NeuroField: A Neural Field Theory simulation toolbox

P.K. Fung*, R.G. Abeysuriya, X. Zhao, P.A. Robinson

School of Physics, University of Sydney, New South Wales, Australia

Abstract

Neural field models are a powerful, computationally efficient approach to modeling large-scale brain activity. NeuroField is an extensible software package to simulate neural field equations in a wide range of models. The basic element of neural field theory (population activity, wave propagation, and synaptic effects) can be assembled into arbitrary networks and integrated numerically to predict brain activity. NeuroField also includes MATLAB and Python routines for higher-level analysis including the power spectrum. NeuroField is implemented in C++ and has been tested on a range of Linux distributions, Microsoft Windows, and Mac OS X. Extensive user documentation and examples are provided, and typical use of NeuroField does not require C++ experience. NeuroField is open-source and available (http://physics.usyd.edu.au/brain/neurofield) under the GNU license for non-commercial use.

Keywords: EEG, neurophysiology, methods, modeling

1. Introduction

Neural field modeling has proved to be a powerful technique for constructing relatively simple, physiologically based models of the brain that are capable of predicting EEG and correlate well with experimental data Deco et al. (2008), Pinotsis et al. (2012). We have developed a neural field corticothalamic model of the brain (Robinson et al., 2002, 2004, 2005, 2001, Rowe et al., 2004) that we

*Corresponding author. Tel. +61 9036 7274
Email address: `ffung@physics.usyd.edu.au` (P.K. Fung)

have previously used to investigate the alpha rhythm (O'Connor and Robinson, 2004, Robinson et al., 2003), age-related changes to the physiology of the brain (van Albada et al., 2010), evoked response potentials (Kerr et al., 2011, Rennie et al., 2002), seizures (Breakspear et al., 2006), and many other phenomena.

The key features of neural field models are captured by the three key equations governing general neural field theory

$$
\begin{aligned}
D_{ab}V_{ab}(\mathbf{r},t) &= \nu_{ab}\phi_{ab}(\mathbf{r},t),\\
Q_a(\mathbf{r},t) &= S_a\big[\sum_b V_{ab}(\mathbf{r},t)\big],\\
\mathcal{D}_{ab}\phi_{ab}(\mathbf{r},t) &= Q_b(\mathbf{r},t-\tau_{ab}).
\end{aligned}
$$

which represent synapto-dendritic smoothing, dendritic summation and firing response, and damped wave propagation, respectively. The differential operators are

$$
\begin{aligned}
D_\alpha(t) &= \frac{1}{\alpha\beta}\frac{d^2}{dt^2} + \left(\frac{1}{\alpha}+\frac{1}{\beta}\right)\frac{d}{dt} + 1,\\
\mathcal{D}_a(\mathbf{r},t) &= \frac{1}{\gamma_a^2}\frac{\partial^2}{\partial t^2} + \frac{2}{\gamma_a}\frac{\partial}{\partial t} + 1 - r_a^2\nabla^2,
\end{aligned}
$$

and the sigmoid population response S_a is given by

$$
Q_a = S(V_a) = \frac{Q_{\max}}{1+\exp[-(V_a-\theta)/\sigma']},
$$

The relationship between these quantities is schematically illustrated in Fig. 1.

The most challenging part of applying neural field theory is the implementation of the numerical solver. Several factors contribute to making the numerical integration of neural field equations difficult. In particular, propagation delays between neural populations result in delay-differential equations that require special handling of temporal history. Further, propagation of neural fields according to a damped wave equation adds two dimensions to the system, and requires a relative sophisticated finite-differencing scheme that takes into account the geometry of the system. In addition, periodic boundary conditions must be correctly handled during the integration.

We have developed NeuroField to provide a software package that solves the neural field equations for arbitrary neural populations, and contains library code for analysis and visualization, thus removing the barriers to quickly testing and analyzing neural field models. The software is designed to be easily extensible with basic C++ programming skills, making it simple to expand upon the basic model to include new phenomena.

2. Method and Results

2.1. Key features/Basic functions

The essential role of NeuroField is to take as input a model and its initial conditions, and to output one or more time series corresponding to the result of integrating the neural field equations. A model is a specification of neural populations (amounting to defining their firing response to input from other populations including synapto-dendritic effects), and connections between the populations including how neural signals propagate through space. Sensory or other stimulus is implemented as a neural population that recieves no input from other populations, and has a pre-defined firing pattern. Integration of the neural field equations provides several quantities of interest. Most notably, the signals from populations can be associated with local field potentials (LFP)

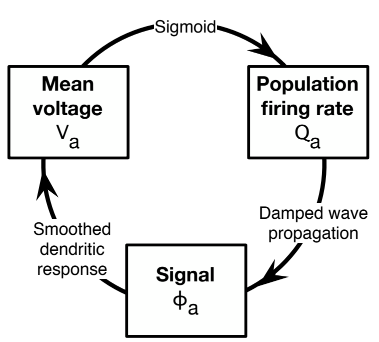

Figure 1: Schematic overview of the key dynamic quantities of neural field models, and the relationships between them.

or EEG depending, and these predictions can be directly compared against experimental data. The soma potential or firing rate of the neural populations can be compared to individual neuron data. Changes to synaptic strength can be monitored when simulating neural plasticity. When simulating spatially extended populations, spatial correlations and patterns of activity can also be analyzed.

Schematic showing blocks for couple, dendrite, qresponse and the relationship between them. Follow the order in which solver evaluates them

The core of NeuroField is a C++ program that accept a human-readable plain text configuration file. The output from NeuroField is a plain text output file containing all of the requested simulation variables for each time step. The syntax of the output file has been designed to be simple to parse, and the NeuroField package includes reference parsers for Matlab and Python. These parsers may also help serve as a starting point for implementation of parsers other programming languages.

There are a number of common ways to analyze the output from neural field models. First, plots of the time series are useful for directly viewing neural oscillations, evoked responses, seizures, monitoring plasticiity, and verifying stability. Second, calculation of the power spectrum, which is often compared to experimental EEG. This can also involve detection of multiple spatial modes of activity and incorporation of volume conduction, to account for effects introduced by electrodes in real-world recordings. Finally, spatial patterns of activity and propagation of waves of activity can be visualized on a surface plot. All of these basic analyses are included as Matlab programs in the NeuroField toolbox.

2.2. Data structures

NeuroField solves each part of the neural field model with an object:

$$
\begin{aligned}
P &= \nu_{ab}\phi_{ab}, && \texttt{Couple}\\
D_{ab}V_{ab} &= P, && \texttt{Dendrite}\\
Q_a &= S_a\big[\sum_b V_{ab}\big], && \texttt{QResponse/Pop}\\
\mathcal{D}_{ab}\phi_{ab} &= Q_b, && \texttt{Propag}
\end{aligned}
$$

The behavior of each of these objects encapsulates the details of the neural field equations.

2.2.1. Populations

A population contains a QResponse object, which defines how the population potential is mapped to firing rate. The QResponse may return the sigmoid of the voltage, as written above. Alternatively it could be an arbitrary function, such as a linearized version of the sigmoid. Effects like bursting can also be included.

Code snippets

2.2.2. Propagators

The neural field generated by a population propagates according to the associated Propagator. For populations where spatial propagation effects are insignificant, this propagator may simply be unity. The propagator object takes into account the spatial geometry of the problem, so can be applied to both 2D sheets and spherical geometries.

Code snippets

2.2.3. Couples

The coupling strength of a connection is typically constant, but can be an arbitrary function of a range of different factors, allowing modeling of a wide variety of plasticity effects including STDP and CaDP

2.2.4. Input and output

- Mention structure of config and output files
- Largely human readable, but relatively simple to construct and parse programatically
- Examples of the output file

Example output, few times, maybe 2 nodes 2 traces

2.3. Visualization and analysis

2.3.1. Helper scripts

NeuroField is packaged with several helper scripts written in MATLAB to assist with running, analyzing and visualizing models.

2.3.2. Reading output files

nf_read() allows users to parse the output file from NeuroField into a MATLAB struct object. nf_grid reshapes the output for handling matrices.

2.3.3. Writing config files

nf_eirs() demonstrates writing a configuration file, running it with nf_run(), and then reading it with nf_read(). This demonstrates a complete MATLAB-based toolchain for using NeuroField.

2.3.4. Calculating power spectra

The power spectrum can be obtained by FFT, but correct normalization and calculation of the power spectrum including multiple spatial modes can be challenging to implement. We have implemented a 3D FFT algorithm that correctly normalizes the output and includes volume conduction effects that selective attenuate spatial modes depending on their wavenumber. The result can be directly compared to analytical predictions.

2.3.5. Visualizing output

The nf_extract() function makes it easy to select data for plotting from a NeuroField object. nf_movie can plot an animation of the output

3. Discussion

We have developed NeuroField to provide an extensible, reliable framework for integrating nonlinear delay differential equations including spatial propagation. NeuroField is aimed for use by researchers who have constructed neural field models of the brain that require numerical integration. In this section, we review some usage and performance considerations.

3.1. White noise stimulus

- White noise requires stocastic DE integrator, effectively Euler

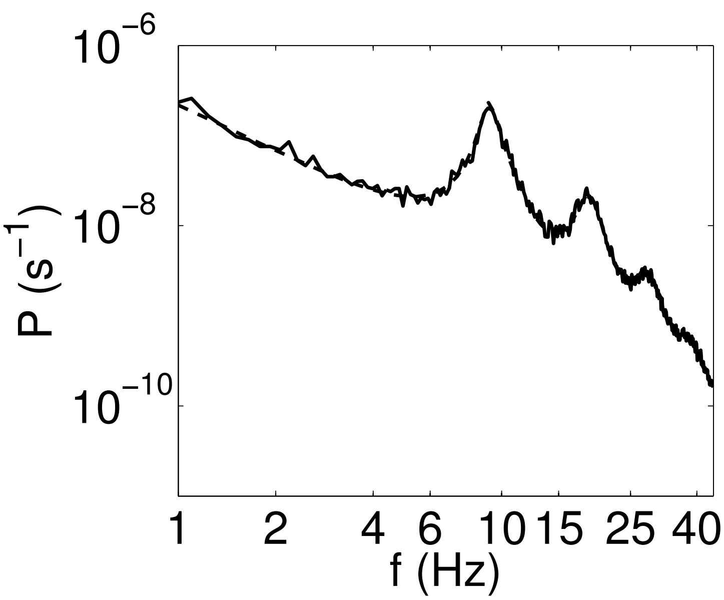

Figure 2: Comparision of linear analytic spectrum with the power spectrum computed using the NeuroField package analysis tools.

- Noise amplitude depends on grid resolution as this affects the possible bandwidth. Similar features depend on frequency domain power so noise needs to be normalized correctly

3.2. Performance

- Some numbers about the runtime and memory requirements of NeuroField

- Note that the memory requirements scale with the grid size, and the grid size depends on Lx and the propagator lengths (automatically enforced)

- Also that the delays in the system cause O(n) increases in memory usage

4. Acknowledgements

This work was supported by the Australian Research Council, National Health and Medical Research Council (through the Center for Integrated Research and Understanding of Sleep), and the Westmead Millennium Institute.

5. References

Breakspear, M., Roberts, J. A., Terry, J. R., Rodrigues, S., Mahant, N., Robinson, P. A., Sep. 2006. A unifying explanation of primary generalized seizures through nonlinear brain modeling and bifurcation analysis. Cerebral Cortex 16 (9), 1296–1313.
URL `http://www.ncbi.nlm.nih.gov/pubmed/16280462`

Deco, G., Jirsa, V. K., Robinson, P. A., Breakspear, M., Friston, K. J., Jan. 2008. The dynamic brain: from spiking neurons to neural masses and cortical fields. PLoS Computational Biology 4 (8), e1000092.
URL `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2519166&tool=pmcentrez&rendertype=abstract`

Kerr, C. C., Rennie, C. J., Robinson, P. A., 2011. Model-based analysis and quantification of age trends in auditory evoked potentials. Clinical Neurophysiology 122, 134–147.

O'Connor, S. C., Robinson, P. A., Jul. 2004. Spatially uniform and nonuniform analyses of electroencephalographic dynamics, with application to the topography of the alpha rhythm. Physical Review E 70 (1), 11911.

Pinotsis, D. A., Moran, R. J., Friston, K. J., Jan. 2012. Dynamic causal modeling with neural fields. NeuroImage 59 (2), 1261–1274.
URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3236998&tool=pmcentrez&rendertype=abstract

Rennie, C. J., Robinson, P. A., Wright, J. J., Jun. 2002. Unified neurophysical model of EEG spectra and evoked potentials. Biological Cybernetics 86 (6), 457–471.
URL http://www.ncbi.nlm.nih.gov/pubmed/12111274

Robinson, P. A., Rennie, C. J., Rowe, D. L., Apr. 2002. Dynamics of large-scale brain activity in normal arousal states and epileptic seizures. Physical Review E 65 (4), 41924.

Robinson, P. A., Rennie, C. J., Rowe, D. L., O'Connor, S. C., 2004. Estimation of multiscale neurophysiologic parameters by electroencephalographic means. Human Brain Mapping 23 (1), 53–72.
URL http://dx.doi.org/10.1002/hbm.20032

Robinson, P. A., Rennie, C. J., Rowe, D. L., O'Connor, S. C., Gordon, E., May 2005. Multiscale brain modelling. Philosophical Transactions of the Royal Society B 360 (1457), 1043–1050.

Robinson, P. A., Rennie, C. J., Wright, J. J., Bahramali, H., Gordon, E., Rowe, D. L., Jan. 2001. Prediction of electroencephalographic spectra from neurophysiology. Physical Review E 63 (2), 21903.

Robinson, P. A., Whitehouse, R. W., Rennie, C. J., Aug. 2003. Nonuniform corticothalamic continuum model of electroencephalographic spectra with application to split-alpha peaks. Physical Review E 68 (2), 21922.

Rowe, D. L., Robinson, P. A., Rennie, C. J., 2004. Estimation of neurophysiological parameters from the waking EEG using a biophysical model of brain dynamics. Journal of Theoretical Biology 231 (3), 413–433.
URL http://dx.doi.org/10.1016/j.jtbi.2004.07.004

van Albada, S. J., Kerr, C. C., Chiang, A. K. I., Rennie, C. J., Robinson, P. A., Jan. 2010. Neurophysiological changes with age probed by inverse modeling of EEG spectra. Clinical Neurophysiology 121 (1), 21–38.
URL http://www.ncbi.nlm.nih.gov/pubmed/19854102