain regions with a higher accuracy than the others. Th mean squared error per each ROI for the first time step is shown in Figure 4.5 bottom right. The Mean squared error was used here instead of r-squared, because the network was trained to minimize this gradient during training and most accurately represents the performance on each ROI. The error was largest in the ROIs in the temporal lobe, namely the entorhinal cortex, parahippocampal gyrus and the temporal

pole. These regions are the least connected to the rest of the network and more connected with subcortical regions, which have not been included in the simulations (Cabral et al., 2011).

Figure 4.5 Multiple Time Step Prediction

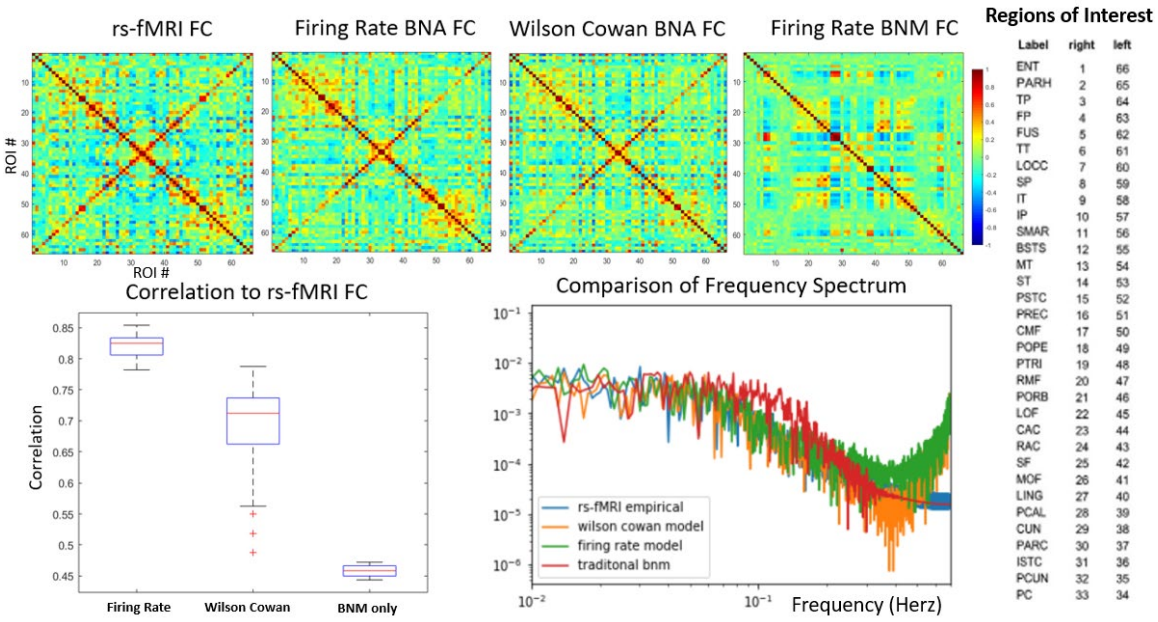


Top Left: Accuracy of our generative model in synthesizing the first few time points. The accuracy of the Firing Rate and Wilson Cowan model are compared on training and test data sets and to the Autoregressive model (ARM). The error compounds and gradually increases until the model diverges completely from the measured signal around 10 seconds and continues along its own dynamics (top right). The accuracy over time for the Wilson Cowan and Firing Rate training overlap as both models do about as well as each other on training data. Bottom left: Histogram of r-squared for each individual in test and train data sets shows that it generalizes across individuals. Bottom right: The mean squared error (MSE) for each region of interest (ROI) in predicting the first time step. The MSE is used here to compare differences across ROIs, because it was the error that was used to train the system and is more reproducible across instantiations.

4.3.3 Analysis of Long Simulations

In order to assess properties of the simulated signal at longer periods, the BNA with the Euler integration was used to generate 1000 time points or 12 minutes of data. Properties of longer simulations of BNA were compared to those of the empirical signal. In Figure 4.6, the average Functional Connectivity and the Power spectrum of the empirical and the BNA as well as a traditional Firing Rate BNM is compared. The traditional BNM FC has a weak correlation with the empirical FC (0.35) and is in the range of most traditional methods (0.3-0.6) (Cabral et al., 2017; Kashyap & Keilholz, 2019; Sanz-Leon et al., 2015; Senden, Reuter, van den Heuvel, Goebel, & Deco, 2017). The BNA performs much better at reproducing the detailed relationship between ROIs seen in FC, compared to the traditional model where groups of ROIs are synchronized over long periods of time causing square patches in the FC due to the ordering of ROIs by highly connected subgraphs (Cabral et al., 2011; Kashyap & Keilholz, 2019). The FC of the BNA has a high correlation of 0.83 (Firing Rate) and 0.7 (Wilson Cowan) to the actual measured signal. The spectral power of the empirical and the simulated signal are indistinguishable in the range 0.01 to the 0.125 Hz, and has the characteristic 1/f linear slope of around 0.9. The traditional BNM has less temporal structure and is relatively flat over the lower frequency compared to all the other models and the empirical signal (Kashyap & Keilholz, 2019). At higher frequencies the model tends to produce much higher levels of noise than the empirical signal and the traditional BNM both which have already been filtered in the pre-processing steps. Before analyzing the simulated signal with dynamical analysis techniques, we therefore filtered it at 0.125 Hz to minimize the high frequency power that would interfere with the dynamic analysis algorithms.

Figure 4.6 Inference FC and Spectrum Comparison



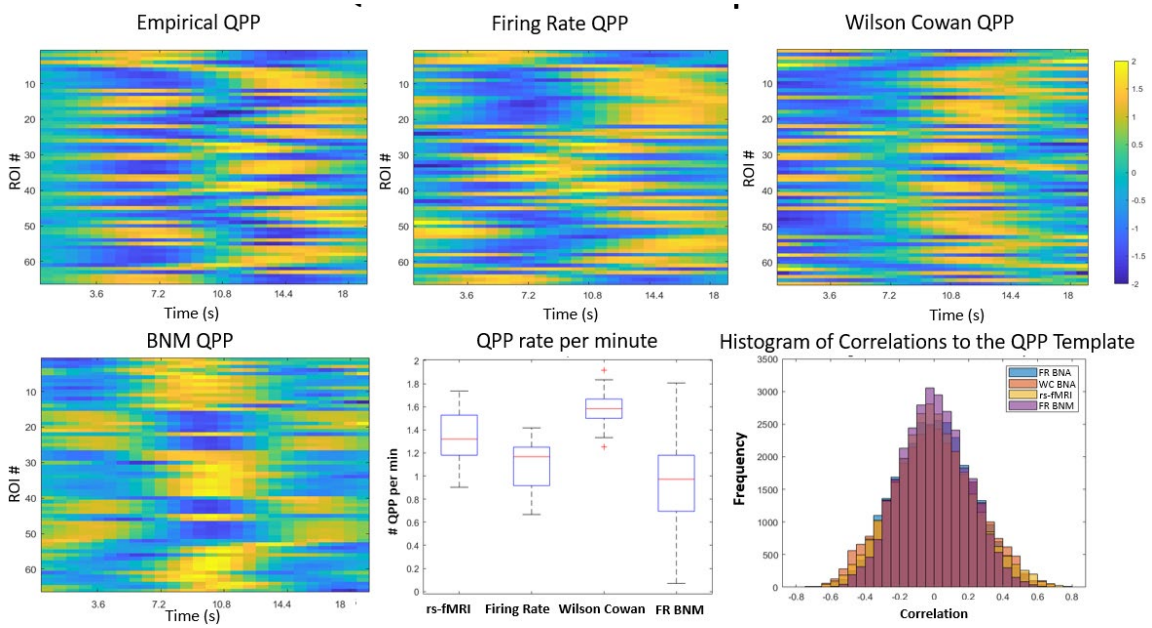
Comparison of average functional connectivity from empirical rs-fMRI (top left), BNA Firing Rate (top right middle), BNA Wilson Cowan (top left middle) and a traditional Firing Rate BNM (top left). The simulated FC matrices have a high degree of correlation 0.7-0.85 with the empirical FC unlike the traditional BNM which have a correlation of 0.5 (bottom left). Each axis in the FC plots represent the regions in the ROI which are shown on the right. The frequency spectrum (bottom right) of the BNA follows that of the empirical signal exactly except at the higher frequencies (>0.125 Hz) where the simulated signal has much larger power. The traditional BNM has less structure in the frequency range (0.01 - 0.1Hz) and has equal power in most of the range compared to the rs-fMRI and the BNA models. The traditional BNM and the empirical signal also have been filtered at 0.125 Hz while the BNA models are not.

4.3.4 Quasi Periodic Pattern Analysis of Simulated Data

We also analysed the simulated signal for unique trajectories known as quasi periodic patterns (QPP), which could also be considered a limit cycle (Majeed et al., 2011). Limit cycles are a property unique to non-linear systems, and reproducing such a property would mean that the generative model reproduces some of the dynamics features of rs-fMRI despite its divergence from measured signals. In Figure 4.7 we have plotted the QPP pattern for the rs-fMRI signal (top left), a traditional generative Firing Rate BNM model

(bottom left) and both the BNA variants (top middle and right). The empirical QPP pattern involves a twenty second trajectory that switches from task positive networks (first half of the template) to the more internal or default mode networks of the brain (second half of the template) (Majeed et al., 2011). After phase adjusting the templates, the maximum correlation of the Firing Rate BNA QPP was 0.75 and the Wilson Cowan BNA QPP was 0.43 to the original template. This is very different than the traditional dynamics seen in BNMA (bottom left) which produce blocky limit cycles, of clusters of nodes that are highly synchronized together and activating together. The BNA produces QPP that are highly structured spatially and temporally. The correlation between the QPP template and the signal is plotted in the bottom middle, where certain time points show high degree of correlation to the trajectory in the QPP template. Thresholding at 95 percent significance, the occurrence of these QPP patterns is around 1.3 times a minute in the rs-fMRI data. The BNA models have similar rates, where the Firing Rate BNA has a occurrence of 1.19 times a minute. The Firing Rate BNM model shows more variance in the number of QPP cycles per minute (bottom middle Figure 4.7).

Figure 4.7 Quasi Periodic Pattern Comparison



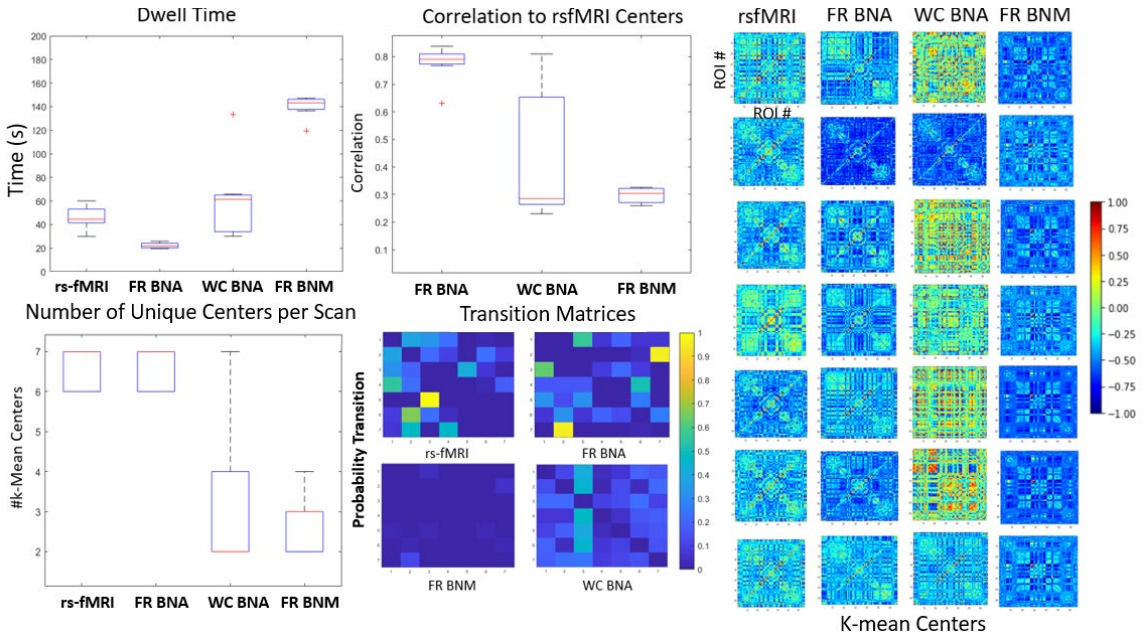
Comparison of the different QPP Templates is shown in the top row between measured data (top left), the Brain Network Autoencoders (BNA) (top middle and right), and the older BNM (bottom left). The QPP template represent a unique 18 sec trajectory of all the ROIs (y axis) that repeats itself on average 1.3 times per minute (bottom middle). The rs-fMRI signal is highly correlated with the template for during specific time-points in its trajectory as seen in the distribution of correlations to the template (bottom right). The Wilson Cowan and the Firing Rate BNA have similar distributions, while the BNM template is least correlated with its own data. The Firing Rate BNA QPP is the closest to the empirical QPP (correlation 0.73) and occurs roughly 1.19 times per minute. The Wilson Cowan BNA QPP occurs a little faster around 1.4 times a minute and has a correlation of 0.43 with the original template. The older BNM QPP is more of an on-off trajectory and does not have the intricate delays and temporal structure as seen in the QPP of the empirical signal or the BNA models.

4.3.5 K-means analysis of Simulated Data

Another property of rs-fMRI that has been studied is the existence of brain states, which can be described as large scale patterns of functional organization that are stable over the span on the order of around 40 seconds (Allen et al., 2012; Liu & Duyn, 2013).The brain transitions through these states over time (Allen et al., 2012). Algorithms such as k-

Means have typically identified 6-7 states. We applied k-Means clustering on short windowed functional connectivity matrices (50 sec) to find these states in the simulated data (see methods for more detail). In Figure 4.8, we show the comparison between our BNA models, the Firing Rate BNM models and the measured signal for cluster centers as a result of the k-means algorithm. We quantified how close the centers are to each other, by taking the maximum correlation of each center to those measured in rs-fMRI. We calculated the length of time in each state (top left), the transition likelihood between states (bottom middle) and how many unique states were observed in a single scan (bottom left). The centers of the BNA models (middle two) compared to the traditional BNM (right most) are much more distinct from each other. The Firing Rate BNA model has the highest correlation with the rs-fMRI states (0.8 on average) and a similar number of states seen during a single scan. However, the Firing Rate BNA has a shorter dwell time and seems to move between states faster than observed in the measured signal. The Wilson Cowan has more variable and diverse centers and tends to have fewer of them in a single scan, but tends to dwell in them around as long as the measured data. The traditional Firing Rate BNM model is the least accurate, has few transitions between states, and dwells in a single state for a very long time.

Figure 4.8 Comparison of K-means Centroids



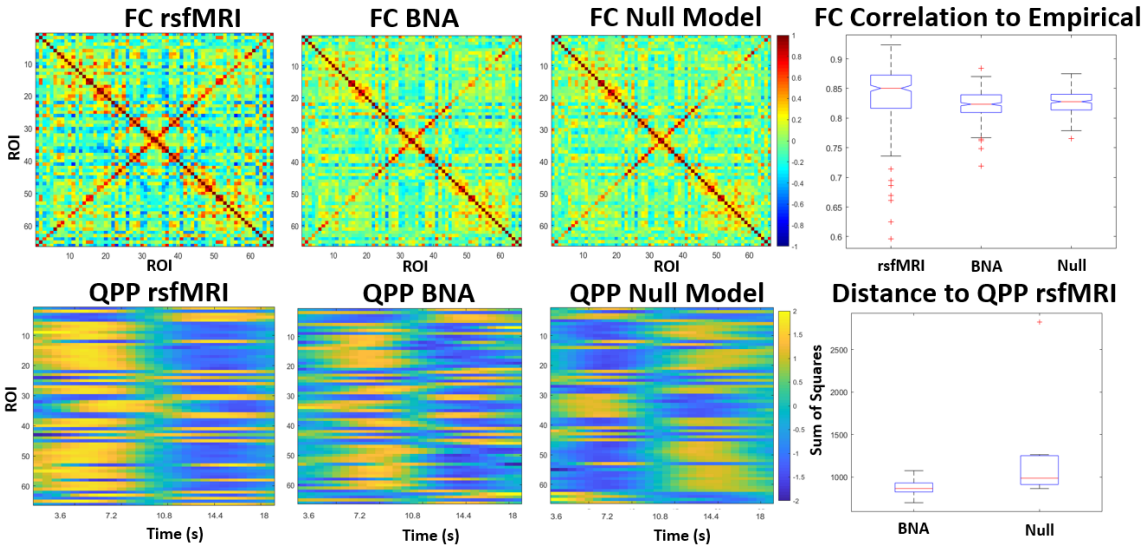
This figure compares the k-means centers and the transitions for the simulated (BNA and naive BNM) and the empirical signal (30 scans of 15min). The seven centers are shown in the far right for each category (FR: Firing Rate, WC: Wilson Cowan). A boxplot measuring the max correlation is shown (middle top) between each of simulated centers to the centers from the rs-fMRI data. The dwell time in seconds in each of these centers is shown top left. The rest of the transition probabilities (diagonal zeroed out) are shown bottom middle. The number of centers in each of the 30 scans is also variable even though they all are defined to have seven clusters across all scans (bottom left).

4.3.6 Comparison to the Machine Learning Null Model in terms of Long Term Metrics

To test whether the structural network improves the performance of the BNA under long term measures, we show the results of a Neural ODE using the Firing Rate model vs a null model consisting only of the RNN. In Figure 4.9, we show that the FC distribution from 3600 simulated runs at 10 min. Both models show a high degree of correlation to the ground truth rs-fMRI and are relatively indistinguishable. However, when we test the generalizability of the system in producing more complex trajectories such as quasi periodic patterns there is a larger difference between the BNA and the null model. Figure

4.9 bottom right shows that the sum of squares of the difference between 10 QPP patterns extracted from simulated data (30 scans each 10 min) and the QPP template extracted from the rs-fMRI data. The BNA with the correct structural matrix simulates data that has QPP templates that are more closely related to those seen using the identity matrix, and therefore suggesting that the BNM does indeed lead to solutions that are more representative of measured rs-fMRI signals.

Figure 4.9 Difference between Null Model and BNA Long Term Predictions



Top row: Comparison of FC between the measured rs-fMRI and two of the simulated firing rate models, one utilizing the BNM and the other one consisting of a RNN standalone system. The two models perform relatively similarly and have a high median correlation to rs-fMRI. Individual scan FC compared to group FC have a similar distribution compared to the model. However, for more difficult measure of reproducing repeating quasi periodic patterns, the RNN in conjunction with a BNM outperforms the RNN only null model and has a lower sum of squares error to the empirical template which is significantly lower.

4.4 Discussion

In this manuscript, we adapted the brain network model with the recurrent neural networks in order to make short-time future predictions from observed rs-fMRI. Using this

approach, we showed that using the previous measured rs-fMRI data point and an RNN in conjunction with a network-based model, we can predict large amounts of variance in the subsequent time step. We then showed that this system generalizes and can generate trajectories that are similar to resting-state trajectories over larger timescales.

4.4.1 Predicting Moment to Moment Variations

We showed that a network-based model can account for up to 95% of the variance in the fMRI signal between two adjacent time points. This reproduction is not unique, however, and can be estimated using any number of latent variables. Although more complex architectures such as variational autoencoder might be able to successfully predict future rs-fMRI data (Pandarinath et al., 2018), the BNM provides an adequate rough guess of the system dynamics for the autoencoder to converge. This information helps the model to converge during training and make accurate predictions. Moreover, unlike a traditional machine learning approach, this approach yields testable latent variables that can be further evaluated using multimodal datasets, such as magnetoencephalography (MEG) recordings that have been used to generate excitatory and inhibitory currents synchronized with concurrent rs-fMRI recordings (Ritter, Schirner, McIntosh, & Jirsa, 2013).

Fluctuations in spontaneous whole-brain activity have been shown to be nonrandom and highly structured (Zalesky et al., 2014). This suggests that rs-fMRI has both deterministic and stochastic components. The variance explained by the BNA at one time prediction represents a lower bound of the amount of determinism that exists in the signal. It is not surprising that this is the major component of rs-fMRI since the signal has been shown to be highly autocorrelated with itself (Arbabshirani et al., 2014). The

simplified first-order autoregressive model, which assumes a steady baseline at the last measured time step, has similar results in performance to the BNA when compared with a single time step and has an R 2 of 0.97. However, for multiple time steps into the future the autoregressive model performs poorly, compared with the BNA models. The two different BNA models perform at short-term scales about as well as each other. This suggests that the trajectory in the short time span is predictable to a certain degree regardless of the approach, but thereafter it starts diverging from the empirical measurements. The divergence from the original trajectory could be due to a number of sources, such as unknown task or stimulus information, noise, not incorporating higher order terms in the BNM, the fallacy of assuming that each ROI behaves in a homogenous fashion, or simply a mismatch between the algorithm and the data that increases over time. Note the BNA itself is not a deterministic system. The latent space variables are modeled as distributions before they are sampled, resulting in a stochastic system.

4.4.2 Evaluations on Long Term Dynamics

Although both rs-fMRI and the BNA models are stochastic, long-term simulations of the network-based model are able to reproduce trajectories that are similar to those seen in rs-fMRI. Individual trajectories are varied but they repeat over time, suggesting that rs-fMRI follows a bounded stable manifold that the model is able to estimate. Therefore, random walks across this manifold have shared properties in both the model and the empirical signal. Our results also suggest that most of the resting-state manifold is strongly related to the network-based activity rather than input or random perturbations from noise sources such as higher neural processing.

The strongest metric demonstrating this relationship is average FC, which has a large correlation to the empirical dynamics (0.9 > correlation > 0.8). This is unsurprising since the traditional BNMs do almost as well as the BNAs in this metric, and correlations as high as 0.7 have been reported in the literature (Senden et al., 2017). Average FC seems to be more related to the structural input than the description of the dynamical system (Cabral et al., 2017; Kashyap & Keilholz, 2019). However, the BNA does better than most BNMs in estimating interhemispheric FC correctly, which is usually challenging in network-based models because there are far fewer interhemispheric than intrahemispheric connections detected with diffusion MRI. The power spectrum profile is also mostly reproducible by the model, except in the very high frequency where the model has a lot more power than the empirical signal. This might occur because of the lack of friction in our model, namely that the signals are constantly propagated through feedback loops in the network without loss of energy, unlike the real system. Since most of predictability of the resting state comes from the structured low-frequency activity, we can filter synthesized signal without losing too much information. Other traditional brain network models using the virtual brain have also reported similar performance on power spectrum profiles (Ritter et al., 2013).

Although most traditional BNMs have been able to reproduce to some degree the long-term-averaged properties such as average FC and power spectrum, they have had a harder time in reproducing faster scale dynamics such as reoccurring unique trajectories or the multistate transitions seen in dynamic FC (Cabral et al., 2017; Hansen, Battaglia, Spiegler, Deco, & Jirsa, 2015; Kashyap & Keilholz, 2019). The results from the QPP analysis, which extracts limit cycles, show that the simulated signal has a similar 20-s

trajectory and that pattern is repeated over the course of minutes. The results from the k-means analysis on time-varying FC matrices show that the simulated signal has similar state transition in terms of both number and the spatial patterns to those seen in empirical rs-fMRI. This suggests that both of these properties arise naturally in the correct nonlinear network-based representation of rs-fMRI that can be inferred from the data using machine learning techniques. The firing rate BNA seems to fit the data better than the Wilson-Cowan BNA. This might be because the Wilson-Cowan BNA has additional nonlinearities due to the interaction between the excitatory and inhibitory currents.

A direct comparison between our model and other brain network models in the literature on complex dynamical metrics is difficult because most brain network models use their own unique metric to compare against rs-fMRI and there is no established standard. The origin of these complex dynamics has been explained in different theoretical ways. These complex transitions can arise because of the particular nonlinearities of the system (Hansen et al., 2015), which can result in multiple attractors and limit cycles naturally. They can result from parameter changes to the network strength or Hopf bifurcations that cause the system to change its dynamics over time (Deco et al., 2018; Senden et al., 2017). They can also be the result of adding external input and stimuli into the system, causing a change from the zero-input manifold and altering the dynamics (Ashourvan et al., 2019; Deco et al., 2019). These are not mutually exclusive and could induce the changes at once. Our implementation is closest to the first interpretation of rs-fMRI. We explain the observed nonlinear properties of the data purely based on network propagation without the need for external input or a change of a bifurcation variable.

4.4.3 Errors across different ROIs

Using our approach, it is not possible to tease apart the origin of the error that could arise because of a mismatch between the model and the empirical data or because of intrinsic noise. However, looking at the error across regions shows that the error is not evenly distributed across all regions of interest, which can give some clues to where it might arise. The error in reproducing the dynamics at one time step is highest in the nodes of the limbic system (Figure 4.4, bottom). We believe that our model performs less accurately in this system because they are highly connected to the amygdala and the hippocampus, which are not simulated in the model, and are the least connected nodes to the rest of the network (Cabral et al., 2011). Moreover, tractography has also been known to underestimate the uncinate fasciculus, the major highway between the temporal lobe and the frontal areas, which forms the backbone of the limbic system. The fiber has a very sharp angle that is hard to follow using tractography (Thomas et al., 2014). The echo planar imaging (EPI) sequence used to obtain rs-fMRI data has also known susceptibility issues at interfaces, which would affect the nodes at the proximity such as the frontal pole and the temporal pole, both of which have larger mean squared error compared with the other nodes.

4.4.4 Comparison to other Machine Learning and Time Forecasting models

Similar time forecasting has been attempted or is being attempted by several different labs at the time of this manuscript. A variant utilizes a variational autoencoder to find a latent space of brain trajectories that would fit the current data (Brown, Pasquini, Lee, & Seeley, 2019). Another RNN-ICA version uses independento component analysis vectors as the latent space, while another method uses hidden Markov model to model the hidden states (Hjelm et al., 2018; Vidaurre et al., 2018). However, our method is unique in

using brain network models as a latent space, whose variables are more interpretable since they represent the state of each neural population's activity and can be tested using multimodal data. Moreover, none of the other architectures use their model for time series foresting or dynamical analysis, hence their results are not directly comparable to our work, although their methods are similar.

4.4.5 Limitations

There are many assumptions that limit the scope of our approach. Machine learning, although good at learning structures in datasets, has a shortcoming of arbitrarily creating a system to fit the data, and every instantiation of the system produces slightly different properties of the simulated system. We tried to address this issue by using various techniques such as using structural constraints, dropout of long short-term memory (LSTM) units, using probability to track the latent variables, and taking the results of multiple runs, in order to make the system more reliable and reproducible. Another limitation of this model is that it needs 50 time points prior to the data point in order to solve for the initial conditions. Shorter time intervals than 50 time points are faster to train, but are less accurate in estimating slow processes. The longer segments required a larger LSTM network and longer training times and were less accurate in our dataset. There are more complex architectures that could solve for the initial conditions faster, such as a forward-backward LSTM architecture (Pandarinath et al., 2018). On the network side, the parcellation scheme reduces the complexity of the signal and discretizes the network. Improvements can be made by allowing for continuous propagation along the cortical sheet, as in the neural field models. Tractography also has its limitations, and better estimates of structural networks should make the model more realistic and improve results

especially in regions that are not very strongly connected to the rest of the network. Simulating more of the central nervous system including subcortical regions would also lead to a more biologically plausible model.

Section 4.3.2 and 4.4.1, the comparison between BNMs on short term metrics is somewhat difficult to evaluate due to the RNN involvement in the dynamics. In order to truly differentiate between BNMs without the dynamics of the RNN, the BNM have to generate all subsequent timepoints via integration without the RNN as described in detail in Chapter 3.

4.5 Conclusion

We set out to investigate the extent to which network-based theory can explain the moment-to-moment variations seen in rs-fMRI signal's. Using a novel machine learning approach, we solve for the initial state of traditional network-based models and show that we can account for most of the variation seen in the signal and predict accurately (> 0.6 R2) for at least five consecutive time points. Longer instantiations of the system show that our model is able to produce complex trajectories of the nonlinear dynamical system on the order of minutes. We believe that our BNA will be useful when a generative model of rest is needed. Moreover, it can be trained to predict in real time, which allows contrast against dynamics that contain deviations from rest such as in task fMRI studies. In the future, it can also be used to investigate deviations from the manifold such as in task input or due to noisy sources.

CHAPTER 5.

EVALUATING OUR CONTRIBUTIONS TO FUTURE AND CURRENT APPLICATIONS OF BRAIN NETWORK MODELS

5.1 Analysis of Brain activity at the Systems Level

The strength of using Brain Network Models to model whole brain activity, is that it provides a framework for translating the activity of individual modular regions to network activity. It has been difficult to study the network activity as a whole, and the bulk of cortical studies in neuroscience have been focused on understanding specialized streams of processing. However, as our knowledge surpasses individual specialized systems, we need network representation of whole brain activity to understand how these individual components are used in conjunction to solve particular tasks and how ongoing activity in other areas affects processing of sensory information.

Some insight into how these modular components might interact on a system wide level can be gained by observing advancements in the Machine Learning Community. Artificial intelligence is moving further away from training large architectures in order to solve a specific problem, i.e., digit recognition, chess, self-driving cars. Rather they train modular components separately to compress sensory information. To solve for complex tasks, the output of the sensory regions are then connected with higher order controllers to create a platform that are more flexible in solving different tasks, analogous to training a model to play video games rather than training it to play a specific one (Mnih et al., 2015, Ha & Schmidhuber 2018). Moreover, access to the hidden states processing that arises

naturally in a modular network like architecture, has allowed for AI to train in their own internal representation of the world (Ha & Schmidhuber 2018). Thus, AI is slowly moving further away from a robotic input output relationship, to making predictions based on their own changing representation of the outside world, a notion usually that is associated with human intelligence.

Thus, understanding this network element that is constantly interacting with the information processing and the outputs of the system, is key in understanding how our intelligent might arise. Therefore, it is important to represent whole brain data in terms of a network activity where the activity of modular components have physical interpretations, and the change in activity can be related as a function of the network they are embedded in.

5.2 Gaps in developing better Brain Network Models

In the previous chapters, we have introduced some of the major gaps on translating a mathematical Brain Network Model to fit rs-fMRI data. Resting state fMRI does not have a clearly defined temporal event that would allow us to initialize our model with the data and thus have made it difficult to compare. Therefore researchers, relied on long time averaging properties such as functional connectivity to compare between models and data. However, this was an issue as functional connectivity has been shown to have problems in characterizing fast dynamics that are present in rs-fMRI. Moreover, some of these faster spatial temporal dynamics are characteristic of rs-fMRI and should be present in a generative model for rs-fMRI.

Therefore, in order to develop better Brain Network Models we addressed the following gaps: 1) the lack of sensitive metrics to distinguish the performance of different models/parameterizations (Chapter 2), 2) no temporal alignment between rs-fMRI and models, making it impossible to directly compare short term dynamics (Chapter 3), 3) inability of existing models to generate rs-fMRI data that displays realistic complex spatial temporal dynamics (Chapter 4).

Significant gaps remain for Brain Network Models to become a practical tool for clinical and behavioral studies. Although BNMs can be built using individual DWI data, they are not yet sensitive enough to account for individual differences in fMRI data. These individual differences can be modeled using complex non-linear models with many parameters, but this leads to a large search space problem that has yet to be solved at the individual level. Moreover, task models for BNM have not yet been thoroughly explored. Being able to predict behavioral metrics would also give the models credence in their depiction of network activity. Without being able to solve practical problems its difficult to justify using these models in order to interpret whole brain activity. Research have also started to focus on analyzing multimodal datasets EEG, MEG and fMRI data using BNM, where some of these gaps could potentially be addressed (Deco et al., 2008, Deco et al., 2016). These combined models have tried to understand how high frequency neural activity measured by these modalities could result in the measured hemodynamic response. Gaps remain in these multimodal studies, however, such as the problem of not having an obvious manner of comparing the simulated signal to now both of the measured modalities. Current models are at the stage of demonstrating viability of these methods can explain features

across multimodal data rather than utilizing them in any real applications. However, since the use of BNMs are fairly new, these open problems may still be addressed in the future.

5.3 Results and Significance

Our approach in addressing these gaps, has been to first to use dynamical tools developed to study rs-fMRI dynamics to analyze simulated BNM data in order to establish new metrics that are more informative in differentiating between models. In Chapter 2, we show that faster and more transient metrics that characterize spatial temporal trajectories are most efficient in being able to differentiate between various BNMs. We also noted that none of the traditional BNMs were really able to produce the complex spatial temporal trajectories seen in K-mean clustering or via the QPP analysis.

In Chapter 3, we develop a method of directly comparing short term trajectories by solving for the initial conditions of the BNMs with respect of the given data. This also allows us to address the problem of synchronizing the BNMs with empirical observations. We showed that we were able to use a new tool from Machine Learning, known as Neural ODE to solve for the known initial conditions on a toy dataset. Then extending our model on fMRI data, we were able to fit a set of BNMs to the empirical measurements. We then demonstrated the effectiveness of our method in differentiating between the BNMs to determine which amongst them recapitulated rs-fMRI dynamics best, via constructed examples of comparing the dynamics of a corrupted structural matrix with the original structural matrix and the differences in dynamics between a simple non-linear and linear model.

Lastly in Chapter 4, we use the system we trained in the previous Chapter as a standalone new generative model. We explore the dynamics of the combined RNN and BNM system which has been used in the original Neural ODE paper as an inference model that might recapitulate the system it was trained to observe (Chen et al., 2018). We observe that our newer generative model is not only able to outperform the traditional BNM models, but is also able to reproduce properties of rs-fMRI that have not been reproduced before. The resulting simulated data contained complex quasi periodic patterns that were very similar to the spatial temporal trajectory observed in rs-fMRI. Moreover, analyzing the data through K-means, has also showed multiple states in the dynamics with similar spatial structure as observed in empirical data. A RNN standalone network, used as a null model, also had similar results in reproducing some of these properties. However, the combined model with the brain network information had better and less variable estimation of these complex trajectories, such as the quasi periodic pattern, when compared to the empirical pattern using the mean squared error.

Although the methods in Chapter 3 and 4 are nearly identical, they are used for very different purposes that are both relevant to the general Brain Network Model research framework. The methods in Chapter 3 represent a tool in order to evaluate the BNM on short term trajectories and after the initial conditions are predicted and the final trajectories are an integral of the given dynamical system. In Chapter 4, we use the same system in a different manner by utilizing the trained RNN network to help generate the next timepoint at every timestep. We lose some of the interpretability of the generative model compared to the methods in Chapter 3, but by utilizing the RNN we show that we are able to reproduce complex rs-fMRI trajectories that have not been simulated by any other model.

This approach can be utilized by training separate RNN models for rs-fMRI data in healthy and clinically relevant cohorts, and comparing them on how they would evolve when synchronized to the same data observations.

5.4 Relationship to existing Literature

Brain Network Models were hypothesized as a model to represent rs-fMRI dynamics in the late 2000s (Honey 2008, Deco 2009), but it wasn't until Cabral et al. 2011, that they were compared to measured rs-fMRI data using functional connectivity. Since then many different models with completely different dynamical systems have been shown to reproduce functional connectivity to a certain degree (Cabral et al., 2012, Sanz Leon et al., 2014, Hanson et al., 2014, Ritter 2013). In a review paper, Cabral et al. 2018, remarked that functional connectivity is no longer a useful metric in gauging the validity of a BNM since so many different models seem to reproduce the metric. Therefore, research in BNM tried to reproduce more complex phenomenon observed in rs-fMRI such as state transitions or bifurcations (Hansen et al., 2014, Deco et al., 2016). However, before Kashyap & Keilholz 2019, there was no rigorous assessment on what metrics are the most informative in differentiating between models. Therefore, the work in Chapter 2 serves as a useful tool for the scientific community in establishing complex dynamical metrics that are able to differentiate the simulations from each other and compare it to empirical observations.

Brain Network Models were not intended to be used as a tool to simulate the most realistic rs-fMRI, but rather to model how different neural populations might interact in order to produce rs-fMRI. Moreover, research has focused on determining which models could be used to gain insight into clinical and behavioral data. For example, specific BNMs

have been developed to model the effect of particular diseases, in order to better simulate the underlying neuropathic condition in Parkinson's and Epilepsy (Jirsa et al., 2014, Saenger et al., 2017). Simulation software has become open source such as The Virtual Brain (TVB), intended to be used as a platform to test different BNMs and fit to healthy and clinical data. TVB also combined multimodal measurements from EEG and fMRI to build better models for individuals. Our approach in developing Machine Learning tools to synchronize BNM, is quite complementary with these efforts in developing more realistic models for individual representations of rs-fMRI, since it is a tool to fit any candidate dynamical system to the data. In fact, I am joining the Virtual Brain in order to develop a Neural ODE backend framework that it will allow researchers to choose their own BNM for a given application and make predictions synchronized to the observed data.

Lastly, our Machine Learning BNM framework resulted in a generative system that when compared to all other currently known generative models of resting state, is much closer to empirical data by every measure. Unfortunately, the exact nature of the generative model becomes less clear due to the Machine Learning component and we lose the precise interpretability that a modular standalone Brain Network Model offers. However, it is able to replicate specific processes observed in rs-fMRI such as Quasi Periodic Patterns and the transitions between states seen in a K-means analysis and reproduce very realistic rs-fMRI data. The strength of the Machine Learning approach is that as long as these differences exist in the real rs-fMRI data, separate models can be trained for different groups consisting of healthy and neuropathic rs-fMRI data. The framework allows the models to predict initial conditions for any dataset, and can be used to synchronize trajectories observed from one dataset with model predictions from the other dataset. This way we can compare the

rs-fMRI between disease models in a synchronized manner. This is currently done in task fMRI by comparing the hemodynamic responses after a certain task stimulus but has not been done in rs-fMRI since there existed no previous method of aligning the two signals.

5.5 Future Applications in modeling Parkinson's Disease

Brain Network Models are being used to model network effects of neurological and psychiatric conditions. They also represent a pathway towards individualized medicine as they are constructed from individual data. To examine the clinical potential of the Brain Network Model with the tools that we have developed, we provide a specific example in utilizing these models to model Deep Brain Stimulation in the context of Parkinson's Disease.

Deep brain simulation (DBS) has been remarkably effective in treating previously intractable neuropathologies. An invasive clinical tool, DBS provides electrical stimulation at very high frequencies (~185Hz) via electrodes placed surgically into very precise locations deep inside the brain that are known to be adversely affected by the disease of concern. Stimulation has been shown to disrupt pathological coupling between different brain regions in Parkinson disease (PD), depression, and recently in animal models for Narcolepsy (Shukla 2017, Mayberg 2008, Rogers 2018). Due to its efficacy, DBS has become a widely used surgical therapy for people with PD, but there remains a gap in understanding how local DBS is able to repair or restore long range and global connectivity problems. Since the translation of local stimulation to network activity is not very well understood, it has resulted in uncertainty in identifying possible targets for electrode placements and effective stimulation parameters that would reduce aberrant brain

dynamics in a controlled manner. Moreover, DBS is not effective in certain people and it is important to be able to identify non-responders. If a model is successfully able to simulate effect of a DBS stimulator on measured rs-fMRI activity, the model could help inform clinicians on important surgical details such as DBS location and stimulation settings.

Our tools can be extended to provide a modeling based approach to study the effects of simulated stimulators in changing the whole brain dynamics and long-range connectivity in PD based on collected structural and resting state functional magnetic resonance imaging (rs-fMRI) data. For these models to be useful in a clinical setting, the model also needs to informative on an individual level by identifying non-responders as well as predicting possible locations and estimating stimulation parameters in order to recover healthy brain dynamics.

Previous studies have modeled the local effects of DBS in PD and shown how stimulation spreads across a population of neurons activating both excitatory and inhibitory cells simultaneously (Mcintyre 2004). However, these models are not able to explain how these local changes in neural activity are able to stabilize the information flow across the network and restore normal function. Therefore, recent papers have started utilizing Brain Network Models in order to model the network activity. In their approach they trained separate network models for when the stimulus condition is turned OFF and when the condition is turned ON using rs-fMRI data. They showed that the DBS had stabilized certain network properties along the striatal thalamic cortical pathway and predicted possible other surgical locations that might produce a similar effect (Saenger et al., 17).

Their work can be extended by training a single combined ON and OFF model using a Machine Learning controller in conjunction with the our Neural ODE approach.

The first task would be to construct a PD specific Brain Network Model, by extending the cortical BNMs with a specialized model representing the basal ganglia as well as simulate the effects of PD. A recent publication has shown that a variant of a BNM can reproduce the local distinct beta oscillations seen in the basal ganglia (BG) of PD patients (Chen et al., 2017). We hypothesize that by applying the PD-model of the BG to our whole brain model and compare it to a normal BG whole brain model, we can replicate the changes seen in functional connectivity as measured in rs-fMRI in PD patients. Moreover, with the tools that we developed in Chapter 3 utilizing the Neural ODE, we can fit this model and test to see specifically if we are simulating the activity of the BG with its relation to its network neighbors by observing how the model evolves when initialized to a time sequence prior to the BG activating. After the model is created and validated, the effects of DBS in will be incorporated as a frequency term in the differential equations that modifies the activity of local neural populations. In this manner, the effects of stimulation at different locations and at different stimulation parameters can be simulated. The parameters and the locations can also be estimated using Reinforcement Learning (RL) techniques. Researchers have used RL to solve the classical non linear control problem of balancing a pole on a moving cart (Konda & Tsitsiklis 2000). Future RL has the promise being able to solve for a controller that can stabilize non-linear dynamical systems.

5.6 Neuroethics

The development of more complex Brain Network Models that can simulate brains of specific neural disorders has great potential in providing relevant clinical information to the community. However, as it becomes more specific towards the individual, it also raises certain issues relating to neural ethics. Since the nervous system represents the individual, any technology that attempts to simulate their processing in some essence is making a judgment on how the individual performs based on a particular model. For example, Neuroscience is increasingly used in court cases in order to make judgments on the individual on the basis of how the nervous system responds, especially when being used in an argument to plead insanity (Martha 2010). Therefore, as any new technology making judgements on how people think could lead to some very dangerous ethical scenarios, it is important to evaluate the impact of Brain Network Models under the lens of neural ethics.

As our ability to record directly from the Central Nervous System improves, we have been able to use increasingly sophisticated tools in order to decode neural data successfully. Whole Brain activity as measured by fMRI, especially under task protocols, have for example measured the hemodynamic response of the amygdala under different stimulus, and made inferences on how well the individual is emotionally responding (Free Solo 2018). Moreover, fMRI studies are being used as lie detectors which is then presented as evidence in court cases (Martha 2010). These neural ethical problems already exist with current fMRI technology, but BNMs with Neural ODEs would allow us to decode this information in a more precise manner which has its own consequences. The synchronizing aspect of the Neural ODE described in this thesis would make it much easier to compare how the individual's fMRI evolves to the dynamics of a group model. This leads to a problem of neuro-realism, where differences that are predicted by the model are used as a

real measure to quantify an individual's response. The algorithms therefore described could potentially be misused for these purposes.

The vulnerable individuals can be defined as those individuals that fit the group model the least and would therefore be evaluated as different relative to most other humans. This knowledge might have potential harm in further alienating groups such as autistic individuals when they are compared to other cohorts. There exists no absolute truth on what the architecture of a 'healthy' person should be, and therefore it is difficult to assess differences between a group model and an individual. BNMs have the potential to be misused by corporate interests to be representing what an architecture should be, and using these differences to make assessments on the individual's future actions such as their culpability as a criminal or their performance at a job. In other words, it could be easily herald a quasi-scientific return of Phrenology in the 21st century.

My personal recommendation is that the focus of BNM research should not compare individuals against a group model but compare individuals to themselves. A certain degree of relativity needs to be enforced, where individuals should be compared using a longitudinal study and the only interpretability comes from changing their brain activity through intervention. In this manner, BNM can be used to inform the effectiveness of a particular intervention in relation to the individual rather than being used as a ruler to quantify the individual's whole brain activity.

5.7 Conclusion

Brain Network Models represent a pathway to understand how modular components of the cortex coordinate with each other to produce macroscale network activity. They are

a promising tool as they can be constructed from individual measurements, and in the future be used to provide personalized care. Moreover, they represent some of the earliest whole brain simulations of the nervous systems, and while these simulations are currently quite naïve, they represent a definite step in the direction of realistic simulations of the human nervous system.

REFERENCES

Allen, E. A., Damaraju, E., Plis, S. M., Erhardt, E. B., Eichele, T., & Calhoun, V. D. (2014). Tracking whole-brain connectivity dynamics in the resting state. Cerebral Cortex, 24(3), 663–676. ,

Arbabshirani, M., Damaraju, E., Phlypo, R., Plis, S., Allen, E., Ma, S., . . . Calhoun, V. (2014). Impact of autocorrelation on functional connectivity.NeuroImage,102, 294-308

Ashourvan, A., Pequito, S., Bertolero, M., Kim, J., Bassett, D., & Litt, B. (2019). A dynamical systems framework to uncover the drivers of large-scale cortical activity.bioRxiv

Atasoy, S., Donnelly, I., & Pearson, J. (2016). Human brain networks function in connectome-specific harmonic waves. Nature Communications, 7(21), 10340. ,

Bassett, D. S., Nelson, B. G., Mueller, B. A., Camchong, J., & Lim, K. O. (2012). Altered resting state complexity in schizophrenia. NeuroImage, 59(3), 2196–2207. ,

Belloy, M. E., Naeyaert, M., Abbas, A., Shah, D., Vanreusel, V., van Audekerke, J., … Verhoye, M. (2018). Dynamic resting state fMRI analysis in mice reveals a set of Quasi-Periodic Patterns and illustrates their relationship with the global signal. NeuroImage, 180(Pt. B), 463–484. ,

Billings, J., Medda, A., Shakil, S., Shen, X., Kashyap, A., Chen, S., . . . Keilholz, S. (2017). Instantaneous brain dynamics mapped to a continuous state space.Neuroimage,162, 344-352.

Biswal, B., Yetkin, F., Haughton, V., & Hyde, J. (1995). Functional connectivity in the motor cortex of resting human brain using echoplanar mri.Wiley,34, 537-541

Breakspear, M. (2017). Dynamic models of large-scale brain activity. Nature Neuroscience, 20(23), 340. ,

Brodman, K. (1909). Monography on the localization of cortical areas in mammalian brains including the human brain.

Brown, J., Pasquini, L., Lee, A., & Seeley, W. (2019). Brain activity gradients dynamically interact to shape functional connectivity. Poster Presentation

Bullmore, E., Long, C., Suckling, J., Fadili, J., Calvert, G., Zelaya, F., … Brammer, M. (2000). Colored noise and computational inference in neurophysiological (fMRI) time series analysis: Resampling methods in time and wavelet domains. Human Brain Mapping, 12(2), 61–78. ,

Bullmore, E., & Sporns, O. (2009). Complex brain networks: Graph theoretical analysis of structural and functional systems. Nature Reviews Neuroscience, 10, 186. ,

Caballero, G. C., Petridou, N., Dryden, I. L., Bai, L., Francis, S. T., & Gowland, P. A. (2010). Detection and characterization of single-trial fMRI BOLD responses: Paradigm free mapping. Human Brain Mapping, 32(9), 1400–1418.

Cabral, J., Hugues, E., Kringelbach, M. L., & Deco, G. (2012). Modeling the outcome of structural disconnection on resting-state functional connectivity. NeuroImage, 62(3), 1342–1353. ,

Cabral, J., Hugues, E., Sporns, O., & Deco, G. (2011). Role of local network oscillations in resting-state functional connectivity. NeuroImage, 57(1), 130–139. ,

Cabral, J., Kringelbach, M. L., & Deco, G. (2014). Exploring the network dynamics underlying brain activity during rest. Progress in Neurobiology, 114, 102–131. ,

Cabral, J., Kringelbach, M. L., & Deco, G. (2017). Functional connectivity dynamically evolves on multiple time-scales over a static structural connectome: Models and mechanisms. NeuroImage, 160, 84–96. ,

Chen, Liu et al. "Neural Mass Models Describing Possible Origin of the Excessive Beta Oscillations Correlated with Parkinsonian State." Neural Networks, vol. 88, 2017, pp. 65–73., doi:10.1016/j.neunet.2017.01.011.

Chen, R., Rubanova, Y., Bettencourt, J., & Duvenaud, D. (2019). Neural ordinary differential equations.ArXiv

Deco, G., Cabral, J., Saenger, V. M., Boly, M., Tagliazucchi, E., Laufs, H., … Kringelbach, M. L. (2018). Perturbation of whole-brain dynamics *in silico* reveals mechanistic differences between brain states. NeuroImage, 169, 46–56. ,

Deco, G., Cruzat, J., Cabral, J., Tagliazucchi, E., Laufs, H., Logothetis, N., & Kringelbach, M. (2019). Awakening: Predicting external stimulation to force transitions between different brain states.Proceedings of the National Academy of Sciences USA

Deco, G., Kringelbach, M. L., Jirsa, V. K., & Ritter, P. (2017). The dynamics of resting fluctuations in the brain: Metastability and its dynamical cortical core. Scientific Reports, 7(1), 3095. ,

Deco, G., Jirsa, V. K., & McIntosh, A. R. (2010). Emerging concepts for the dynamical organization of resting-state activity in the brain. Nature Reviews Neuroscience, 12, 43. ,

Deco, G., Jirsa, V., McIntosh, R., Sporns, O., & Ktter, R. (2009). Key role of coupling, delay, and noise in resting brain fluctuations.Proceedings of the National Academy of Sciences USA,106, 10302-10307

Deco, G., Jirsa, V.K., Robinson, P.A., Breakspear, M. & Friston, K. The dynamic brain: from spiking neurons to neural masses and cortical fields. PLoS Comput. Biol. 4, e1000092 (2008).

Deco, G., Ponce-Alvarez, A., Mantini, D., Romani, G. L., Hagmann, P., & Corbetta, M. (2013). Resting-state functional connectivity emerges from structurally and dynamically shaped slow linear fluctuations. Journal of Neuroscience, 33(27), 11239. ,

Desikan, R. S., Ségonne, F., Fischl, B., Quinn, B. T., Dickerson, B. C., Blacker, D., … Killiany, R. J. (2006). An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. NeuroImage, 31(3), 968–980. ,

Graves, A., Mohamed, A.-r., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks.Institute of Electrical and Electronics Engineering

Graves, A., & Schmidhuber, J. (2009). Offline handwriting recognition with multidimensional recurrent neural networks. Advances in Neural Information Processing Systems, 545552

Graves, A., Wayne, G., & Danihelka, I. (2014). Neural turing machines.ArXiv.

Farah, Martha J. Neuroethics an Introduction with Readings. The MIT Press, 2010.

Fornito, A., Zalesky, A., & Breakspear, M. (2013). Graph analysis of the human connectome: Promise, progress, and pitfalls. NeuroImage, 80, 426–444. ,

Friston, K. J., Harrison, L., & Penny, W. (2003). Dynamic causal modelling. NeuroImage, 19(4), 1273–1302. ,

Ha, D. and J. Schmidhuber (2018). "World Models". In: CoRR. arXiv: 1603.10122.

Hagmann, P., Cammoun, L., Gigandet, X., Meuli, R., Honey, C. J., Wedeen, V. J., & Sporns, O. (2008). Mapping the structural core of human cerebral cortex. PLoS Biology, 6(7), e159. ,

Hansen, E. C. A., Battaglia, D., Spiegler, A., Deco, G., & Jirsa, V. K. (2015). Functional connectivity dynamics: Modeling the switching behavior of the resting state. NeuroImage, 105, 525–535. ,

Helmich, R.c., et al. "Spatial Remapping of Cortico-Striatal Connectivity in Parkinson's Disease a Resting State FMRI Study." NeuroImage, vol. 47, 2009, doi:10.1016/s1053-8119(09)71050-x.

Hjelm, D., Damaraju, E., Cho, K., Laufs, H., Plis, S., & Calhoun, V. (2018). Spatio-temporal dynamics of intrinsic networks in functional magnetic imaging data using recurrent neural networks.Frontiers in Neuroscience,12(9).

Honey, C. J., K¨otter, R., Breakspear, M., & Sporns, O. (2007). Network structure of cerebral cortex shapes functional connectivity on multiple time scales.Proceedings of the National Academy of Sciences USA,104(24), 10240–10245

Hutchison, R. M., Womelsdorf, T., Allen, E. A., Bandettini, P. A., Calhoun, V. D., Corbetta, M., … Chang, C. (2013). Dynamic functional connectivity: Promise, issues, and interpretations. NeuroImage, 80, 360–378. ,

Jirsa, Viktor K., et al. “On the Nature of Seizure Dynamics.” Brain, vol. 137, no. 8, 2014, pp. 2210–2230., doi:10.1093/brain/awu133.

Jirsa, Viktor K. “Neural Field Dynamics with Local and Global Connectivity and Time Delay.” Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 367, no. 1891, 2009, pp. 1131–1143., doi:10.1098/rsta.2008.0260.

Joffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv.

Karahanoğlu, F. I., & Van De Ville, D. (2015). Transient brain activity disentangles fMRI resting-state dynamics in terms of spatially and temporally overlapping networks. Nature Communications, 6, 7751. ,

Kashyap, A., & Keilholz, S. (2019). Dynamic properties of simulated brain network models and empirical resting-state data. Network Neuroscience,3(2), 405-426.

Kashyap A, Keilholz S. Brain network constraints and recurrent neural networks reproduce unique trajectories and state transitions seen over the span of minutes in resting-state fMRI. Netw Neurosci. 2020 May 1;4(2):448-466. doi: 10.1162/netn_a_00129. PMID: 32537536; PMCID: PMC7286308.

Keilholz, S., Caballero-Gaudes, C., Bandettini, P., Deco, G.,
& Calhoun, V. (2017). Time-resolved resting-state functional magnetic resonance imaging analysis: Current status, challenges, and new directions. Brain Connectivity, 7(8), 465–481. ,

Kim J Z and Bassett D S 2019 Linear dynamics & control of brain networks (arXiv:1902.03309)

Le Bihan D, Breton E. Imagerie de diffusion in vivo par résonance magnétique nucléaire. C R Acad Sci Paris. 1985;T.301, Série II:1109–1112.

Liang, L., Deng, B., Wang, J., Wang, R., Wei, X., Yu, H.,
… Yang, C. (2015). Granger causality analysis in the neural mass model. 2015 34th Chinese Control Conference (CCC), 28AD, 4641–4645. ,

Liu, X., & Duyn, J. H. (2013). Time-varying functional network information extracted from brief instances of spontaneous brain activity. Proceedings of the National Academy of Sciences, 110(11), 4392–4397. ,

Mayberg, H. "Deep Brain Stimulation for Treatment-Resistant Depression." Journal of Affective Disorders, vol. 107, 2008, doi:10.1016/j.jad.2007.12.153

Majeed, W., Magnuson, M., Hasenkamp, W., Schwarb, H., Schumacher, E.
H., Barsalou, L., & Keilholz, S. D. (2011). Spatiotemporal dynamics of low frequency bold fluctuations in rats and humans. NeuroImage, 54(2), 1140–1150. ,

Majeed, W., Magnuson, M., & Keilholz, S. D. (2009). Spatiotemporal dynamics of low frequency fluctuations in BOLD fMRI of the rat. Journal of Magnetic Resonance Imaging, 30(2), 384–393. ,

Margulies, D., Ghosh, S., Goulas, A., Falkiewicz, M., Huntenburg, J., Langs, G., . . . Smallwood, J. (2016). Situating the default-mode network along a principal gradient of macroscale cortical organization.Proceedings of the National573Academy of Sciences USA,113(44), 12574-12579

Mesbahi, M., & Egerstedt, M. (2010).Graph theoretic methods in multiagent networks(Vol. 1). Princeton Series Applied Mathematics

Mcintyre, Cameron C, et al. “Uncovering the Mechanism(s) of Action of Deep Brain Stimulation: Activation, Inhibition, or Both.” Clinical Neurophysiology, vol. 115, no. 6, 2004, pp. 1239–1248., doi:10.1016/j.clinph.2003.12.024.

Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. Nature 518, 529–533 (2015). https://doi.org/10.1038/nature14236

Natalia, P., Gaudes, C. C., Dryden, I. L., Francis, S. T., & Gowland, P. A. (2012). Periods of rest in fMRI contain individual spontaneous events which are related to slowly fluctuating spontaneous activity. Human Brain Mapping, 34(6), 1319–1329.

Ouyang, G., Li, X., Dang, C., & Richards, D. A. (2008). Using recurrence plot for determinism analysis of EEG recordings in genetic absence epilepsy rats. Clinical Neurophysiology, 119(8), 1747–1755. ,

Pandarinath, C., Ames, C., Russo, A., Farshchian, A., Miller, L., Dyer, E., & Kao, J. (2018). Latent factors and dynamics in motor cortex and their application to brainmachine interfaces.Journal of Neuroscience,38(44), 9390-9401

“Recurrent Neural Networks (RNN) with Keras; TensorFlow Core.” TensorFlow, www.tensorflow.org/guide/keras/rnn.

Rogers, Anna, et al. “Direct Electrical Stimulation of the Lateral Hypothalamus Consolidates Wake and Ameliorates Cataplexy in a Mouse Model of Narcolepsy Type 1.” Georgia Clinical & Translational Science Conference. 23 Feb. 2018, Braselton, Chateau Elan.

Ritter, P. (2017). Multimodal simulations based on realistic structural connectivity using the virtual brain platform. Organization for Human Brain Mapping, Vancouver, Canada.

Ritter, P., Schirner, M., McIntosh, A. R., & Jirsa, V. K. (2013). The virtual brain integrates computational modeling and multimodal neuroimaging. Brain Connectivity, 3(2), 121–145. ,

Ritter, P, et al. "Multi-scale principles of brain function." Brain Modes Conference. 12 Dec. 2019, Pokhara, Nepal

Sanz-Leon, P., Knock, S. A., Spiegler, A., & Jirsa, V. K. (2015). Mathematical framework for large-scale brain network modeling in the virtual brain. NeuroImage, 111, 385–430. ,

Saenger VM, Kahan J, Foltynie T, Friston K, Aziz TZ, Green AL, Hartevelt TJ, Cabral J, Stevner ABA, Fernandes HM, Deco, G.: Uncovering the underlying mechanisms and whole-brain dynamics of deep brain stimulation for Parkinson's disease. Sci. Rep. 2017, 7:1–14.

Senden, M., Reuter, N., van den Heuvel, M. P., Goebel, R., & Deco, G. (2017). Cortical rich club regions can organize state-dependent

functional network formation by engaging in oscillatory behavior. NeuroImage, 146, 561–574. ,

Shakil, S., Lee, C.-H., & Keilholz, S. D. (2016). Evaluation of sliding window correlation performance for characterizing dynamic functional connectivity and brain states. NeuroImage, 133, 111–128. ,

Shen, K., Hutchison, R. M., Bezgin, G., Everling, S., & McIntosh, A. R. (2015). Network structure shapes spontaneous functional connectivity dynamics. Journal of Neuroscience, 35(14), 5579. ,

Skudlarski, P., Jagannathan, V., Kanchana amd Calhoun, Hampson, M., & Skudlarska, G., Beata Pearlson. (2008). Measuring brain connectivity: Diffusion tensor imaging validates resting state temporal correlations.NeuroImage,43, 554-561

Smith, S. M., Fox, P. T., Miller, K. L., Glahn, D. C., Fox, P. M., Mackay, C. E., … Beckmann, C. F. (2009). Correspondence of the brain's functional architecture during activation and rest. Proceedings of the National Academy of Sciences, 106(31), 13040. ,

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting.Journal of Machine Learning Research,15, 1929-1958

Stam, C. J., van Straaten, E. C. W., Van Dellen, E., Tewarie, P., Gong, G., Hillebrand, A., … Van Mieghem, P. (2016). The relation between structural and functional connectivity patterns in complex brain networks. International Journal of Psychophysiology, 103, 149–160. ,

Stephan, K. E., Weiskopf, N., Drysdale, P. M., Robinson, P. A., & Friston, K. J. (2007). Comparing hemodynamic models with DCM. NeuroImage, 38(3-3), 387–401. ,

Tagliazucchi, E., Balenzuela, P., Fraiman, D., & Chialvo, D. R. (2012). Criticality in large-scale brain fMRI dynamics unveiled by a novel point process analysis. Frontiers in Physiology, 3, 15. ,

Tessitore A, Giordano A, Russo A, Tedeschi G. Structural connectivity in Parkinson's disease. Parkinsonism & Related Disorders. 2016;22 Suppl 1:S56–9. pmid:26394678

Thompson, G. J., Pan, W.-J., Magnuson, M. E., Jaeger, D., & Keilholz, S. D. (2014). Quasi-periodic patterns (QPP): Large-scale dynamics in resting state fMRI that correlate with local infraslow electrical activity. NeuroImage, 84, 1018–1031. ,

Thomas, C., Ye, F., Irfanoglu, O., Modi, P., Saleem, K., Leopold, D., & Pierpaoli, C. (2014). Anatomical accuracy of brain connections derived from diffusion mri tractography is inherently limited.Proceedings of the National Academy of Sciences USA,111(46), 16574-16579

Tournier, J. D., Calamante, F., & Connelly, A. (2012). MRtrix: Diffusion tractography in crossing fiber regions. International Journal of Imaging Systems and Technology, 22(1), 53–66. ,

Van Essen, D. C., Smith, S. M., Barch, D. M., Behrens, T. E., Yacoub, E., Ugurbil, K., & WU-Minn HCP Consortium. (2013). The WU-Minn Human Connectome Project: An overview. NeuroImage, 80, 62–79. ,

Vidaurre, D., Abeysuriya, R., Becker, R., Quinn, A., Alfaro-Almagro, F., Smith, S., & Woolrich, M. (2018). Discovering dynamic brain networks from big data in rest and task.NeuroImage,180B, 646-656

Wagle Shukla A, Zeilman P, Fernandez H, Bajwa JA, Mehanna R. DBS Programming: An Evolving Approach for Patients with Parkinson's Disease. Parkinson's Disease. 2017;2017:8492619. doi:10.1155/2017/8492619.

Webber, C. L., & Marwan, N. (Eds.). (2015). Recurrence quantification analysis: Theory and best practices. Cham, Switzerland: Springer. ,

Yousefi, B., Shin, J., Schumacher, E. H., & Keilholz, S. D. (2018). Quasi-periodic patterns of intrinsic brain activity in individuals and their relationship to global signal. NeuroImage, 167, 297–308. ,

Zalesky, A., Fornito, A., Cocchi, L., Gollo, L. L., & Breakspear, M. (2014). Time-resolved resting-state brain networks. Proceedings of the National Academy of Sciences USA, 201400181.