Leveraging Julia's automated differentiation and symbolic computation to increase spectral DCM flexibility and speed

David Hofmann[a,b], **Anthony G. Chesebro**[a,d], **Chris Rackauckas**[b], **Lilianne R. Mujica-Parodi**[a,c,d,f], **Karl J. Friston**[e], **Alan Edelman**[b], **and Helmut H. Strey**[a,c,d]

[a]Laufer Center for Physical and Quantitative Biology, State University of New York at Stony Brook, NY; [b]Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology, MA; [c]Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital and Harvard Medical School, MA; [d]Department of Biomedical Engineering, State University of New York at Stony Brook, NY; [e]Wellcome Trust Centre for Neuroimaging, University College London, London UK; [f]Baszucki Endowed Chair for Metabolic Neuroscience, Stony Brook University, NY

This manuscript was compiled on October 27, 2023

Using neuroimaging and electrophysiological data to infer neural parameter estimations from theoretical circuits requires solving the inverse problem. Here, we provide a new Julia language package designed to i) compose complex dynamical models in a simple and modular way with ModelingToolkit.jl, ii) implement parameter fitting based on spectral dynamic causal modeling (sDCM) using the Laplace approximation, analogous to MATLAB implementation in SPM12, and iii) leverage Julia's unique strengths to increase accuracy and speed by employing Automatic Differentiation during the fitting procedure. To illustrate the utility of our flexible modular approach, we provide a method to improve correction for fMRI scanner field strengths (1.5T, 3T, 7T) when fitting models to real data.

neural | circuit | causal | model | DCM | SPM | neuroimaging | LFP | fMRI | Julia | Neuroblox

Answering fundamental questions in neuroscience often requires estimating circuits from neuronal populations (1–4). Causal models of interactions between different brain regions further require fitting a model that can capture both underlying neuronal activity and accurately assess the distortion and noise introduced by the measurement modality (1, 2). One powerful approach is Dynamic Causal Modeling (DCM), introduced as part of the Statistical Parametric Mapping (SPM) toolbox (1, 2). DCM provides methods and an associated pipeline for modeling interactions between neuronal populations at the cortical level, fitting models from local field potential (LFP) recordings (5–7), electroencephalography (EEG) data (8), and functional MRI (fMRI) sequences (9–14). Neuroscience applications of DCM have provided insights into both healthy (3, 4, 15, 16) and impaired (17) neuronal functioning, in response to task-based (3, 4, 16) and resting-state (2, 17) stimuli. Additionally, the underlying mathematical approach of DCM has been extended to applications outside of neuroscience that benefit from causal inference (18).

Broadly speaking, DCM is a procedure used to infer model structure from data (19, 20). Models are defined that describe the underlying neuronal dynamics, typically at the meso- or macro-scale, as well as the measurement process (BOLD, EEG, LFP). A variational Bayes approach is used to estimate the parameter posterior distribution by optimizing the evidence lower bound (2, 6). Finally, the evidence lower bound or free energy is used for model comparison (21, 22). By changing the prior distribution, a user can include and exclude parameters, such as connection weights between regions. Additionally, through model comparison, the user can judge which regions are most likely connected and whether they are excitatory or inhibitory. Over time, this approach has evolved and become more powerful, including the adoption of a computationally more efficient DCM algorithm in the spectral domain (2). Signals, assumed to be stationary, are transformed to the spectral domain by the computation of cross-spectral densities (CSD).

The most current implementation of DCM, as part of the Statistical Parametric Mapping toolbox (SPM12; available at fil.ion.ucl.ac.uk/spm/software/spm12/), has been extensively and successfully applied to a wide variety of data modalities, cognitive functions, and disease processes (3, 4, 16, 17). Here, we build on this achievement to extend its capabilities in two ways. First, one of the practical roadblocks to the wide adoption of multi-scale computational neuroscience has been limitations due to computation speed. DCM's implementation in MATLAB introduces inherent computational barriers that become increasingly problematic as datasets scale to meet current reproducibility standards (23). For this reason, we chose to code our package in Julia, as part of the Neuroblox computational neuroscience platform (neuroblox.org), as Julia is a high-performance computing language that is known to show orders of magnitude speed improvements in ODE systems compared to both MATLAB and Python (24). Second, we wanted to incorporate additional flexibility into the pipeline in terms of easy access to choices for neural mass models, model assembly, transfer functions within and between scales and modalities, as well as parameter fitting.

In this work, we provide a new implementation of DCM in the Julia programming language (julialang.org), building on the ModelingToolkit package (25). We demonstrate that this implementation can offer results identical to the original SPM12 implementation at a fraction of the computation speed in an open-source language. Additionally, we construct this

[2]To whom correspondence should be addressed. E-mail: helmut.strey@stonybrook.edu

implementation to allow for great user flexibility in model selection and specification, opening the door to new modeling techniques not achieved in prior implementations. We then illustrate the benefits of this flexibility by showing how the new version can correct for MRI scanner field strengths, a feature that did not exist prior to this implementation.

Spectral Dynamic Causal Modeling. The distinctive feature of spectral DCM as compared to other forms of DCM is that it models the cross-spectral density of a signal. This implies the assumption of stationarity of the system during measurement. The usual state space formulation is assumed

$$\begin{aligned}\dot{\mathbf{x}} &= f(\mathbf{x};\boldsymbol{\theta}_f) + \boldsymbol{\epsilon}_f \\ \mathbf{y} &= g(\mathbf{x};\boldsymbol{\theta}_g) + \boldsymbol{\epsilon}_g\end{aligned} \quad [1]$$

For Eqns. 1, $\mathbf{y}$ denotes the measurement of the (hidden) dynamics $\mathbf{x}$, which could for instance be the neuronal dynamics and hemodynamic response. $\boldsymbol{\epsilon}_f$ and $\boldsymbol{\epsilon}_g$ denote intrinsic or endogenous noise and measurement noise, respectively. In spectral DCM these are assumed to follow 1/f noise; that is, noise that follows a power law distribution in frequency space. In fMRI measurements, the states x comprise not just the neuronal dynamics (x_n) but also the hemodynamic response to the neuronal activity (x_h). A further assumption made by spectral DCM is linearity of the hidden neuronal dynamics $f_n(\mathbf{x}_n;\boldsymbol{\theta}_n) = \boldsymbol{\theta}_n\mathbf{x}_n$. The dynamics for the hemodynamic response $f_h(\mathbf{x}_h;\boldsymbol{\theta}_h)$ are modeled by the balloon model, and the measured signal is the blood oxygen level depletion (BOLD) denoted by $g(\mathbf{x};\boldsymbol{\theta}_g)$. Both models are detailed in (20), which is based on work by Buxton et al. (26). Here we describe the model used in the code in more detail. The balloon model describes the dynamics of blood vessels in response to the neuronal signals. The cerebral blood flow $u(t)$ is driven by the neuronal activity x_n through the induction of a vasodilatory signal $s(t)$ governed by the following dynamics:

$$\dot{s} = x - \kappa s - \gamma(u-1) \quad [2]$$

$$\dot{u} = s \quad [3]$$

The balloon model describes the change in volume $v(t)$ due to the flow as well as the related changes of the blood oxygenation level $q(t)$. Its dynamics according to (26) are

$$\begin{aligned}\tau\dot{v} &= u - v^{1/\alpha} \\ \tau\dot{q} &= uE(u,E_0)/E_0 - v^{1/\alpha}q/v\end{aligned} \quad [4]$$

with $E(u,E_0) = 1-(1-E_0)^{1/u}$ as a reasonable approximation for $E(u,E_0)$. Finally, based on blood oxygenation $q(t)$ and blood volume $v(t)$ we compute the BOLD signal according to the following equation

$$\lambda(q,v) = V_0\left[k_1(1-q) + k_2\left(1-\frac{q}{v}\right) + k_3(1-v)\right] \quad [5]$$

with the following parameter definitions

$$\begin{aligned}k_1 &= 4.3\theta_0 E_0 T_E \\ k_2 &= \epsilon r_0 E_0 T_E \\ k_3 &= 1-\epsilon\end{aligned}$$

where ϵ is ratio of intra- and extravascular signal, E_0 is the resting oxygen extraction fraction, and r_0 is the slope of the relation between the intravascular relaxation rate R^*_{2I} and oxygen saturation. The solution to the differential equations Eq. (1) can be approximated by the first order of a Taylor expansion as

$$\mathbf{y}(t) = \kappa(\tau) * \boldsymbol{\epsilon}_f(t) + \boldsymbol{\epsilon}_g(t) \quad [6]$$

with $\kappa(\tau) = \nabla_x g \exp(\mathcal{J}_f\tau)$. To find an expression for the cross-spectral density we Fourier transform Eq. (6)

$$\tilde{\mathbf{y}}(\omega) = \tilde{\kappa}(\omega)\tilde{\boldsymbol{\epsilon}}_f(\omega)\tilde{\kappa}(\omega)^* + \tilde{\boldsymbol{\epsilon}}_g(\omega) \quad [7]$$

where $\tilde{\kappa}$, $\tilde{\boldsymbol{\epsilon}}_f$, and $\tilde{\boldsymbol{\epsilon}}_g$ denote the Fourier transforms of the convolution kernel and noises respectively. The cross-spectral density is then

$$\begin{aligned}S_y(\boldsymbol{\theta}) &= \langle\tilde{\mathbf{y}}(\omega)\cdot\tilde{\mathbf{y}}(\omega)^*\rangle \\ &= \tilde{\kappa}(\omega)S_{\epsilon_f}\tilde{\kappa}(\omega)^* + S_{\epsilon_g}\end{aligned} \quad [8]$$

S_{ϵ_f} and S_{ϵ_g} are the cross-spectral densities of the state noise and measurement noise, respectively. Note that due to our assumption of uncorrelated noises, these are diagonal matrices. While the cross-spectrum of the measurement noise will have identical values on the diagonal, the state noise is composed of a global and a region-specific component and thus diagonal elements can vary. This deviates from the presentation in (2) but is consistent with the implementation in SPM12.

We thus derived a linear approximation of the cross-spectral density $S_y(\boldsymbol{\theta})$ that is based on the details of the time-continuous state space model given in Eq. (1). Next we would like to estimate the parameters of the model as well as the noise parameters and therefor need to define an optimization procedure. We follow the approach pursued by DCM, that is a Variational Bayes approach with Laplace approximation, termed Variational Laplace for brevity (27).

Variational Laplace.

An inverse problem consists of estimating parameters from data. In a Bayesian setting this implies computing the posterior distribution

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{Z} \qquad [9]$$

In the ideal setting where we can compute an analytical expression for the posterior given the likelihood, a prior distribution $p(\theta)$ and the normalization Z, also called the model evidence or marginalized likelihood ($p(y) = \int p(y|\theta)p(\theta)\mathrm{d}\theta$), the Bayes formula (Eq. (9)) would provide the true posterior distribution. However, typically there is no such closed form solution and instead we have to resort to sampling techniques (e.g., Markov Chain Monte Carlo) which are unbiased estimators, or we compute an approximation to the posterior. Variational Bayes is an approach that borrows from variational calculus by introducing a parameterized function $q(\theta)$ that will be shaped, through an optimization procedure, to approximate the true posterior $q(\theta) \approx p(\theta|y)$. This optimization procedure requires the definition of an objective function which is typically the evidence lower bound (ELBO), sometimes also called the variational lower bound or free energy (particularly in the context of DCM) due to its resemblance of the functional form of the free energy in physics (27). There are different ways to derive the ELBO; one approach is to start from the Kullback-Leibler (KL) divergence between the variational distribution and the posterior $D_{KL}(q(\theta)||p(\theta|y))$. Minimizing D_{KL} is equivalent to maximizing the ELBO

$$ELBO = \log(p(y)) - D_{KL}(q(\theta)||p(\theta|y)) \qquad [10]$$

Note that the KL divergence is only well defined if q has probability 0 at the same intervals as p; in other words, the support of the two distributions must be the same. Applying the Laplace approximation to simplify this problem means assuming a Gaussian variational posterior $q(\theta) = \mathcal{N}(\mu_\theta, \Sigma_\theta)$, hence the term variational Laplace. This is the approach taken by DCM.

Here, we present several advances that expand the capabilities of spectral DCM as implemented in SPM12. We provide a toolbox that has a user-friendly interface to add new models and link them in arbitrarily complex circuits, with built-in corrections for different acquisition field strengths not previously considered in DCM analyses. This software leverages ModelingToolkit.jl, a symbolic computation framework written in the programming language Julia, to achieve a high degree of modularity and thereby easy model composability (Fig. 1). While the implementation of DCM in Julia already provides a speed increase over the traditional MATLAB version, we also employ automatic differentiation to further improve computation speed.

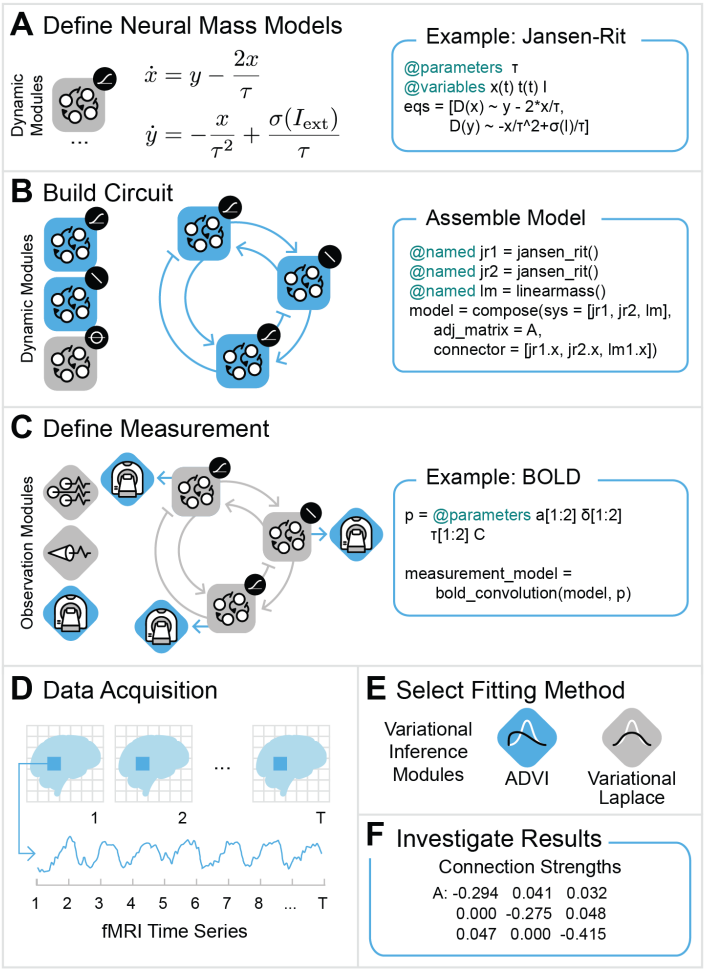

Fig. 1. Modular design for user-friendly circuit model assembly and data fitting. Blue boxes show code snippets that illustrate an example where two Jansen-Rit models (defined in **A**) and one linear model are connected to represent the neural activity in three cortical regions (**B**). In addition to a model for the neuronal activity, we can also define the measurement procedure; in this particular case, we assume fMRI measurements and thus add a BOLD signal model (**C**). Our Julia toolbox leverages ModelingToolkit.jl for easy modular circuit building. A user can either choose from a variety of predefined models, such as the Jansen-Rit neural mass model, or define their own. A model is defined symbolically based on ModelingToolkit.jl, thus it is straightforward to write the mathematical formulas as code. It is necessary to define a junction variable that accounts for the input to a region model as well as a connector variable that corresponds to the output of that region. Every region can be described by a different neural mass model (as illustrated here in terms of two Jansen-Rit models and one linear neural mass model) which substantially expands modeling power over the SPM12 toolbox, where regions are constrained to have dynamics of the same functional form. The function connect will then *connect* all blocks based on the adjacency matrix as well as the input and output variables of each block. Lastly, a fitting procedure can be selected (E) and parameter values estimated (F) from acquired data (D). (Graphic in (D) adopted from CC-licensed graphic in (28))

Modular design for user-friendly circuit model assembly and data fitting.

Defining a model should be as simple as writing down its equations. For this purpose our toolbox design leverages ModelingToolkit.jl (25) and Symbolics.jl (29) for symbolic computation which provide a simple and intuitive way to define a model that consists of differential equations (see Fig. 1A).

For instance, the user can define arbitrary models describing the dynamics of a particular brain region and assemble them into an overall model through an adjacency matrix (see Fig. 1B). In this paper we studied two different systems for the neuronal

dynamics: a simple linear ODE system as is typically used in SPM12 for the spectral DCM analysis of fMRI signals (2), and a more complex canonical micro-circuit model which is used to model LFP signals (6, 30, 31) (see Fig. 2). Definition of an observation model that relates the (hidden) neuronal activity to the actual neurophysiological measurement completes the model assembly (see Fig. 1C).

By providing measurements corresponding to the observation modality (Fig. 1D) the user can then pick a variational inference algorithm (Fig. 1E) to estimate the posterior probability of the model parameters (Fig. 1F). The toolbox provides a Variational Laplace inference algorithm analogous to the SPM12 implementation, as well as a Variational Laplace implementation that leverages automatic differentiation for the computation of the model Jacobian, both with and without symbolic computation. The inference is performed on the cross-spectral densities of the measured time series - thus, we assume that the underlying model is in a stationary state. The initial conditions of the states need to be selected and defined accordingly.

Using symbolic computation provides two important advantages: first, it facilitates modular definitions of models which makes them easily extendable, alterable, and composable; second, it provides a means of automatic analytical computation of gradients and Jacobians which avoids the potentially imprecise numerical computation of derivatives, or the effort of hard-coding the analytical derivatives for each new model.

Julia implementation provides substantial speed improvements.

Computational complexity can pose a bottleneck when analyzing high-dimensional datasets covering multiple brain regions. Julia provides a high-performance computing environment that reaches computational speeds comparable to C/C++ due to just-in-time compilation. We implemented the variational inference method based on the Laplace approximation, also called Variational Laplace, for parameter estimation analogous to the implementation in SPM12. We compare our method on simulated toy data with the SPM12 implementation to ensure consistency and correctness of our implementation in Julia.

We find that the Julia implementation provides a meaningful speed improvement (Fig. 3) while yielding the exact same results. Note that in SPM12 we need to set the DCM option "maxnodes" to a value equal to or larger than the number of regions we are inferring to ensure a correct comparison. Maxnodes will otherwise reduce the dimensionality of the problem to the value provided, which leads to a lower computational load.

Next we make the code amenable to automatic differentiation and use Julia's ForwardDiff.jl library to perform automatic differentiation on the Variational Laplace optimization. We find a further speed improvement as can be observed in Fig. 3, yielding an almost order of magnitude speed improvement for 10 regions over the MATLAB SPM12 implementation.

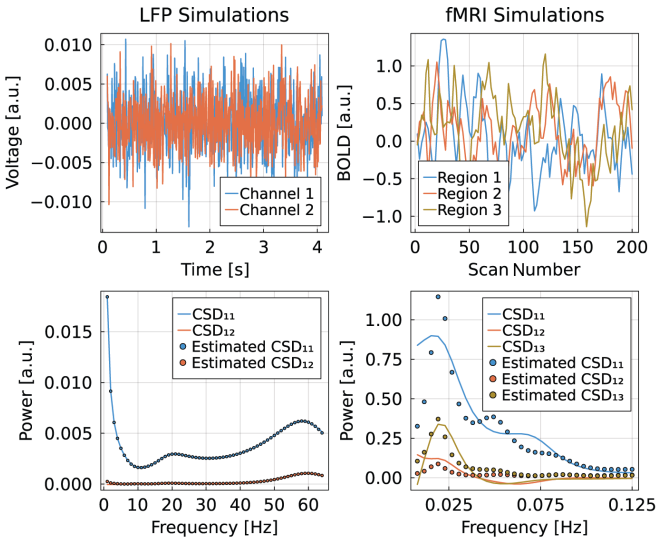

Fig. 2. Estimation of cross-spectral densities of simulated data. Top row are examples of simulated signals, left are LFP measurements of two channels, and right are fMRI measurements of three regions. Plots in the bottom row are examples of simulated (lines) and estimated (dots) cross-spectral densities. We find a nearly perfect match for LFP signals, while fMRI signals show only a very rough agreement. The key difference is the nonlinear transformation of the neuronal signals through the hemodynamic response function; the LFP signals are only affected by a constant gain term that may vary for each channel.

Automatic Differentiation.

Julia provides different libraries for performing automatic differentiation. We chose ForwardDiff.jl due to its robustness and good performance for systems with up to 100 dimensions (32, 33).

The SPM12 implementation of the matrix-exponential of the kernel $\kappa(\tau)$ given in Eq. (6) requires an eigenvalue decomposition. This poses a particular challenge since the eigenvalue decomposition in the standard library LinearAlgebra.jl is not a native Julia implementation but based on the external, compiled BLAS library and thereby is not amenable to automatic differentiation. The package GeneralizedLinearAlgebra.jl provides linear algebra operations that are fully implemented in Julia; however, its eigenvalue decomposition only provides eigenvalues but not the corresponding eigenvectors at the point of writing this manuscript.

Thus we created a dispatch of the eigenfunction defined in LinearAlgebra.jl. The dispatch implements the analytical derivative of eigenvalues and eigenvectors (34, 35). However, the analytic derivative of eigenvectors with degenerate eigenvalues has poles and is thus numerically unstable around degeneracies. Indeed, the

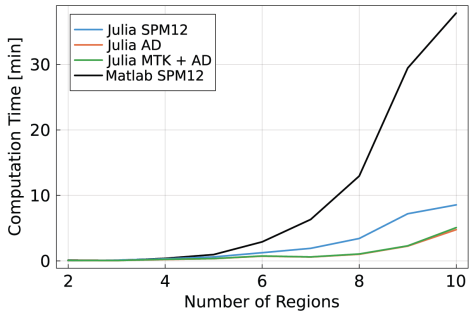

Fig. 3. Julia implementation provides substantial speed improvements. Graphs show the speed increase with respect to the MATLAB SPM12 (black) implementation in dependence on the number of regions. Notable speed improvement is already achieved by a plain Julia re-implementation of the SPM12 DCM (blue). Employing Automatic Differentiation (AD) results in an additional speed improvement of up to 10-fold for 10 regions (orange). When additionally employing symbolic computation with ModelingToolkit.jl (MTK) to ensure compositionality of model building, computation speed suffers by a negligible factor as can be appreciated by observing the green line.

standard SPM initial conditions are set to have degenerate eigenvalues, therefore, we introduce small Gaussian perturbations to the initial conditions as well as to the prior of the connection matrix.

Complex numbers pose an additional hurdle since the Dual type of ForwardDiff.jl is not defined for complex numbers. This was simply overcome by dealing with the real and imaginary parts separately and then assembling the Dual of the real part and the Dual of the imaginary part into the complex number type. We improved in speed performance over a previous implementation that addresses these problems (36).

Influence of field strength on hemodynamic variables in Balloon-Windkessel model.

In the hemodynamic model used by DCM, there are six parameters that can affect the fit substantially. Five of them have been discussed at length elsewhere (37–39), and so will be briefly restated here. The echo time (T_E) is always taken directly from the data being fit and is thus specified by individual users (2). The resting venous volume percent (V_0) and resting oxygen extraction fraction (E_0) are physiological parameters that - while variable across individuals - do not vary systematically in different field strengths and are thus assumed to be single values established in prior work (37–39). The ratio of intra- to extravascular signal (ϵ) is fit on a region-by-region basis, which has been shown to improve accuracy (20). The slope of intravascular relaxation rate as a function of oxygen (r_0) can potentially vary at different field strengths; however, at fields above 3T, the contribution of this term becomes negligible due to the dominance of k_1 in the balloon model (see Appendix A.3 in (38) for a full discussion of this effect).

The remaining parameter of the balloon model is the maximum frequency offset at the outer surface of a magnetized vessel (ν_0). This value has been shown to vary in different field strengths, and the exact derivation of this quantity has been addressed at length elsewhere. Summarizing this work briefly (see (39)), the maximum frequency offset at the surface of a blood vessel is dependent on the gyromagnetic ratio of protons in a field ($\gamma \approx 128$MHz), the susceptibility difference ($\Delta\chi_0 = 0.264 \times 10^{-6}$), the hematocrit concentration ($H = 0.44$), the venous saturation ($Y = 0.6$), and the field strength B_0, yielding the final expression of frequency offset

$$\nu_0 = \gamma \Delta\chi_0 H (1 - Y) B_0$$

Prior work has estimated different values depending on different assumptions of hematocrit, venous saturation, or susceptibility difference, and individual variation (e.g., sex differences in baseline hematocrit concentration (39)) can cause significant variability of this constant. However, for the work at hand, we note the linear dependence on field strength, with a doubling of this constant when moving from 1.5T to 3T, and even greater differences when moving to ultra high fields (7T and 9.4T) used in modern human imaging studies. We probe these differences further in simulations and real data.

Influence of field strength on hemodynamic response function.

While there are many possible reasons for the effect that varying ν_0 with field strength has on the fitting results of DCM, one important consideration is a direct effect on the shape of the hemodynamic response function (HRF). In this section, we probe the effect of field strength on the Balloon-Windkessel model HRF to determine what, if any, these effects are.

First, we must compute the HRF from the balloon model. To do so, we simulated the model 50x at four different field strengths (1.5T, 3T, 7T, and 9.4T) common in human neuroimaging studies. Rather than using a simulated neuronal signal, we input white noise into the model to allow for computing the HRF directly from simulated data via division in the Fourier domain.

Considering the original neuronal signal $x(t)$ and the output BOLD signal $y(t)$, their relationship can be described by a convolution with an HRF $h(t)$ such that

$$(x * h)(t) = y(t)$$

Converting this to the Fourier domain, this becomes a simple multiplication problem

$$x(\omega)h(\omega) = y(\omega)$$

and the HRF can be determined by simple division if the input signal $x(t)$ is white noise (guaranteeing $x(\omega)$ is a constant). We note that this approach does not work in most cases, as real signals can easily produce zero values in the Fourier domain, leading to a division by zero. However, in the case of pure noise as is considered here, we know the Fourier transform to be nearly constant with only minor variability, and so this approach is justified.

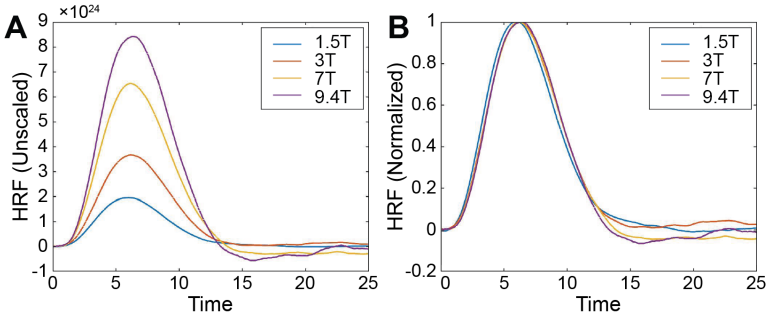

Fig. 4. Field strength affects magnitude and shape of hemodynamic response function in Balloon-Windkessel model. A. The primary effect of field strength (captured by ν_0) is the increase in magnitude of the HRF. The HRFs plotted here are the result of averaging across all 50 runs and temporal smoothing with a 1s filter. B. To examine more subtle effects which remain after normalizing the fMRI signal, all field strengths from A are plotted on a normalized scale. We note that higher field strengths demonstrate the characteristic undershoot after the primary response, a feature lacking at lower field strengths.

The results of these computations are shown in Fig. 4. Fig. 4a shows the strongest effect of field strength on the HRF - namely, the increase in magnitude of the response curve. However, as this can be removed by standard fMRI pre-processing techniques (signal normalization), Fig. 4b shows the same HRF curves normalized to a common scale. In this case, we observe a slight delay (∼250ms) in the peak times of higher fields (3T, 7T, and 9.4T) compared to the lowest field (1.5T). More

importantly, the characteristic undershoot of the HRF is only present in higher fields (7T and 9.4T). Finally, it appears that the overall width of the peaks does not vary significantly. To demonstrate this, we computed the mean peak and full width at half of the peak of each signal (results are shown in Table 1). We note that although there is a slight delay in the average time-to-peak of higher fields compared with 1.5T, this lies comfortably within the variance between simulations, and does not therefore make a strong contribution to variance in fits at different field strengths. Likewise, the duration of the HRF does not vary significantly across field strengths. This leads us to conclude that if the shape of the HRF does indeed play a role in goodness of DCM fits, it is in the more accurate representation of HRF undershoot at higher field strengths. We note, however, that there are still many questions regarding optimal fitting of the HRF at different field strengths that are left to a more complete discussion in future work.

	1.5T	3T	7T	9.4T
Time to peak (ms)	6023 (506)	6235 (390)	6180 (476)	6223 (575)
Width at half peak (ms)	6323 (1091)	6480 (1091)	6283 (591)	6289 (698)

Table 1. HRF characteristics at varied field strengths.

fMRI pre-processing methods.

As we are validating our pipeline against spectral DCM (2), we use a sample subject from the FC1000 project (available at www.fil.ion.ucl.ac.uk/spm/data/spDCM/) - one of the group used by Friston et al. (2) in their initial validation of spectral DCM - to demonstrate the coherence between SPM and Neuroblox implementations of DCM. Preprocessed data was available from this prior work, so this was the data used in the current study. Briefly, this scan was acquired at the University of Oxford Centre for Clinical Magnetic Resonance Research using a 3T Siemens scanner. Whole-brain fMRI was acquired using a gradient-echo EPI sequence (Repetition Time (T_R) = 28ms, Echo Time (T_E) = 28ms, flip angle = 89,° field of view = 224mm, voxel size 3mm x 3mm x 3.5mm, 180 volumes acquired). This scan was accompanied by a T1-weigheted structural MRI (T_R = 2040ms, T_E = 4.7ms, flip angle = 8°, field of view = 192mm, voxel size = 1mm x 1mm x 1mm) to provide tissue segmentation. The fMRI scan was preprocessed by removing the first five volumes, realigning to the reference scan, normalizing to MNI space, correcting for slice-timing, and spatially smoothing the data with a Gaussian kernel (6mm FWHM). A GLM containing movement regressors was used to remove confounds, along with confound GLMs fit from the ventricle (CSF) and white matter signals (see SPM manual for details). Four ROIs were selected from the default mode network (DMN); all were chosen as 8mm radius spheres centered in each location. The four regions are the posterior cingulate cortex (PCC; center [0, -52, 26]), medial prefrontal cortex (mPFC; [3, 54, -2]), left intraparietal cortex (LIPC; [-50, -63, 32]) and right intraparietal cortex (RIPC; [48, -69, 35]).

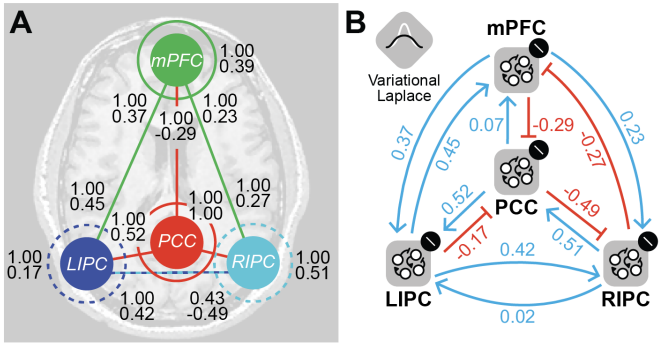

Fig. 5. Network connectivity fit to human fMRI data by SPM12 and Julia show identical results. A. 3D locations of the four regions of the default mode network (PCC, mPFC, LIPC, and RIPC). Visualization is done in SPM12 after using standard fitting procedures used in (2) B. Visualization of the DMN connections as fit by our variational Laplace approach in Julia. Solid lines represent positive connections, dashed lines represent negative connections. We note that the connections are identical to those found in A.

Comparison of SPM12 and Julia DCM fitting procedures on fMRI data.

The DCM was fit in SPM12 using a standard pipeline. A fully connected (i.e., all initial connection weights set to 1) DCM with no exogenous inputs was specified using the default parameter values (sequential data, $T_E = 0.04$, no modulatory effects, one state per region, no stochastic effects, no centering of input), then inverted using spectral DCM. Additionally, we probed the effects of field strength correction on the results by fitting at three different field strengths (1.5T, 3T, and 7T), presented in Table 2. Each field strength yielded significantly different results than the other, but all fits agreed in the directions of inter-regional connections, with the exception of LIPC intraregional connectivity (reversed at 1.5T). The locations of the DMN ROIs selected for this analysis are shown in Fig. 5A along with the connections fit by SPM12, and the fit network given by our method is shown in Fig. 5B.

Influence of field strength on hemodynamic response function.

As field strength noticeably impacts the shape of the HRF, and can potentially impact model fitting as a consequence, we fit the same fMRI data presented in the previous section at three different field strengths: 1.5T, 3T, and 7T. The results of these fits are presented in Table 2 and visualized in Fig. 6. First, we note that the results are significantly different across field strengths, with the variance of fit connection weights more than an order of magnitude smaller than the parameters themselves. Second, we note that these associations are not simple linear multiples based on the field strength but significantly impact the relative weights of the associations. For example, PCC $\rightarrow$ mPFC estimated connection strength modestly decreases when fit at 3T compared to 1.5T (0.09 $\rightarrow$ 0.07), but greatly increases when fit at 7T compared to 3T (0.07 $\rightarrow$ 0.15). Finally, we note that although these results are significantly different, when examining modern field strengths (3T and 7T), all directions of associations are the same (i.e., the signs of all connection weights agree), and in the case of 1.5T only one parameter (LIPC

↔ LIPC) has a different sign. Thus, while fitting 7T data using 3T parameters is suboptimal, this approach will likely yield directionally accurate results, even if the relative weights are suboptimal. Taking these results together, we believe there is a compelling case that field strength should be accounted for when fitting fMRI data to provide the most accurate results.

1.5T	PCC	mPFC	LIPC	RIPC
PCC	0.37 (0.092)	0.09 (0.087)	0.50 (0.072)	-0.45 (0.078)
mPFC	-0.33 (0.087)	0.37 (0.092)	0.48 (0.071)	0.10 (0.081)
LIPC	-0.15 (0.061)	0.31 (0.059)	-0.12 (0.098)	0.37 (0.056)
RIPC	0.45 (0.067)	-0.27 (0.082)	0.09 (0.066)	0.03 (0.091)
3T	PCC	mPFC	LIPC	RIPC
PCC	0.43 (0.086)	0.07 (0.088)	0.52 (0.069)	-0.49 (0.076)
mPFC	-0.29 (0.083)	0.39 (0.087)	0.37 (0.066)	0.23 (0.076)
LIPC	-0.17 (0.068)	0.45 (0.067)	0.09 (0.089)	0.42 (0.060)
RIPC	0.51 (0.065)	-0.27 (0.081)	0.02 (0.063)	0.03 (0.091)
7T	PCC	mPFC	LIPC	RIPC
PCC	0.76 (0.077)	0.15 (0.091)	0.54 (0.070)	-0.29 (0.087)
mPFC	-0.11 (0.088)	0.67 (0.077)	0.42 (0.069)	0.14 (0.084)
LIPC	-0.12 (0.070)	0.37 (0.070)	0.18 (0.085)	0.43 (0.067)
RIPC	0.32 (0.079)	-0.12 (0.085)	0.13 (0.068)	0.49 (0.073)

Table 2. Fitting DCM from 3T rs-fMRI yields different results across field strengths. Results are presented as DCM fit (covariance).

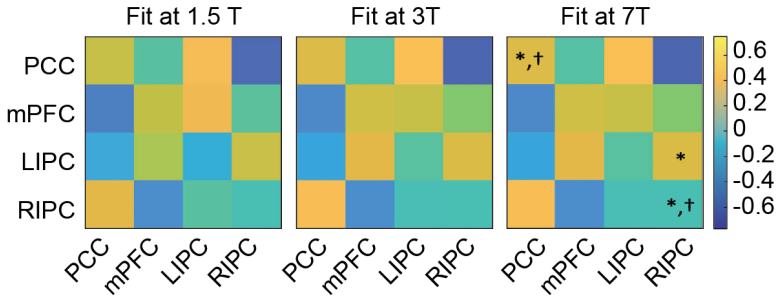

Fig. 6. Visualization of connectivity strengths fit at different HRF field strength corrections. *Indicates significantly different than 3T fit. †Indicates significantly different than 1.5T fit. It is important to note that while the direction of all fits is the same in almost all cases (the exception being the LIPC-LIPC connection at 1.5T), there are more differences in connectivity strengths, particularly noticeable at 7T, the current state of the art for clinical neuroimaging.

Discussion

DCM is one of the most widely used and validated methods of extracting underlying neural connectivity from fMRI data (5, 10–14, 17). The aim of the work presented here is two-fold: to improve the computational efficiency and accessibility of the well-established spectral DCM framework provided by SPM12 (2), and increase the application domain by providing a modular framework for model-fitting, thus empowering researchers to tailor their models to specific experimental setups and research questions. We demonstrate our implementation offers up to an order of magnitude improvement in computational efficiency in our implementation compared to the standard SPM12 DCM, an enhancement that substantially accelerates computations and enables efficient analysis of large-scale datasets and complex models. Additionally, thanks to the modular design of our approach, this speed comes with built-in flexibility to specify different fitting regimes not allowed by the standard SPM12 implementation.

Our work builds upon spectral DCM by re-implementing it in Julia, a high-performance programming language (24, 33). By implementing the variational Laplace approach presented here in Julia, we provide a method that recovers identical connectivity values - in both simulated and human fMRI data - as the SPM12 DCM approach while reducing computation time significantly. While SPM12 is implemented as open-source and easily accessible code, it relies upon Matlab, which inherently limits accessibility (to those who have a Matlab license) and computational speed (a known issue of Matlab code - see (24)). Julia, on the other hand, has been built as open-source from the ground up, with a specific focus on optimizing for scientific computation (33, 40–44). Even when fully parallelized Julia code offers order-of-magnitude speed benefits over similar Matlab implementations (24, 43). Although the translation of code into Julia is already sufficient to see a significant speed increase, we further optimize the approach by employing automatic differentiation (32). The adoption of automatic differentiation is also set to improve accuracy results for more complex non-linear models as presented in this study (45).

A limitation of standard DCM implementation is that all but the most technically savvy users who feel comfortable editing the base code are constrained to the modeling choices imposed by the software. These pre-specified models are a linear neural mass (attached to a Balloon model for fMRI fitting (2)) or a few varieties of canonical microcircuits (for fitting LFP recordings (6)). In our implementation, we provide these standard approaches, but allow the user freedom to choose a variety of other neural mass models during fitting, or to specify their own choices. This is again made possible by implementation in

Julia, specifically through the symbolic expressions offered in ModelingToolkit (25). As ModelingToolkit allows for symbolic expressions of neural masses, the process can be as simple as the Jansen-Rit example in Fig. 1A.

To illustrate the utility of a flexible modular approach, we provide a method to improve correction for MRI scanner field strengths when fitting a model to real data. Given the ubiquity and power of DCM approaches, it is necessary to consider advances in MRI scanner technology that have happened since the initial implementations of DCM, which used then state-of-the-art 1.5T and 3T scanners (1, 2). Here, we demonstrate how accounting for higher field strengths (e.g., 7T) that have become more common in human neuroimaging studies is an important consideration, as the results of fitting with the wrong field strength yield significantly different results. We note that the directionality of inter-regional connections is largely stable to this field strength correction (i.e., network topology does not change although the connection strengths do), giving us confidence that prior studies using this technique are still valid. However, given the noticeably different magnitudes of effective connectivity strengths, and the fact that this metric is used in many studies as the outcome of interest, it is important to apply the field strength of the original data when fitting a DCM, and we provide that capability in the package presented here.

The optimizations we provide here are only the beginning of future work in advanced neural modeling - indeed, this implementation is specifically built to encourage future development and open-sourced optimization. Some potential approaches to build on this include the substitution of alternatives the variational Laplace optimization, such as the ADVI (46) and NUTS samplers (47) offered through Turing.jl (48). By being built within the Julia ecosystem, the method we present here is therefore poised for extensibility to accelerate the next generation of neural modeling.

Accessing Code/Tools. All methods shown here, which include an easy-to-use GUI as well as documentation and tutorials, are made freely available through the Neuroblox computational neuroscience platform (neuroblox.org). Neuroblox can be accessed through the JuliaHub repository (juliahub.com). The Julia code-base without GUI is available in this repository: https://github.com/Neuroblox/Spectral-Dynamic-Causal-Modeling.git

Author Contributions. Conceptualization: D.H., L.M.P. and H.H.S.; Formal analysis: D.H. and A.C.; Funding acquisition: C.R., A.E., L.M.P. and H.H.S.; Investigation: D.H.; Methodology: D.H.; Project administration: L.M.P.; Software: D.H., C.R. and H.H.S.; Supervision: C.R., K.J.F. and H.H.S.; Visualization: D.H. and A.C.; Writing – original draft: D.H. and A.C.; Writing - review & editing: D.H., A.C., L.M.P., K.J.F. and H.H.S.

Competing Interests. The authors declare the following competing interests: authors C.R., L.M.P., A.E., and H.H.S. are co-founders of Neuroblox Inc., a company spun out of SUNYSB, MIT, and Dartmouth to develop a commercial-grade software platform for multi-scale computational neuroscience with applications to diagnosis and treatment of brain-based disorders.

ACKNOWLEDGMENTS. We thank Annabel Driussi for her valuable support with graphics. The research presented here was funded by the Baszucki Brain Research Fund, United States (LRMP). KJF is supported by funding for the Wellcome Centre for Human Neuroimaging (205103/Z/16/Z). AGC acknowledges the NIHGM MSTP Training Award, United States (T32-GM008444).

References

1. KE Stephan, et al., Nonlinear Dynamic Causal Models for fMRI. *NeuroImage* **42**, 649–662 (2008).
2. KJ Friston, J Kahan, B Biswal, A Razi, A DCM for resting state fMRI. *NeuroImage* **94**, 396–407 (2014).
3. SM Lee, R Tibon, P Zeidman, PS Yadav, R Henson, Effects of face repetition on ventral visual stream connectivity using dynamic causal modelling of fMRI data. *NeuroImage* **264**, 119708 (2022).
4. A Hahn, et al., Reconfiguration of functional brain networks and metabolic cost converge during task performance. *eLife* **9**, e52443 (2020) Publisher: eLife Sciences Publications, Ltd.
5. RJ Moran, et al., Alterations in Brain Connectivity Underlying Beta Oscillations in Parkinsonism. *PLoS Comput. Biol.* **7**, e1002124 (2011).
6. RJ Moran, et al., Dynamic Causal Models and Physiological Inference: A Validation Study Using Isoflurane Anaesthesia in Rodents. *PLoS ONE* **6**, e22790 (2011).
7. KJ Friston, AM Bastos, D Pinotsis, V Litvak, LFP and oscillations—what do they tell us? *Curr. Opin. Neurobiol.* **31**, 1–6 (2015).
8. O David, et al., Dynamic causal modeling of evoked responses in EEG and MEG. *NeuroImage* **30**, 1255–1272 (2006).
9. R Sladky, et al., Disrupted Effective Connectivity Between the Amygdala and Orbitofrontal Cortex in Social Anxiety Disorder During Emotion Discrimination Revealed by Dynamic Causal Modeling for fMRI. *Cereb. Cortex* **25**, 895–903 (2015).
10. ML Seghier, KJ Friston, Network discovery with large DCMs. *NeuroImage* **68**, 181–191 (2013).
11. D Bernal-Casas, HJ Lee, AJ Weitz, JH Lee, Studying brain circuit function with dynamic causal modeling for optogenetic fMRI. *Neuron* **93**, 522–532.e5 (2017).
12. J Cha, et al., Clinically Anxious Individuals Show Disrupted Feedback between Inferior Frontal Gyrus and Prefrontal-Limbic Control Circuit. *J. Neurosci.* **36**, 4708–4718 (2016) Publisher: Society for Neuroscience Section: Articles.
13. S Rupprechter, et al., Blunted medial prefrontal cortico-limbic reward-related effective connectivity and depression. *Brain* **143**, 1946–1956 (2020).
14. M Tik, et al., Ultra-high-field fMRI insights on insight: Neural correlates of the Aha!-moment. *Hum. Brain Mapp.* **39**, 3241–3252 (2018) _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/hbm.24073.
15. Q Bukhari, SF Ruf, X Guell, S Whitfield-Gabrieli, S Anteraper, Interaction Between Cerebellum and Cerebral Cortex, Evidence from Dynamic Causal Modeling. *The Cerebellum* **21**, 225–233 (2022).
16. R Sladky, et al., Dynamic Causal Modeling of the Prefrontal/Amygdala Network During Processing of Emotional Faces. *Brain Connect.* **12**, 670–682 (2022) Publisher: Mary Ann Liebert, Inc., publishers.
17. TS Zarghami, P Zeidman, A Razi, F Bahrami, GA Hossein-Zadeh, Dysconnection and cognition in schizophrenia: A spectral dynamic causal modeling study. *Hum. Brain Mapp.* **44**, 2873–2896 (2023) _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/hbm.26251.
18. T Parr, et al., Dynamic causal modelling of immune heterogeneity. *Sci. Reports* **11**, 11400 (2021).
19. WD Penny, KE Stephan, A Mechelli, KJ Friston, Modelling functional integration: a comparison of structural equation and dynamic causal models. *NeuroImage* **23 Suppl 1**, S264–274 (2004).
20. KE Stephan, N Weiskopf, PM Drysdale, PA Robinson, KJ Friston, Comparing hemodynamic models with DCM. *NeuroImage* **38**, 387–401 (2007).
21. K Friston, J Mattout, N Trujillo-Barreto, J Ashburner, W Penny, Variational free energy and the Laplace approximation. *NeuroImage* **34**, 220–234 (2007).
22. A Razi, et al., Large-scale DCMs for resting-state fMRI. *Netw. Neurosci.* **1**, 222–241 (2017).
23. S Marek, et al., Reproducible brain-wide association studies require thousands of individuals. *Nature* **603**, 654–660 (2022).
24. C Rackauckas, ODE Solver Multi-Language Wrapper Package Work-Precision Benchmarks (MATLAB, SciPy, Julia, deSolve (R)) (2023).
25. Y Ma, et al., ModelingToolkit: A Composable Graph Transformation System for Equation-Based Modeling. *arXiv* **2103.05244** (2021).
26. RB Buxton, K Uludağ, DJ Dubowitz, TT Liu, Modeling the hemodynamic response to brain activation. *NeuroImage* **23**, S220–S233 (2004).
27. P Zeidman, K Friston, T Parr, A primer on Variational Laplace (2022).
28. P Sidén, *Scalable Bayesian spatial analysis with Gaussian Markov random fields.* (2020).
29. S Gowda, et al., High-performance symbolic-numerics via multiple dispatch (2022) arXiv:2105.03949 [cs].

30. R Moran, D Pinotsis, K Friston, Neural masses and fields in dynamic causal modeling. *Front. Comput. Neurosci.* **7**, 57 (2013).
31. AM Bastos, et al., Canonical Microcircuits for Predictive Coding. *Neuron* **76**, 695–711 (2012).
32. J Revels, M Lubin, T Papamarkou, Forward-Mode Automatic Differentiation in Julia (2016) arXiv:1607.07892 [cs].
33. Y Ma, V Dixit, M Innes, X Guo, C Rackauckas, A Comparison of Automatic Differentiation and Continuous Sensitivity Analysis for Derivatives of Differential Equation Solutions. *arXiv:1812.01892 [cs]* (2021) arXiv: 1812.01892.
34. N Van der Aa, H Ter Morsche, R Mattheij, Computation of eigenvalue and eigenvector derivatives for a general complex-valued eigensystem. *The Electron. J. Linear Algebr.* **16** (2007).
35. RM Lin, JE Mottershead, TY Ng, A state-of-the-art review on theory and engineering applications of eigenvalue and eigenvector derivatives. *Mech. Syst. Signal Process.* **138**, 106536 (2020).
36. T Thummerer, L Mikelsons, Eigen-informed NeuralODEs: Dealing with stability and convergence issues of NeuralODEs (2023) arXiv:2302.10892 [cs].
37. KJ Friston, L Harrison, W Penny, Dynamic causal modelling. *NeuroImage* **19**, 1273–1302 (2003).
38. T Obata, et al., Discrepancies between BOLD and flow dynamics in primary and supplementary motor areas: application of the balloon model to the interpretation of BOLD transients. *NeuroImage* **21**, 144–153 (2004).
39. RB Buxton, The physics of functional magnetic resonance imaging (fMRI). *Reports on progress physics. Phys. Soc. (Great Britain)* **76**, 096601 (2013).
40. J Bezanson, et al., Julia: dynamism and performance reconciled by design. *Proc. ACM on Program. Lang.* **2**, 120:1–120:23 (2018).
41. S Krämer, D Plankensteiner, L Ostermann, H Ritsch, QuantumOptics.jl: A Julia framework for simulating open quantum systems. *Comput. Phys. Commun.* **227**, 109–116 (2018).
42. R Huang, W Xu, Y Wang, S Liverani, AE Stapleton, Performance Comparison of Julia Distributed Implementations of Dirichlet Process Mixture Models in *2019 IEEE International Conference on Big Data (Big Data)*. pp. 3350–3354 (2019).
43. F Danford, E Welch, J Cárdenas-Ródriguez, MM Strout, Analyzing Parallel Programming Models for Magnetic Resonance Imaging in *Languages and Compilers for Parallel Computing*, Lecture Notes in Computer Science, eds. C Ding, J Criswell, P Wu. (Springer International Publishing, Cham), pp. 188–202 (2017).
44. J Freeman, Open source tools for large-scale neuroscience. *Curr. Opin. Neurobiol.* **32**, 156–163 (2015).
45. A Griewank, A Walther, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. (Society for Industrial and Applied Mathematics, Philadelphia, PA), 2nd edition edition, (2008).
46. A Kucukelbir, D Tran, R Ranganath, A Gelman, DM Blei, Automatic Differentiation Variational Inference, Technical report (2016).
47. MD Homan, A Gelman, The No-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *The J. Mach. Learn. Res.* **15**, 1593–1623 (2014).
48. H Ge, K Xu, Z Ghahramani, Turing: a language for flexible probabilistic inference. *Proc. 21st Int. Conf. on Artifi- cial Intell. Stat. (AISTATS) 2018, Lanzarote, Spain. PMLR* **84** (2018).