



A fast dynamic causal modeling regression method for fMRI

Haifeng Wu^{a,b,c,*}, Xinhang Hu^a, Yu Zeng^{a,b,c}

^a School of Electrical and Information Engineering, Yunnan Minzu University, Kunming, 650500, China

^b Yunnan Provincial Key Laboratory of Unmanned Autonomous Systems, Kunming, 650500, China

^c Yunnan Provincial Colleges and Universities Intelligent Sensor Network and Information System Technology Innovation Team, Kunming 650504, China

ARTICLE INFO

Keywords:

GLM
DCM
Sparse
Effective connectivity
Computational complexity

ABSTRACT

Dynamic Causal Modeling (DCM) is a crucial tool for studying brain effective connectivity, offering valuable insights into brain network dynamics through functional magnetic resonance imaging (fMRI) and electrophysiology (EEG and MEG). However, its high computational complexity limits its applicability in large-scale network analysis. To address this issue, we propose a regression algorithm that integrates the Generalized Linear Model (GLM) with Sparse DCM, termed GSD. This algorithm enhances computational performance through three key optimizations: (1) utilizing the symmetry of the Fourier transform to convert complex frequency domain calculations into real number operations, thereby reducing computational complexity; (2) applying GLM and filtering techniques to minimize the effects of noise and confounds, enhancing parameter estimation accuracy; and (3) defining a new cost function to optimize variational inference and filter parameters, further improving parameter estimation accuracy. We validated the GSD algorithm using three public fMRI datasets: simulated Smith small-world network data, attention and motion measured data, and face recognition repetition effect measured data. The experimental results demonstrate that the GSD algorithm reduces computation time by over 50 % while maintaining parameter estimation performance comparable to traditional methods. These findings offer a new perspective on balancing model interpretability and computational efficiency, potentially broadening the application of DCM across various fields.

1. Introduction

The human brain is a dynamic system composed of interconnected regions, exhibiting structural connectivity, functional connectivity, and effective connectivity (Friston, 2011). Studying these interconnections is crucial for understanding brain mechanisms. Unlike structural and functional connectivity, effective connectivity provides more directional and mechanistic information. It not only describes correlations between brain regions but also reveals the causal relationships and information flow or message passing underlying these correlations.

With the continuous development of brain imaging technology, several methods have emerged to study effective connectivity using functional magnetic resonance imaging (fMRI). These methods include structural equation modeling (SEM) (McIntosh and Gonzalez-Lima, 1994), multivariate autoregressive modeling (MAR) (Harrison et al., 2003), Granger causal modeling (GCM) (Friston et al., 2013), and dynamic causal modeling (DCM) (Kiebel et al., 2008; Bönstrup et al., 2016; Klingner et al., 2015; Friston et al., 2003; Wang et al., 2019; Frank et al., 2001). Among these, GCM and DCM are the most representative.

GCM is a relatively simple and intuitive statistical method that estimates causal relationships between time series (observational data). However, it is very sensitive to noise and confound in the data, which may lead to misleading causal inferences. DCM, based on dynamic system theory and state space models (Friston, 2002; Hamilton, 1994), offers a more comprehensive consideration of the dynamic coordination within brain networks. It explains brain activity based on prior neuroscience knowledge, providing richer information. This makes DCM the mainstream method for studying effective connectivity, and numerous new methods have been developed in combination with it (Rigoux and Daunizeau, 2015; Ostwald and Starke, 2016; Liang et al., 2022; OU et al., 2022; Shi and Li, 2024; K J FRISTON et al., 2002; K J FRISTON et al., 2002).

Despite its advantages, DCM's computational cost is relatively high. Firstly, DCM uses complex neural dynamic models, such as neural mass models (David O, 2003) and neural population models (Zetterberg et al., 1978), to describe the dynamic coordination between brain regions. These models are typically based on differential or difference equations and require numerical methods for solving. Secondly, DCM employs

* Corresponding author.

E-mail address: whf5469@gmail.com (H. Wu).

<https://doi.org/10.1016/j.neuroimage.2024.120954>

Received 10 August 2024; Received in revised form 20 November 2024; Accepted 21 November 2024

Available online 30 November 2024

1053-8119/© 2024 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY-NC license (<http://creativecommons.org/licenses/by-nc/4.0/>).

state space models (Stephan and Roebroeck, 2012) to link the neural dynamic model with actual observation data, accurately describing the relationship between the observed data and hidden neural activity. Solving this mathematical model demands substantial computational resources and time. Lastly, the core of DCM involves fitting actual observation data by estimating parameters of the brain network model, often through complex optimization problems like the expectation-maximization (EM) algorithm and variational Laplace (VL) (Moon, 1996), which requires multiple iterations to converge. The computational limitations make model inversion costly, especially for complex models, restricting DCM's application to large-scale networks with more brain regions.

To improve DCM's computational speed and reduce its cost, various new variants have been proposed. One approach simplifies the model, such as by simplifying the neural dynamics model or considering certain biological processes or connections as constants or fixed values, thereby reducing the number of parameters to estimate (Frässle et al., 2017; Prando et al., 2020; Stephan et al., 2008). Another approach simplifies the model structure by limiting connectivity types or topological structures to reduce parameter estimation (Seghier and Friston, 2013). However, these simplified models may lose some biological details, reducing the model's interpretability and predicted validity. They may also struggle to capture the dynamic processes of complex brain networks, thus not fully meeting researchers' needs. Another variant improves DCM's computational speed by enhancing parameter estimation and optimizing computational methods, such as through approximate inference (Frässle et al., 2018; Sahoo et al., 2019; Frässle et al., 2021) and transform domain methods (Razi et al., 2017; Novelli et al., 2024; Friston K et al., 2014; Sato et al., 2009). While these methods progress in improving computational speed, they often require assumptions about the model, potentially affecting accuracy and reliability. Additionally, data-driven approaches use machine learning techniques to automatically learn and extract features and dynamics of brain networks from large amounts of observational data (Sotero and Sanchez-Bornot, 2024; Zhou et al., 2024). Compared to traditional models based on prior biological knowledge, data-driven methods better adapt to different subjects' brain network characteristics and conditions, improving model applicability and predicted validity. However, these methods require expensive labeled data.

From the above, DCM's interpretability and complexity are often in conflict. Methods with strong interpretability tend to be highly complex, while those with lower complexity offer faster computation but weaker interpretability. To address these issues, we propose a sparse DCM algorithm combined with the Generalized Linear Model (GLM) (Worsley and Friston, 1995; Poline and Brett, 2012), termed the combined Generalized Linear Model and sparse DCM (GSD) algorithm. This algorithm aims to balance model interpretability and computational complexity, ensuring a certain level of complexity while achieving faster computation. The contributions of this paper are as follows:

- Utilizing the symmetric characteristics of the Fourier transform to convert complex frequency domain calculations into real number calculations, reducing computational complexity.
- Applying GLM and filtering techniques to eliminate some observed signal noise, improving parameter estimation accuracy.
- Using a defined cost function to optimize variational inference parameters and filter parameters, further enhancing parameter estimation accuracy.

In summary, we introduce a novel and efficient (GSD) scheme for characterizing effective connectivity using DCM. The key conceptual advance is to include data feature selection as part of variational model inversion. In more detail, we first commit to an efficient kind of DCM known as regression DCM (rDCM) (Frässle et al., 2017; Lin Tiger et al., 2020). In brief, rDCM replaces the inversion of a multivariate timeseries by estimating the incoming effective (afferent) connectivity to each

node (i.e., region of interest) one node at a time, using the observed responses of the remaining nodes as driving or modulatory inputs. This allows us to use a simple bilinear model — upon which DCM for fMRI is based — in which the effective connectivity maps from inputs to the first time derivative of the target node. To make this scheme more efficient, we reformulate the (bilinear) model using a Fourier transform. This means that Fourier transformed data can be reduced in size by selecting appropriate frequencies. Crucially, this data selection — which corresponds to filtering in frequency space — can be optimised using the principles of optimal experimental design (Lindley, 1956). Namely, selecting frequencies that reduce uncertainty or maximise information gain, under the current posterior estimates of the generative model. We approximate the expected information gain of low-pass, high-pass and bandpass filtering (specified by hyperparameters) with an objective function that replaces the observed data with data generated by the model. This provides an inversion scheme that complies with the dual principles of Bayes optimality; namely, maximizing the evidence or marginal likelihood of the generative model, while maximizing the expected information gain afforded by data-selection.¹ This scheme is equipped with two further devices to reduce computational complexity. First, an elementary transform is used to finesse operations on complex numbers, leveraging the symmetry of the Fourier transform (of real valued timeseries). Finally, during the variational inversion of each bilinear regression model, we introduce sparsity priors based upon the Bernoulli distribution. This enables the elimination of connections, such that the scheme only has to estimate connectivity from the Markov blanket of each node. In standard DCM schemes, this sparsity prior would normally be implemented after model inversion using Bayesian model reduction. However, by incorporating the sparsity constraints explicitly during model inversion, one can progressively reduce computational load as certain parameters (i.e., effective connections) are removed. Together, GSD affords an extremely efficient form of complex system modelling for sparsely coupled nodes, under mild prior constraints on the dynamics. We validated the algorithm using three datasets: simulated Smith small-world network data (Smith et al., 2011), measured attention and motion data (Büchel and Friston, 1991; Friston K and Frith, 1995), and measured face recognition repetition effect data (Wakeman and Henson, 2015; Lee et al., 2022; Ewbank et al., 2013). The experimental results show that the GSD algorithm reduces computation time by at least 50 % for small problems, increasing to one or two orders of magnitude as the size of the problem scales, while maintaining similar parameter estimation performance and improving precision and accuracy metrics.

2. Related work

DCM is a prominent method for studying brain effective connectivity. First introduced by Friston et al. in 2003 (Friston et al., 2003), DCM is based on Bayesian theory and dynamic system modeling, aimed at estimating causal relationships between brain regions in neural networks. Initially applied to fMRI data, it was later extended to electroencephalogram (EEG) and magnetoencephalogram (MEG) data (Kiebel et al., 2008; Bönstrup et al., 2016; Klingner et al., 2015). By establishing a neural dynamics model, DCM describes the dynamic coordination of brain activity, revealing functional connectivity patterns and modulation mechanisms within brain networks. Continuous advances in brain imaging technology and computational methods have led to ongoing improvements in DCM (Frässle et al., 2017) (Frässle et al., 2018)

¹ It is in terms of efficient evidence accumulation. In other words, instead of trying to ingest all available data in parallel, it can be computationally more efficient to assimilate small amounts of informative data, much like saccadic eye movements select the salient parts of the visual scene (the analogy here is that there are salient regions in frequency space that speak to a Bayes optimal filtering).

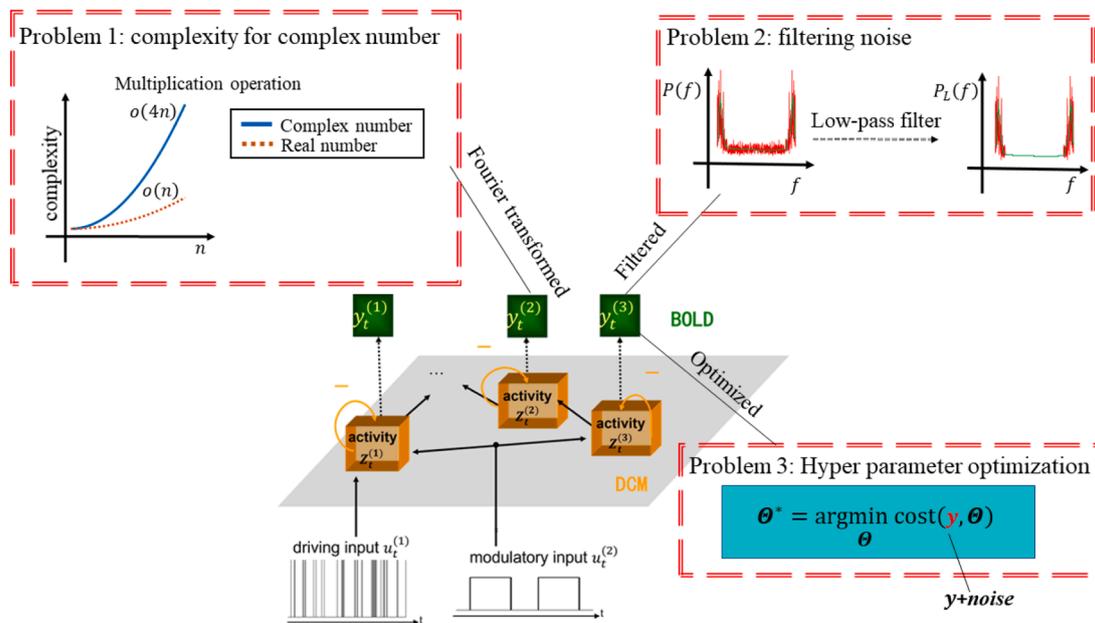


Fig. 1. Problem statement in this paper.

(Novelli et al., 2024; Friston et al., 2014).

One of the early advances in DCM involved transforming causal connectivity parameters into the frequency domain, such as with the partial directional coherence (PDC) algorithm (Sato et al., 2009). This method enhanced calculation speed by simplifying the complex neural dynamics model and model regression, enabling the decomposition of multivariate partial correlation from the multivariate autoregressive model. Researchers have also made nonlinear extensions to DCM (Stephan et al., 2008) to simulate gating processes at the neural soma level, essential for controlling conduction gain between neuronal populations. Moreover, combining DCM with other technologies has expanded its application in neuroscience. For example, behavioral dynamic causal modeling (bDCM) (Rigoux and Daunizeau, 2015) analyzes the brain's transformation of stimuli into behavioral outcomes, examining the contributions of brain regions and their connections in generating behavior. Another example is combining DCM with transfer entropy (Shi and Li, 2024), which uses asymmetry detection of transfer brain entropy to quantify differences in predictive information between forward and backward time series, establishing a more accurate causal relationship detection standard. DCM has also been applied to event-related potentials (ERP) recorded by electroencephalograms (OSTWALD and STARKE, 2016), proposing a probabilistic latent variable delayed differential equation model to improve ERP-DCM's technical accessibility. Model selection theory is another crucial aspect of DCM. Efficiently searching for the target model from the model space is vital for model selection, with the greedy algorithm (Ou et al., 2022) being a common search strategy that lowers the threshold for exploratory DCM analysis. Additionally, model selection should implement inference at the group level. Hierarchical applications (Liang et al., 2022) (Friston et al., 2002; Friston et al., 2002) of DCM have become current; for example, the use of Parametric Empirical Bayes (PEB) to test hypotheses about differences between groups or, hierarchical structure planning in which DCM's can be hierarchically composed.

Reducing DCM's computational complexity has always been a key goal, aiming to enhance computational speed and enable causal relationship inference in large-scale brain networks. Simplifying the model is a common approach to reducing complexity, such as by decreasing the number of parameters to estimate. For instance, rDCM (Frässle et al., 2017) ignores the bilinear effect in DCM, significantly reducing the parameters to estimate. Another approach treats DCM as a sparse

regression model (Seghier and Friston, 2013; Frässle et al., 2018) or a low-dimensional multivariate regression model (Sahoo et al., 2019). The former uses variational inference technology to accelerate operations, like in sparse regression DCM (sparse rDCM), while the latter reduces data dimensionality in space and time, combining optimization strategies like extrapolation and gradient descent. The algorithms effectively capture valid DCM parameters between brain regions by reducing redundant estimation parameters. Simplifying the model structure itself is another approach, such as transforming continuous neural dynamics models into discrete models and using a linear, regionalized hemodynamic response function for hemodynamic models (Prando et al., 2020).

However, simplifying models to reduce computational complexity can compromise interpretability. To avoid this, optimizing computational methods and improving efficiency can achieve fast computation without sacrificing interpretability. A notable method is combining spectral DCM with resting-state fMRI (rs-fMRI) technology (Razi et al., 2017; Novelli et al., 2024; Friston K et al., 2014). Spectral DCM inherently improves computational speed, and its combination with rs-fMRI omits the estimation of hidden neural states, significantly enhancing algorithm efficiency. Recently, the rapid development of machine learning has introduced ways to reduce running time by combining DCM with machine learning techniques. For instance, physical neural networks, graph neural networks, and long short-term memory networks (Sotero and Sanchez-Bornot, 2024; Zhou et al., 2024) have been used to infer causal connectivity parameters. While combining DCM with machine learning does not directly reduce running time due to the extensive training required, especially with large data volumes, the output process is swift once the model is trained.

3. Problem statement

This paper primarily focuses on the inversion of DCM in the frequency domain. The key advantage of frequency domain DCM is that it does not require solving the hidden states of neurons (Razi et al., 2017; Novelli et al., 2024; Friston et al., 2014), simplifying the calculation and speeding up computations. Additionally, in the frequency domain, it is easier to implement filtering such as low-pass, high-pass, and band-pass filters. By setting the response value of the filtered signal band to zero in the frequency domain filter, an ideal filtering function can be achieved. The goal of this paper is to enhance the calculation speed of DCM while

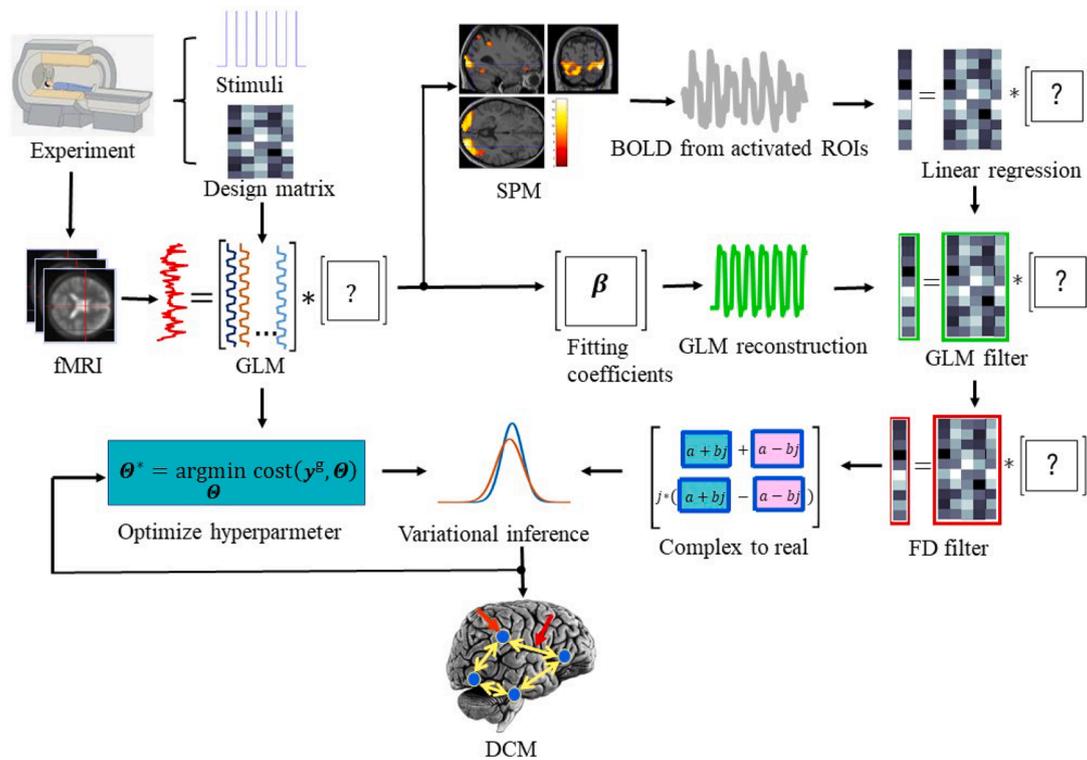


Fig. 2. Flowchart of the proposed method.

maintaining its regression performance. To achieve this, we will address several issues with frequency domain DCM, as illustrated in Fig. 1.

Firstly, although spectrum calculations simplify the model computation process, a real signal becomes a complex signal after a Fourier transform. Complex number operations typically increase computational complexity compared to real number operations. For example, multiplying two complex numbers requires four multiplications (one for each real or complex part of the numbers), resulting in $O(4n)$ complexity if repeated n times. In contrast, multiplying two real numbers only requires n multiplications, resulting in $O(n)$ complexity. Therefore, the first issue this paper addresses is whether the additional computational complexity introduced by converting real numbers to complex numbers can be reduced after applying the Fourier transform.

Secondly, after transforming DCM into the frequency domain, noise reduction processing is necessary to improve efficiency. Common methods involve using bandpass or bandstop filters (Wang et al., 2019). In theory, effective noise reduction requires the noise frequency to be in the stopband and the signal frequency to be in the passband. However, if both the noise and signal are broadband signals with overlapping frequencies, the noise reduction performance of the filter is compromised. Existing studies (Frank et al., 2001) confirm that fMRI noise originates not only from low-frequency heartbeats and breathing but also from other high-frequency signals. Thus, the second issue this paper addresses is whether to rely solely on filters for noise reduction or to consider combining them with other noise reduction techniques.

Lastly, calculating DCM in the frequency domain involves setting hyperparameters, including initial values, prior information, and parameters for noise reduction. Proper hyperparameters can significantly improve DCM efficiency. One approach to setting hyperparameters is using empirical values, but this method faces uncertainty under changing environments or conditions. Another approach is using an objective function, such as minimizing error, maximizing a posteriori probability, or maximizing negative free energy (Friston and Mattout, 2007). This method aims to ensure the model state aligns with the observed values when hyperparameters are optimal. However, observed values often contain noise, potentially causing hyperparameters to

match the noise rather than the true signal, thereby reducing efficiency. Therefore, the third issue this paper addresses is how to design a robust objective function to obtain optimal hyperparameter values.

4. Algorithm

In this section, we introduce the various modules of the proposed algorithm, with its flowchart shown in Fig. 2. First, we obtain the corresponding stimulation signals based on the experimental design and construct the design matrix. We then use the GLM algorithm and Statistical Parametric Mapping (SPM) software to accurately identify activated brain regions and extract the BOLD signal from the regions of interest (ROIs). Since neuronal activity in the original DCM is an unobservable hidden state, we transform the original model into a linear regression equation in the frequency domain, where we replace the inversion of a multivariate timeseries by estimating the incoming effective connectivity to each ROI at a time, and select appropriate frequencies for regression (such as frequency-domain filter later). This transformation avoids the need to estimate the hidden neuronal state. To further reduce noise interference in the observed signal, we introduce two filtering techniques. The first technique involves reconstructing the GLM signal using the fitting coefficients estimated by GLM and replacing the original BOLD signal in the linear regression equation, achieving the first level of filtering. Next, we apply a frequency domain filter for the second level of filtering. Although the linear regression equation remains in complex form after two rounds of filtering, we convert it into a real equation through elementary transformations, simplifying the calculation process further. Finally, we optimize the cost function to determine the optimal hyperparameters and use variational inference techniques to obtain the final DCM regression parameters.

4.1. DCM linear regression

The original DCM describes the derivative \dot{z}_t of the neuronal state in the brain region as the sum of the intrinsic coupling between the neuronal states, the coupling regulated by the exogenous stimulus, and

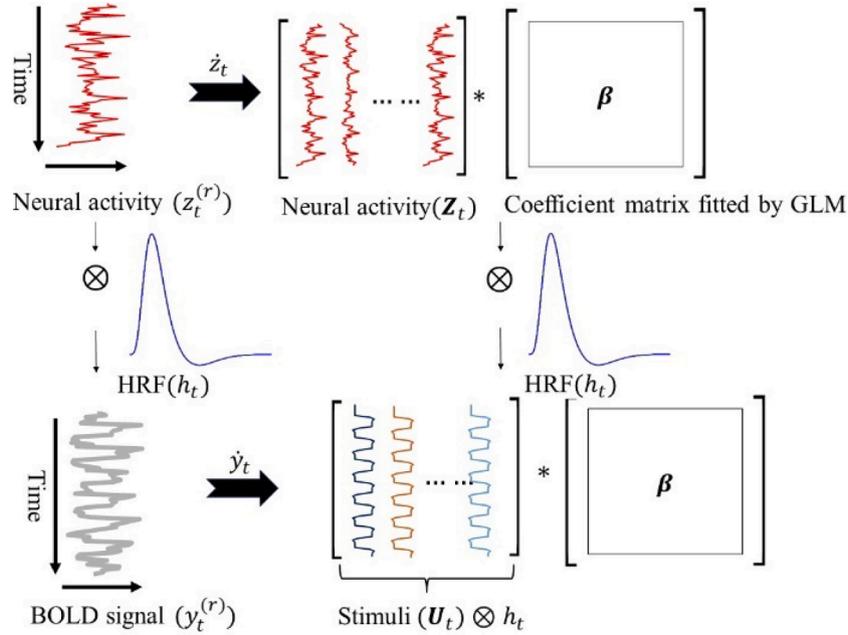


Fig. 3. Schematic diagram of the linear regression model construction without hidden state.

the exogenous stimulus (Friston et al., 1997). Since z_t cannot be obtained directly and is considered a hidden state, a state space model is typically needed to solve the coupling relationship. In this paper, we utilize the convolution model of the BOLD signal to convolve the hemodynamic response function (HRF) on both sides of the original DCM equation. With the fitting coefficients from the GLM, the original equation can be transformed such that the BOLD signal y_t represents the product of the exogenous stimulus convolved with HRF and the fitting coefficients, as shown in Fig. 3. In the new equation, y_t is the observed signal, and the exogenous stimulus, which is derived from the experimental design, is considered known. This eliminates the need for a state space model. Additionally, since the BOLD signal is a discrete signal, its derivative can be approximated differentially and then Fourier transformed, resulting in a linear regression equation in the frequency domain. Below, we will describe the linear regression equation in detail.

Let $z_t^{(r)}$ be the neuronal activity state of the r -th brain region at time t . According to dynamic theory and DCM (Friston et al., 2003), the differential \dot{z}_t of the neuronal state of the brain region can be expressed as (for simplicity, r is omitted)

$$\dot{z}_t = \left(\mathbf{A} + \sum_{j=1}^K u_t^j \mathbf{B}^j \right) \mathbf{Z}_t + \mathbf{C} \mathbf{U}_t \quad (1)$$

where

u_t^j represents the j -th stimulus at time t ,

$\mathbf{Z}_t = [z_t^{(r)}] \in \mathbb{R}^{R \times 1}$ represents the column vector composed of the neuronal activities of R brain regions at time t ,

$\mathbf{U}_t = [u_t^k] \in \mathbb{R}^{K \times 1}$ represents the column vector composed of K stimulus at time t ,

$\mathbf{A} \in \mathbb{R}^{1 \times R}$ represents the row vector of endogenous network connectivity coefficients that are independent of stimulation,

$\mathbf{B}^j \in \mathbb{R}^{1 \times R}$ represents the row vector of endogenous network connectivity coefficients modulated by the j -th stimulus,

$\mathbf{C} \in \mathbb{R}^{1 \times K}$ represents the row vector of neuronal activity directly affected by the stimulus.

According to the convolutional linear model, the BOLD signal can be expressed as

$$y_t = z_t \otimes h_t \quad (2)$$

where h_t is HRF, so convolving both sides of (1) with HRF yields

$$\dot{y}_t = \mathbf{A} \boldsymbol{\psi}_t + \sum_{j=1}^K \mathbf{B}^j \mathbf{y}_t^j + \mathbf{C} \mathbf{y}_t'' \quad (3)$$

where

$$\boldsymbol{\psi}_t = \mathbf{Z}_t \otimes h_t = [y_t^{(r)}] \in \mathbb{R}^{R \times 1},$$

$$\mathbf{y}_t^j = u_t^j \mathbf{b}^T \mathbf{U}_t \otimes h_t \text{ (see appendix for proof),}$$

\mathbf{b} is the fitting coefficient matrix obtained by GLM,

$$\mathbf{y}_t'' = \mathbf{U}_t \otimes h_t$$

If the BOLD signal y_t is expressed in discrete form and its value at the n -th sampling point is

$$y_n = y_{n\Delta t} \quad (4)$$

where Δt represents the sampling interval, i.e. repetition time (TR), then \dot{y}_n can be approximated as

$$\dot{y}_n \approx \frac{y_{n+1} - y_n}{\Delta t} \quad (5)$$

Substituting (5) into (3), and performing digital Fourier transform on $n = 1, 2, \dots, N$ sampling points, we have

$$\dot{Y}_n = \mathbf{A} \boldsymbol{\Psi}_n + \sum_{j=1}^K \mathbf{B}^j \mathbf{Y}_n^j + \mathbf{C} \mathbf{Y}_n'' \quad (6)$$

where \dot{Y}_n , $\boldsymbol{\Psi}_n$, \mathbf{Y}_n^j and \mathbf{Y}_n'' are the N -point Fourier transform results of \dot{y}_n , $\boldsymbol{\psi}_n$, \mathbf{y}_n^j and \mathbf{y}_n'' , respectively.

When performing the Fourier transform, the number of basis functions is set to match the time points of the BOLD signal. Let

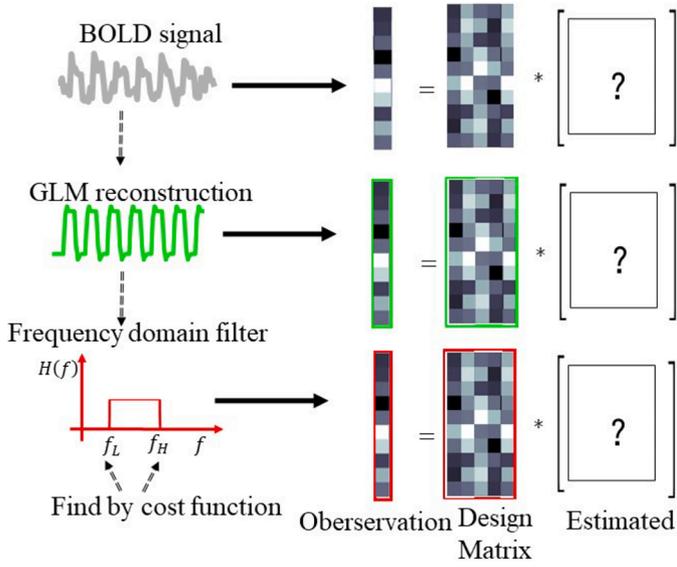


Fig. 4. GLM filtering and frequency domain filtering.

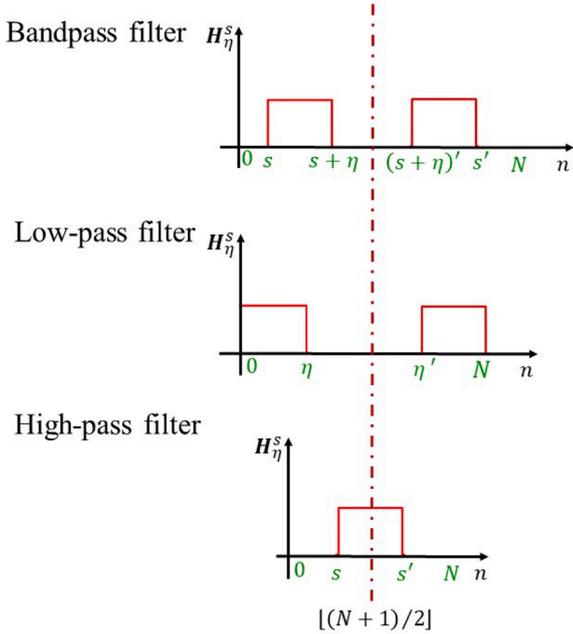


Fig. 5. A frequency domain filter H_{η}^s used in this algorithm.

$$\mathbf{X}^o = \begin{bmatrix} \Psi_1 & \Psi_2 & \dots & \Psi_N \\ \mathbf{Y}_1^1 & \mathbf{Y}_2^1 & \dots & \mathbf{Y}_N^1 \\ \mathbf{Y}_1^2 & \mathbf{Y}_2^2 & \dots & \mathbf{Y}_N^2 \\ \vdots & \vdots & \dots & \vdots \\ \mathbf{Y}_1^K & \mathbf{Y}_2^K & \dots & \mathbf{Y}_N^K \\ \mathbf{Y}_1^r & \mathbf{Y}_2^r & \dots & \mathbf{Y}_N^r \end{bmatrix}^T \in \mathbb{C}^{N \times (R+RK+K)} \quad (7)$$

$$\boldsymbol{\theta} = [\mathbf{A}\mathbf{B}^1 \mathbf{B}^2 \dots \mathbf{B}^K \mathbf{C}]^T \in \mathbb{R}^{(R+RK+K) \times 1} \quad (8)$$

$$\dot{\mathbf{Y}}^o = [\dot{Y}_1 \dot{Y}_2 \dots \dot{Y}_N]^T \in \mathbb{C}^{N \times 1} \quad (9)$$

Then, (6) becomes

$$\dot{\mathbf{Y}}^o = \mathbf{X}^o \boldsymbol{\theta} \quad (10)$$

In (10), the vector $\dot{\mathbf{Y}}^o$ is composed of the derivative of the BOLD signal and is known. The design matrix \mathbf{X}^o is composed of the BOLD signal, the stimulus and the GLM fitting coefficients, and it is also known. The parameter vector $\boldsymbol{\theta}$ represents the neural connectivity parameter to be estimated and is unknown. Therefore, (10) can be regarded as a linear regression equation.

Note that the estimated connectivity parameters of the original DCM model are matrices, which means that the estimation is the inversion of a multivariate timeseries. In (10), the parameter $\boldsymbol{\theta}$ is only a vector for the neural connectivity between one brain region and others, so the regression is carried out region by region.

4.2. GLM filtering

In the linear regression equation constructed in the previous section, both the observation on the left and the design matrix on the right involve the BOLD signal, which is extracted from the fMRI image and inevitably contains noise. To improve estimation accuracy, this section uses filtering techniques to reduce noise, employing both GLM filtering and frequency domain filtering, as illustrated in Fig. 4. For GLM filtering, we first use GLM to reconstruct the BOLD signal. We then replace the original BOLD signal in the observation and design matrices with the reconstructed GLM signal to obtain new observation and design matrices, as shown in the green box in Fig. 4. For frequency domain filtering, we filter the observation and design matrices obtained after GLM filtering to further refine them, as shown in the red box in Fig. 4. The high-frequency and low-frequency cutoff frequencies of the filter are optimized using the cost function (see Section 4.6). Next, we will describe the process of GLM filtering.

Using GLM (Worsley and Friston, 1995; Poline and Brett, 2012), the GLM output signal $\mathbf{y}^g = [\gamma_n] \in \mathbb{R}^{N \times 1}$ of a brain region can be expressed as

$$\mathbf{y}^g = \mathbf{x}^g \boldsymbol{\beta} \quad (11)$$

where $\mathbf{x}^g \in \mathbb{R}^{N \times K}$ is the GLM design matrix for K stimulus and $\boldsymbol{\beta} \in \mathbb{R}^{K \times 1}$ is the fitting coefficient matrix. If the observed BOLD signal y_n of the brain region is replaced by the GLM output γ_n , then we have

$$y_n = \gamma_n + \xi_n \quad (12)$$

where ξ_n is the error between the BOLD signal and the GLM output signal. Substituting (12) into (10), the Fourier transform signal \mathbf{Y}^g of the GLM output of a brain region can be expressed as (see the appendix for proof):

$$\mathbf{Y}^g = \mathbf{X}^g \boldsymbol{\theta} + \boldsymbol{\Xi}^g \quad (13)$$

where $\mathbf{Y}^g = [\hat{\Gamma}_n] \in \mathbb{C}^{N \times 1}$ and $\hat{\Gamma}_n$ is the Fourier transform of the GLM output derivative $\dot{\gamma}_n = \frac{\gamma_{n+1} - \gamma_n}{\Delta t}$. The corresponding design matrix \mathbf{X}^g is expressed as:

$$\mathbf{X}^g = \begin{bmatrix} \Psi_1^g & \Psi_2^g & \dots & \Psi_N^g \\ \mathbf{Y}_1^1 & \mathbf{Y}_2^1 & \dots & \mathbf{Y}_N^1 \\ \mathbf{Y}_1^2 & \mathbf{Y}_2^2 & \dots & \mathbf{Y}_N^2 \\ \vdots & \vdots & \dots & \vdots \\ \mathbf{Y}_1^K & \mathbf{Y}_2^K & \dots & \mathbf{Y}_N^K \\ \mathbf{Y}_1^r & \mathbf{Y}_2^r & \dots & \mathbf{Y}_N^r \end{bmatrix}^T \in \mathbb{C}^{N \times (R+RK+K)} \quad (14)$$

where

Ψ_n^g is the result of replacing the Fourier transform of the BOLD signal y_n of Ψ_n in (7) with the Fourier transform of the GLM output γ_n ,

$\boldsymbol{\Xi}^g$ is the error vector generated when ξ_n in (12) is substituted into (10).

In (13), the BOLD signal involved in the vector \mathbf{Y}^g has been replaced

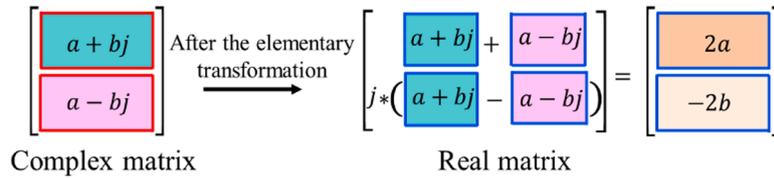


Fig. 6. Complex to real number in this regression.

Table 1

Prior distribution of parameters.

Model parameter	Prior distribution
Connectivity parameters	$p(\theta) = \mathcal{N}(\theta; \mu_0, \Sigma_0) = \frac{1}{\sqrt{(2\pi)^D}} \exp\left(-\frac{1}{2}(\theta - \mu_0)^T \Sigma_0^{-1} (\theta - \mu_0)\right)$
Noise accuracy	$p(\tau) = \text{Gamma}(\tau; \alpha_0, \beta_0) = \frac{\beta_0^{\alpha_0}}{\Gamma(\alpha_0)} \tau^{\alpha_0-1} \exp(-\beta_0 \tau)$
Binary random variable	$p(\zeta_i) = \text{Bern}(\zeta_i; p_0^i) = (p_0^i)^{\zeta_i} (1-p_0^i)^{1-\zeta_i}$

Table 2

Parameters in the prior distributions.

Parameter	Description	Parameter	Description
μ_0	Mean of a Gaussian distribution	β_0	Rate parameter of the Gamma distribution
Σ_0	Covariance of Gaussian distribution	τ_i	Noise accuracy of the i -th connectivity parameter
α_0	Shape parameter of the Gamma distribution	Γ	Gamma function
D	The number of incoming connections and driver inputs into the region	p_0^i	The probability of the Bern distribution for the i th connectivity parameter

Note: The recommended settings for the Gamma distribution parameters α_0 and β_0 are 2 and 1.

by the GLM signal, and the BOLD signal involved in the design matrix \mathbf{X}^s has also been replaced by the GLM signal.

4.3. Frequency domain filtering

This section describes the frequency domain filtering shown in Fig. 4 in detail. The filter will select appropriate frequencies to reduce the size of the Fourier transformed data, and select frequencies that reduce uncertainty or maximize information gain under the current posterior estimate of the generative model. Let $\mathbf{H}_\eta^s \in \{0, 1\}^{N \times 1}$ be an ideal digital filter vector in the Fourier transform domain, whose $(n-s)$ -th element is expressed as

$$\mathbf{H}_\eta^s(n-s) = g_\eta^s(n-s) + g_\eta^s(n-s) \quad (15)$$

where

$g_\eta^{s=0}(n)$ is a gate function with width η , that is:

$$g_\eta^{s=0}(n) = \begin{cases} 1, & 0 < n \leq \eta \\ 0, & \eta < n \leq N \end{cases} \quad (16)$$

s is a shift,

$g_\eta^s(n-s)$ is the mirror of the gate function $g_\eta^s(n-s)$ about axis $n = \lfloor (N+1)/2 \rfloor$.

The filter \mathbf{H}_η^s can be shown in Fig. 5, and its cutoff frequency can be set by s and η . When $s = 0$, it is low-pass filtering; when $s \geq (N+1)/2 - \eta$, it is high-pass filtering; the others are band-pass filtering. Filtering \mathbf{Y}^s

is to multiply both sides of (13) by the filter \mathbf{H}_η^s , and we have

$$\mathbf{Y}^f = \mathbf{X}^f \boldsymbol{\theta} + \boldsymbol{\varepsilon}^f \quad (17)$$

where \odot represents a Hadamard product and $(\bullet)^*$ represents removing zero elements, so $\mathbf{Y}^f = (\mathbf{Y}^s \odot \mathbf{H}_\eta^s)^*$ means the Hadamard product of two vectors and then removes the zero elements and $\mathbf{Y}^f \in \mathbb{C}^{M \times 1}$ in which M is the number of non-zero elements after the Hadamard product. Similarly, the design matrix is $\mathbf{X}^f = (\mathbf{X}^s \odot \mathbf{H}_\eta^s)^* \in \mathbb{C}^{M \times (R+RK+K)}$, and the noise vector after the filter is $\boldsymbol{\varepsilon}^f \in \mathbb{C}^{M \times 1}$.

In (17), after filtering, new observation and design matrices are obtained. Through frequency domain filtering, we aim to eliminate noise outside the desired frequency band while retaining useful signals within it. It is important to note that the parameters s and η in the filter design represent the low-frequency and high-frequency cutoff frequencies of the filter, respectively. The parameters are determined by optimizing the cost function, as introduced in Section 4.6.

4.4. Conversion to real numbers

Since this paper solves DCM in the frequency domain, the original real signal becomes a complex signal, making complex operations generally more complicated than real operations. Therefore, this section aims to convert the complex equation into a real one to reduce complexity. Due to the conjugate symmetry in the upper and lower parts of the digital Fourier transform result, and the fact that linear equations retain their solutions after elementary transformations, we achieve the real number conversion by adding or subtracting the upper and lower parts of the transformed result, as illustrated in Fig. 6.

Therefore, we perform elementary transformations on \mathbf{Y}^f and \mathbf{X}^f in (17) respectively, and have

$$\mathbf{Y}^f = \begin{bmatrix} \mathbf{Y}^f \left(1 : \frac{M}{2}\right) + \mathbf{Y}^f \left(\frac{M}{2} + 1 : M\right) \\ j \left(\mathbf{Y}^f \left(1 : \frac{M}{2}\right) - \mathbf{Y}^f \left(\frac{M}{2} + 1 : M\right) \right) \end{bmatrix} \in \mathbb{R}^{M \times 1} \quad (18)$$

$$\mathbf{X}^f = \begin{bmatrix} \mathbf{X}^f \left(1 : \frac{M}{2}\right) + \mathbf{X}^f \left(\frac{M}{2} + 1 : M\right) \\ j \left(\mathbf{X}^f \left(1 : \frac{M}{2}\right) - \mathbf{X}^f \left(\frac{M}{2} + 1 : M\right) \right) \end{bmatrix} \in \mathbb{R}^{M \times (R+RK+K)} \quad (19)$$

where $j = \sqrt{-1}$, $(m : n)$ represent the elements from the m -th row to the n -th row of the matrix. Replace the observation and design matrices in (17) with (18–19), and we have:

$$\mathbf{Y}^f = \mathbf{X}^f \boldsymbol{\theta} + \boldsymbol{\varepsilon}^f \quad (20)$$

where $\boldsymbol{\varepsilon}^f$ is the noise vector corresponding to the elementary transformation of (17). In (20), the dimensions of the matrices and vectors have not changed, but the elements are no longer complex numbers, but real numbers.

Table 3
Final iteration scheme.

Parameter	Iterative formula
Covariance in Gaussian distribution	$\Sigma_{\theta \dot{Y}} = \left(\frac{\alpha_{\tau \dot{Y}}}{\beta_{\tau \dot{Y}}} \left(P_{\zeta \dot{Y}} \mathbf{X}^T \mathbf{X} P_{\zeta \dot{Y}} + (\mathbf{X}^T \mathbf{X}) \circ (P_{\zeta \dot{Y}} - P_{\zeta \dot{Y}}^2) \right) + \Sigma_0^{-1} \right)^{-1}$
Statistics	$\mathbf{V} = \mathbf{X}^T \dot{\mathbf{Y}}$
Statistics	$\mathbf{W} = \mathbf{X}^T \mathbf{X}$
Mean in Gaussian distribution	$\mu_{\theta \dot{Y}} = \Sigma_{\theta \dot{Y}} \left(\frac{\alpha_{\tau \dot{Y}}}{\beta_{\tau \dot{Y}}} P_{\zeta \dot{Y}} \mathbf{X}^T \dot{\mathbf{Y}} + \Sigma_0^{-1} \mu_0 \right)$
Shape parameter of the Gamma distribution	$\alpha_{\tau \dot{Y}} = \alpha_0 + \frac{N}{2}$
Rate parameter of the Gamma distribution	$\beta_{\tau \dot{Y}} = \beta_0 + \frac{1}{2} \left((\dot{\mathbf{Y}} - \mathbf{X} P_{\zeta \dot{Y}} \mu_{\theta \dot{Y}})^T (\dot{\mathbf{Y}} - \mathbf{X} P_{\zeta \dot{Y}} \mu_{\theta \dot{Y}}) + \text{tr} \left(P_{\zeta \dot{Y}} \mathbf{X}^T \mathbf{X} P_{\zeta \dot{Y}} \Sigma_{\theta \dot{Y}} \right) \dots + \frac{1}{2} \left(\mu_{\theta \dot{Y}}^T (\mathbf{X}^T \mathbf{X}) \circ (P_{\zeta \dot{Y}} - P_{\zeta \dot{Y}}^2) \right) \mu_{\theta \dot{Y}} + \text{tr} \left((\mathbf{X}^T \mathbf{X}) \circ (P_{\zeta \dot{Y}} - P_{\zeta \dot{Y}}^2) \right) \Sigma_{\theta \dot{Y}} \right)$
Parameters of the Bernoulli prior	$P_{\zeta \dot{Y}}^i = \frac{1}{1 + \exp(-g_i)}$
Statistics	$g_i = \frac{\alpha_{\tau \dot{Y}} \mu_{\theta \dot{Y}}^i V_i - \frac{\alpha_{\tau \dot{Y}}}{2\beta_{\tau \dot{Y}}} \left((\mu_{\theta \dot{Y}}^i)^2 W_{ii} + 2\mu_{\theta \dot{Y}}^i \sum_{j \neq i} P_{\zeta \dot{Y}}^j \mu_{\theta \dot{Y}}^j W_{ij} \right) - \frac{\alpha_{\tau \dot{Y}}}{2\beta_{\tau \dot{Y}}} \left(W_{ii} \Sigma_{\theta \dot{Y}}^{ii} + 2 \sum_{j \neq i} P_{\zeta \dot{Y}}^j W_{ij} \Sigma_{\theta \dot{Y}}^{ij} \right) + \ln \left(\frac{P_0^i}{1 - P_0^i} \right)$

Note: Symbol \circ represents the corresponding multiplication of the elements in two matrices, tr is the trace of the matrix; $\mu_{\theta|\dot{Y}}$ represents the mean μ of the variational distribution $q(\theta)$ given \dot{Y} , and the other symbols $\Sigma_{\theta|\dot{Y}}$, $\alpha_{\tau|\dot{Y}}$ and $\beta_{\tau|\dot{Y}}$ are expressed in a similar way; $P_{\zeta|\dot{Y}}$ represents the posterior probability of the variational distribution $q(\zeta)$ given \dot{Y} .

Table 4
Expressions for the components of negative free energy.

Component	Expression
Expectation of the likelihood	$\langle \ln p(\theta, \tau, \zeta \dot{Y}) \rangle_{q(\theta, \tau, \zeta)} = -\frac{N}{2} \ln 2\pi + \frac{N}{2} (\mathcal{S}(\alpha_{\tau \dot{Y}}) - \ln \beta_{\tau \dot{Y}}) - \frac{\alpha_{\tau \dot{Y}}}{2\beta_{\tau \dot{Y}}} \left((\dot{\mathbf{Y}} - \mathbf{X} P_{\zeta \dot{Y}} \mu_{\theta \dot{Y}})^T (\dot{\mathbf{Y}} - \mathbf{X} P_{\zeta \dot{Y}} \mu_{\theta \dot{Y}}) - \text{tr} \left(P_{\zeta \dot{Y}} \mathbf{W} P_{\zeta \dot{Y}} \Sigma_{\theta \dot{Y}} \right) - \frac{\alpha_{\tau \dot{Y}}}{2\beta_{\tau \dot{Y}}} \left(\mu_{\theta \dot{Y}}^T \mathbf{W} \circ (P_{\zeta \dot{Y}} - (P_{\zeta \dot{Y}})^2) \mu_{\theta \dot{Y}} - \text{tr} \left(\mathbf{W} \circ (P_{\zeta \dot{Y}} - (P_{\zeta \dot{Y}})^2) \Sigma_{\theta \dot{Y}} \right) \right)$
Expectation of the prior on θ	$\langle \ln p(\theta) \rangle_{q(\theta)} = -\frac{D}{2} \ln 2\pi - \frac{1}{2} \ln \Sigma_0 - \frac{1}{2} (\mu_{\theta \dot{Y}} - \mu_0)^T \Sigma_0^{-1} (\mu_{\theta \dot{Y}} - \mu_0) - \frac{1}{2} \text{tr} (\Sigma_0^{-1} \Sigma_{\theta \dot{Y}})$
Expectation of the prior on τ	$\langle \ln p(\tau) \rangle_{q(\tau)} = \alpha_0 \ln \beta_0 - \ln \Gamma(\alpha_0) + (\alpha_0 - 1) (\mathcal{S}(\alpha_{\tau \dot{Y}}) - \ln \beta_{\tau \dot{Y}}) - \beta_0 \frac{\alpha_{\tau \dot{Y}}}{\beta_{\tau \dot{Y}}}$
Expectation of the prior on ζ_i	$\langle \ln p(\zeta_i) \rangle_{q(\zeta_i)} = \ln(1 - P_0^i) + P_{\zeta \dot{Y}}^i \ln \frac{P_0^i}{1 - P_0^i}$
Entropy of θ	$-\langle \ln q(\theta) \rangle_{q(\theta)} = \frac{D}{2} (1 + \ln 2\pi) + \frac{1}{2} \ln \Sigma_{\theta \dot{Y}} $
Entropy of τ	$-\langle \ln q(\tau) \rangle_{q(\tau)} = \alpha_{\tau \dot{Y}} - \ln \beta_{\tau \dot{Y}} + \ln \Gamma(\alpha_{\tau \dot{Y}}) - (\alpha_{\tau \dot{Y}} - 1) \mathcal{S}(\alpha_{\tau \dot{Y}})$
Entropy of ζ_i	$-\langle \ln q(\zeta_i) \rangle_{q(\zeta_i)} = -P_{\zeta \dot{Y}}^i \ln P_{\zeta \dot{Y}}^i - (1 - P_{\zeta \dot{Y}}^i) \ln(1 - P_{\zeta \dot{Y}}^i)$

Note: $\langle \cdot \rangle_{q(\cdot)}$ represents the expectation of variational distribution $q(\cdot)$, \mathcal{S} is the digamma function.

4.5. Sparse variational Bayesian inference

Once the linear regression equation is established, we use the sparse variational Bayes method to solve for the model parameters. If the variational distribution of the model parameters is denoted as $q(\cdot)$, we continuously optimize it to approximate the true posterior distribution.

We assume the following prior information to build the model: (i) To ensure that the required model parameters exhibit sparsity, we introduce sparse constraints based on the method in reference (Frässle et al., 2018), using the binary random variable ζ_i for the i -th connectivity parameter as the feature selector in the likelihood function. Its prior follows a Bernoulli distribution with probability P_0^i . (ii) The prior of the neural connectivity parameter θ follows a Gaussian distribution with mean μ_0 and covariance matrix Σ_0 . (iii) The prior of the precision parameter τ for the noise Ξ follows a Gamma distribution with parameters α_0 and β_0 . Table 2 presents the prior distributions of the model parameters involved in this algorithm, and Table 1 explains the parameters listed in Table 2. Additionally, it is important to explain the Bernoulli distribution probability P_0^i , which is the prior probability of the existence of a connectivity parameter. When performing model regression using the GSD, the sparsity of the brain functional network is encoded by setting different values for P_0^i . This allows some connections to be eliminated, so that the scheme only needs to estimate the connectivity from the Markov blanket of each node. During the model inversion, the computational load can be gradually reduced as some effective connections are removed.

Since the model regression is performed under the mean field theory (Højten-Sørensen et al., 2002), the model parameters entering different regions are independent of each other and the model parameters of each brain region can be solved independently. Therefore, the posterior distribution of the model parameters in a single brain region is

$$p(\theta, \tau, \zeta | \dot{Y}, \mathbf{X}) \propto p(\dot{Y} | \mathbf{X}, \theta, \tau, \mathbf{S}) p(\theta) p(\tau) \prod_{i=1}^D p(\zeta_i) \quad (21)$$

where

$$p(\dot{Y} | \mathbf{X}, \theta, \tau, \mathbf{S}) = \mathcal{N}(\dot{Y}; \mathbf{X} \mathbf{S} \theta, \tau^{-1} \mathbf{I}_{N \times N})$$

$\mathbf{S} = \text{diag}(\{\zeta_1, \zeta_2, \dots, \zeta_R, \zeta_{R+1}, \dots, \zeta_{R+K}\})$ is a set of binary random variables, which encode the network structure as a sparse term,

$$\zeta_i \in \{0, 1\}.$$

Since (21) does not necessarily have a closed solution, we use the variational distribution $q(\cdot)$ to make an approximate estimate of the posterior distribution of θ , τ and ζ_i , and assume that the above parameters are independent of each other, then the approximate posterior distribution of (21) is

$$q(\theta, \tau, \zeta | \dot{Y}, \mathbf{X}) \approx q(\theta | \dot{Y}, \mathbf{X}) q(\tau | \dot{Y}, \mathbf{X}) \prod_{i=1}^D q(\zeta_i | \dot{Y}, \mathbf{X}) \quad (22)$$

Here, the variational distribution $q(\cdot)$ can be specified as the conjugate distribution of the prior distribution of the corresponding parameter.

Within the framework of factorization (22), the variational update equations for $q(\theta | \dot{Y}, \mathbf{X})$, $q(\tau | \dot{Y}, \mathbf{X})$, and $q(\zeta_i | \dot{Y}, \mathbf{X})$ can be derived using the variational Bayes. The optimal posterior distribution is obtained by iterating the update equations until convergence. The optimization condition is to maximize the negative (variational) free energy, which for a single region is expressed as the sum of the expected energy of the system (the logarithm of the joint distribution) and the entropy of q under the variational density. Due to space constraints, we provide only the final iteration scheme of the variational update equations and the expression of the negative free energy for a single region in Tables 3 and 4. For a detailed derivation, please refer to reference (Bishop and Nasrabadi, 2006).

4.6. Hyperparameter optimization

In variational inference, selecting hyperparameters such as the cutoff frequencies s and η , the prior mean of the connectivity parameter μ_0 , and the Bernoulli probability P_0^i is crucial for performance. This section determines the hyperparameters by optimizing a cost function. This

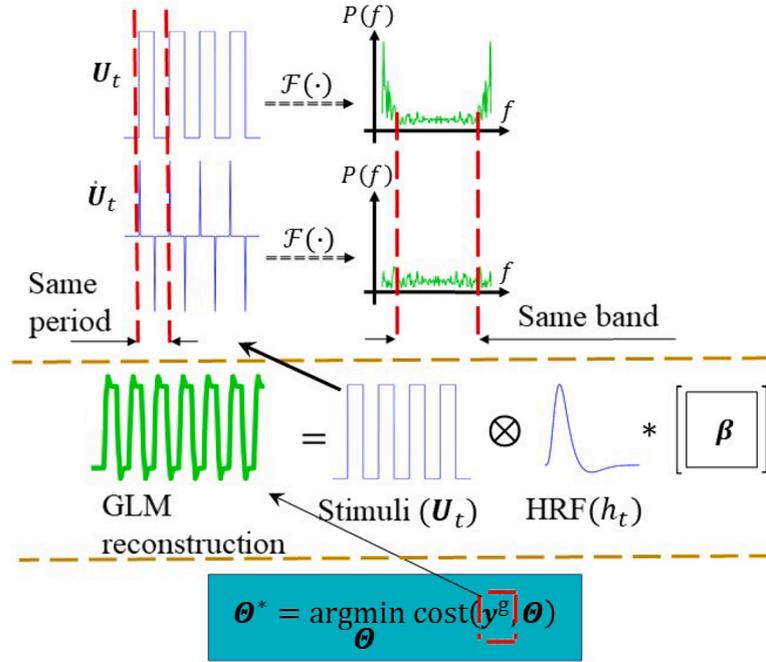


Fig. 7. Cost function for hyperparameter optimization.

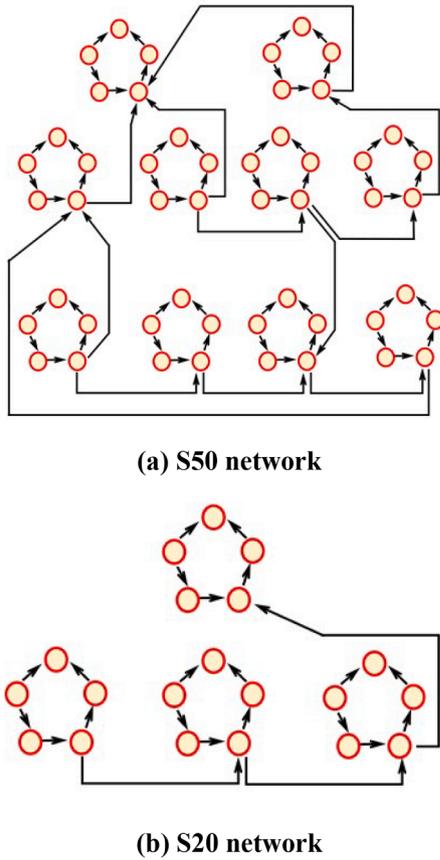


Fig. 8. Simulation data: S50 and S20 network based on the small-world architecture of the human brain (Lin Tiger et al., 2020; FRÄSSLE et al., 2018).

objective function replaces observed data with model-generated data in a way that maximizes both the evidence or marginal likelihood of the generative model and the expected information gain provided by the

data selection.

Let the hyperparameter vector be $\Theta = \langle s, \eta, \mu_0, p_0 \rangle$, where μ_0 and p_0 represent the mean and Bernoulli probability of the i -th connectivity parameter (the same value is used for all i , so i is omitted). The optimal hyperparameters are obtained by

$$\Theta^* = \underset{\Theta}{\operatorname{argmin}} \|\mathbf{y}^g - \mathcal{F}^{-1}(\mathbf{X}^g \hat{\theta}^\Theta)\|_2 \quad (23)$$

where $\hat{\theta}^\Theta$ represents the estimate of θ obtained using the hyperparameter Θ , and \mathcal{F}^{-1} is the inverse Fourier transform function.

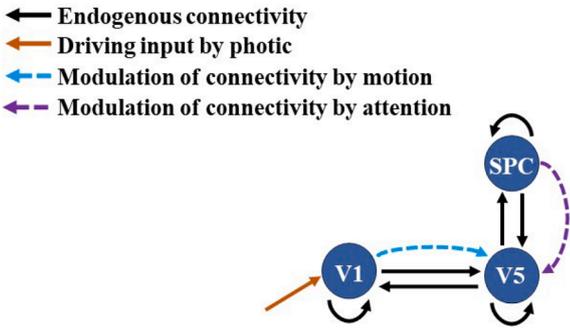
Compared with the traditional cost function method based on maximum a posteriori probability and negative free energy, the cost function of this paper does not involve the observed value $\mathbf{y} = [y_n]$, but uses the output $\mathbf{y}^g = [\gamma_n]$ of GLM, which can avoid introducing the error in y_n into the cost function. In addition, it is noted that $\mathbf{y}^g = [\gamma_n]$ is used in the cost function instead of its derivative $\dot{\mathbf{y}}^g = [\dot{\gamma}_n]$. We will explain it as follows, as shown in Fig. 7. Based on the convolutional model, (11) can be transformed into

$$\dot{\gamma}_n = (\dot{\mathbf{U}}_t^T \otimes h_t) \beta \quad (24)$$

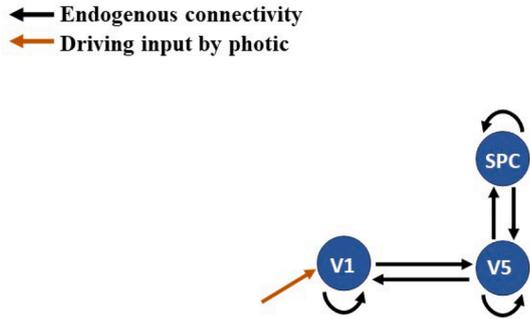
Since \mathbf{U}_t is an exogenous stimulus and can be regarded as a step signal, it has the same period as its differential $\dot{\mathbf{U}}_t$, so the two are similar in some frequencies in the spectrum, as shown in Fig. 7. From (23), it can be seen that \mathbf{y}^g and $\dot{\mathbf{y}}^g$ are also similar in the spectrum, so the cost function in (23) directly replaces its differential with the GLM output.

5. Experimental setup

To evaluate the proposed algorithm, we used three datasets: one simulated dataset and two measured datasets. In the simulated dataset, we primarily focused on performance metrics such as specificity, sensitivity, and precision. The above performance metrics were chosen to reflect various aspects of parameter recovery; namely, different measures of goodness of fit and accuracy. However, because we are using variational procedures, we also have the important opportunity to assess the bounds on model evidence (i.e., variational free energy). This is essential for Bayesian model comparison; i.e., inference about models



(a) A model for GSD and DCM



(b) A model for sparse rDCM

Fig. 9. Measured data 1: Two connectivity modes in the attention and motion dataset.

as opposed to the parameters of any given model. Therefore, we also present results in the measured datasets that reflect the model selection capability of GSD. Detailed descriptions of these datasets are provided below.

5.1. Simulation data

The simulation data represents a brain function network structure based on the small-world architecture of the human brain. Following the research from reference (Smith et al., 2011), we constructed two networks: S50 and S20, as shown in Fig. 8. The S50 network comprises 10

subnetworks, each containing 5 nodes with unidirectional connections. The nodes receive experimental stimuli, and the subnetworks are interconnected through one node. The S20 network is a simplified version of the S50, consisting of only 4 subnetworks, thus involving 20 nodes in total. The construction of the networks follows the method outlined in reference (Frässle et al., 2018), achieving an actual sparsity of approximately 0.05. The simulation data is not generated by GSD, but rather through a method that generates synthetic fMRI data under a given signal-to-noise ratio (SNR) with a fixed hemodynamic convolution kernel.

The parameters in the tested algorithms are set as follows:

- sparse rDCM (Frässle et al., 2018): The binary indicator variables for the experimental stimulus are determined by the designed stimulus input pattern. For non-zero positions in the C matrix, the Bernoulli prior probability p_0^i is set to 1, while the rest are set to 0. For positions in the A matrix, p_0^i is set to 0.15, except for self-connections, which are set to 1. The initial values for μ_0 and Σ_0 use the standard prior values of DCM10 in SPM8 (version R4290). The cutoff frequency for the frequency domain filter is calculated using the filter in the algorithm.
- GSD: This is the algorithm proposed in this paper. Its parameters p_0^i , μ_0 , Σ_0 and filter parameters match those of sparse rDCM. Other parameter settings are described in Section 4.
- DCM: This is the traditional DCM method proposed in the literature (Friston et al., 2003). All parameters are set using the DCM12.5 settings of the SPM12 software (Revision 7771). The software download address is <https://www.fil.ion.ucl.ac.uk/spm/>.

In this simulation data, TR is set to 0.5, and the SNR is set to 1, 3, 5, 10, and 100. It should be noted that due to the high computational complexity of DCM in model regression, if DCM is used to invert the S50 network in the experimental platform (operating system: Windows 11, processor: AMD Ryzen 5 5600H with Radeon Graphics, RAM: 16.0 GB, simulation software: MATLAB R2020a), memory overflow will occur. Therefore, in the S50 network, only the regression results of GSD and sparse rDCM are given. In the S20 network, due to the small number of nodes and relatively small amount of calculation, the analysis results of DCM are given.

In the simulation data, the TR is set to 0.5 s, and the SNR varies between 1, 3, 5, 10, and 100. Note that due to the high computational complexity of model regression with DCM, attempting to regress the S50 network on our experimental platform (Windows 11, AMD Ryzen 5 5600H processor with Radeon Graphics, 16.0 GB RAM, MATLAB

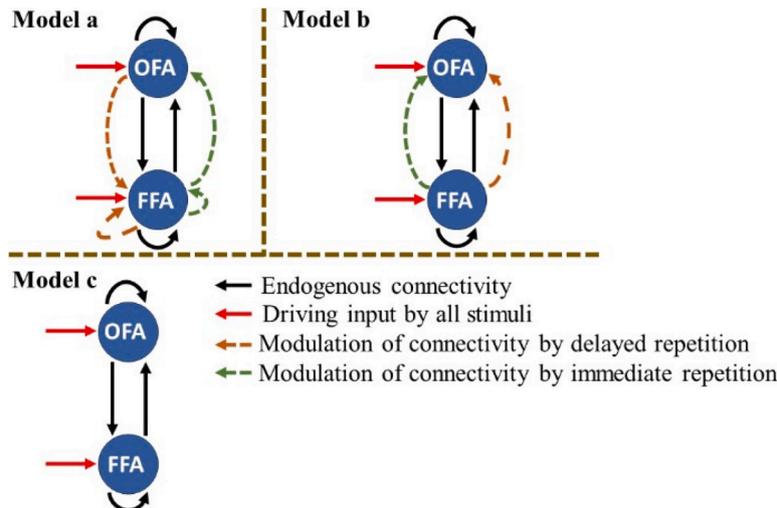


Fig. 10. Measured data 2: three connectivity models in the face fMRI dataset.

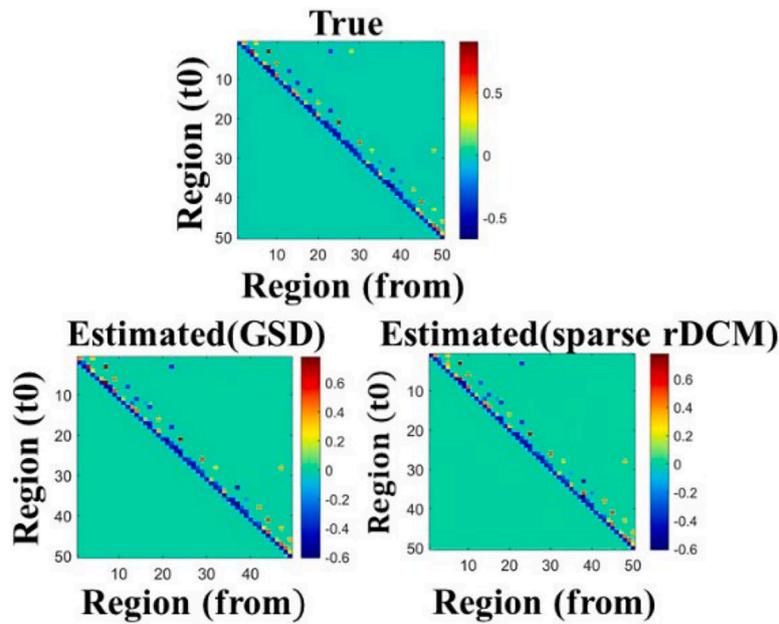


Fig. 11. Heat maps for the estimated and expected connections of GSD and sparse rDCM when the threshold is 10^{-2} (A matrix of S50 network).

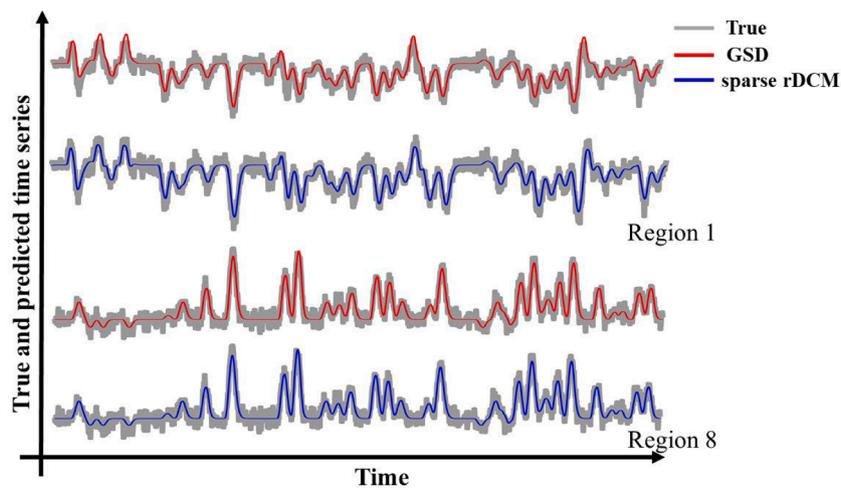


Fig. 12. When the threshold is 10^{-2} , the fitting curve of GSD and sparse rDCM (the first brain region and the eighth brain region).

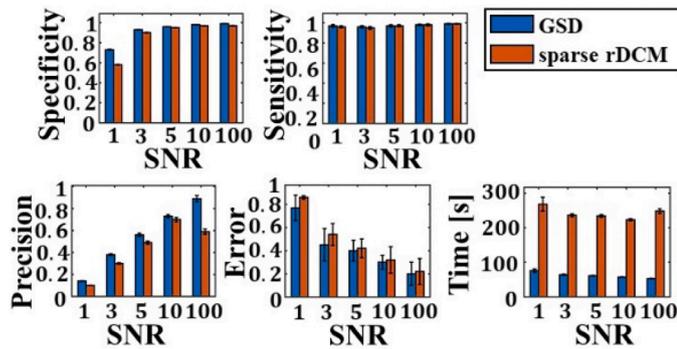


Fig. 13. Performance of GSD and sparse rDCM for S50 network regression when the threshold is 10^{-5} .

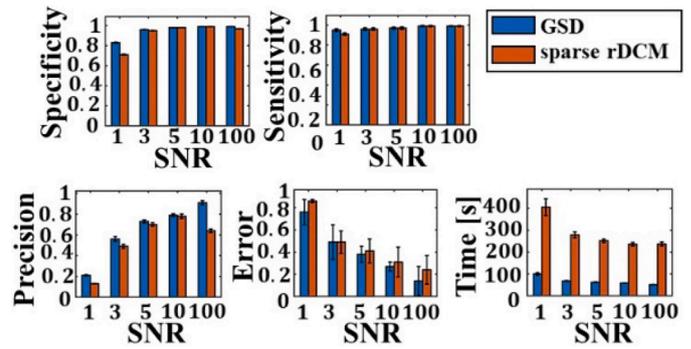


Fig. 14. Performance of GSD and sparse rDCM for S50 network regression when the threshold is 10^{-3} .

R2020a) results in memory overflow. Consequently, we provide regression results for only GSD and sparse rDCM in the S50 network. For the S20 network, due to its smaller size and reduced computational

demands, we include analysis results from DCM as well.

The performance metrics include the model's regression speed, sensitivity, specificity, precision, and estimation error. Sensitivity is

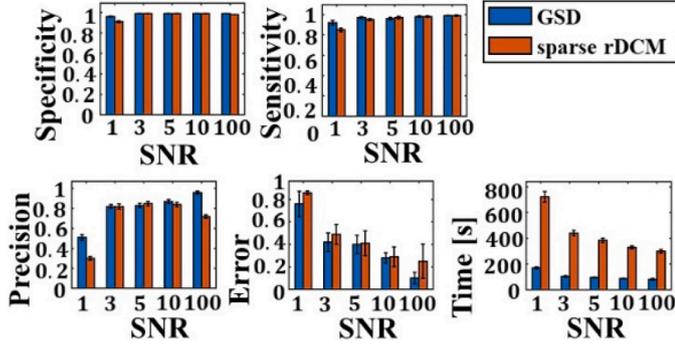


Fig. 15. Performance of GSD and sparse rDCM for S50 network regression when the threshold is 10^{-2} .

defined as:

$$Sensitivity = \frac{N_{tp}}{N_p} \quad (25)$$

where N_{tp} is the number of true positives obtained by regression (the expected value of the parameter is non-zero, and the regressed value of the parameter is also non-zero). If the absolute value of the regressed value is greater than the threshold, it is considered non-zero. This experiment adopts multiple thresholds, including 10^{-5} , 10^{-3} , and 10^{-2} , and N_p is the total number of positives (the expected value of the parameter is non-zero). Specificity is defined as:

$$Specificity = \frac{N_{tn}}{N_n} \quad (26)$$

where N_{tn} is the number of true negatives obtained by regression (the expected value of the parameter is zero, and the inverted value of the parameter is also zero), and N_n is the total number of negatives (the expected value of the parameter is zero). The precision is defined as:

$$Precision = \frac{N_{tp}}{N_{tp} + N_{fp}} \quad (27)$$

where N_{fp} is the number of false positives obtained from the regression (the expected value of the parameter is zero, and the inverted value of the parameter is non-zero). The estimation error is defined as:

$$Error = \frac{\|\hat{\theta} - \theta\|_2}{\|\theta\|_2} \quad (28)$$

where $\hat{\theta}$ is the parameter estimate and $\|\cdot\|_2$ represents the Euclidean

norm.

It is important to note that both GSD and sparse rDCM introduce sparse terms during model regression. The use of a threshold to determine whether the regression value is non-zero is a step in the regression process. Unless otherwise specified, the regression results of GSD and sparse rDCM use a default threshold of 10^{-5} .

5.2. Measured data 1: attention and motion dataset

The attention and motion dataset is commonly used to study the modulatory effects of attention on visual pathway connectivity. DCM can be employed to analyze the modulatory effects. The dataset is available at <http://www.fil.ion.ucl.ac.uk/spm/data/attention/>, with detailed information found in references (Büchel and Friston, 1991; Friston and Frith, 1995). The data has been preprocessed and includes smoothed, spatially normalized, realigned, and slice-time corrected functional images, as well as spatially normalized structural images.

The data acquisition experiment included 4 conditions: (I) fixation (F); (II) stationary point (S); (III) no attention, i.e., moving point but not requiring attention (N); (IV) attention, i.e., moving point requiring attention (A). Data was acquired using a 2 Tesla MRI scanner (Siemens, Erlangen), capturing 360 T2*-weighted functional images using a gradient echo-planar imaging (EPI) sequence (TR = 3220 ms, TE = 40 ms, voxel size $3 \times 3 \times 3$ mm³).

To identify activated regions and extract the BOLD signal, we

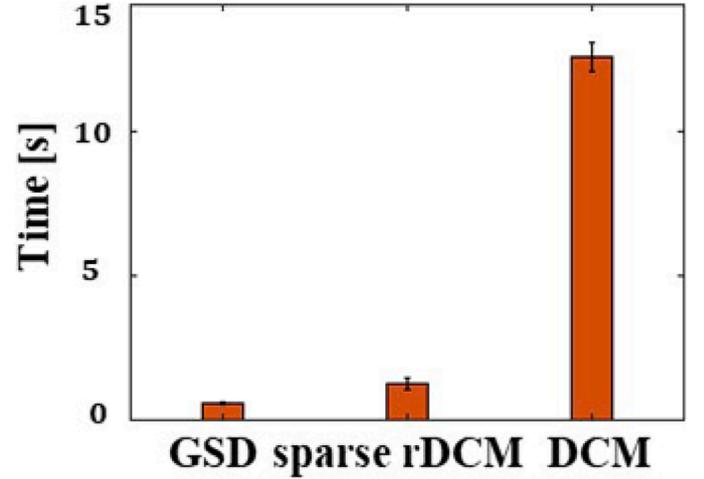


Fig. 17. Regression time for GSD, sparse rDCM and DCM.

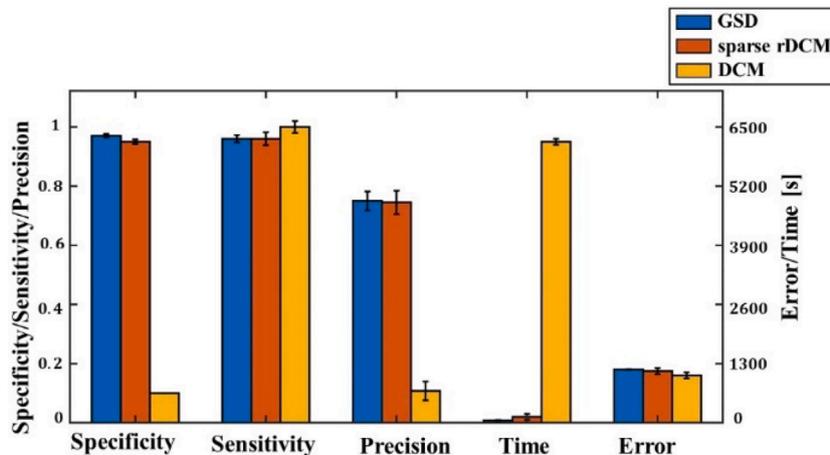
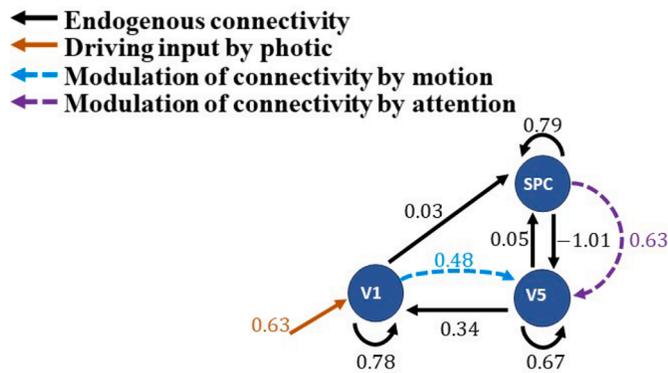
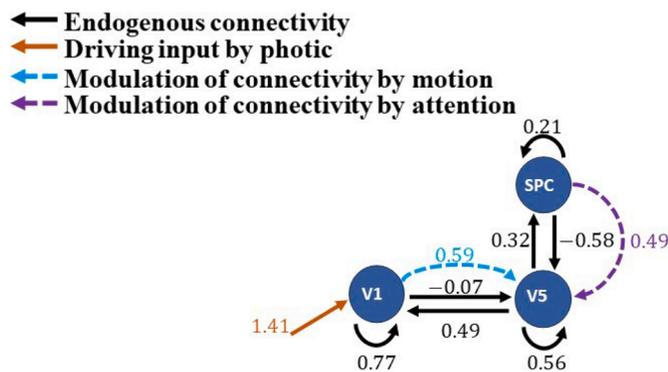


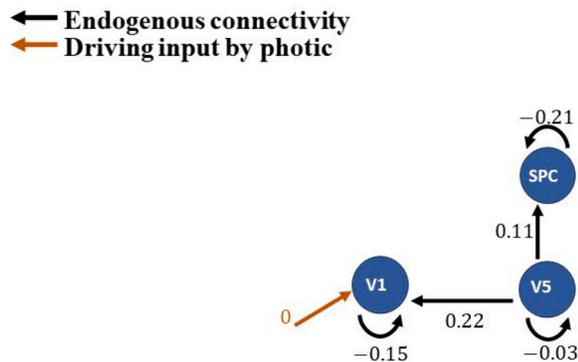
Fig. 16. Performance of GSD, sparse rDCM and DCM for S20 network regression when the threshold is 10^{-2} and SNR is 3.



(a) Regression of GSD



(b) Regression of DCM



(c) Regression of sparse rDCM

Fig. 18. Regression results for GSD, DCM and sparse rDCM.

performed a GLM analysis, recombining the above four experimental conditions to obtain three regressors: (1) photic perception (S, N, and A), (2) motion (N and A), and (3) attention (A). Based on the results of the GLM analysis, we extracted BOLD signals from three ROIs: the primary visual cortex (V1), the motion-sensitive area (V5), and the attention-sensitive superior parietal cortex (SPC), from a small sphere centered at $[0, -93, 18]$, $[-36, -87, -3]$, and $[0, -93, 18]$ with a radius of 8 mm.

According to reference (Büchel and Friston, 1991), the connectivity model of DCM is shown in Fig. 9(a), where the endogenous connectivity (\mathbf{A} matrix) includes the self-connection of three brain regions, the forward connectivity (SPC to V5) and the backward connectivity (V5 to SPC) between SPC and V5, and the forward connectivity (V1 to V5) and

backward connectivity (V5 to V1) between V1 and V5; the input modulation (\mathbf{B} matrix) includes motion modulates the endogenous connectivity from V1 to V5, attention modulates the endogenous connectivity from SPC to V5; in terms of driving input (\mathbf{C} matrix), photic stimulation drives the activity in V1. In this experiment, GSD and traditional DCM adopted the model of Fig. 9(a), while sparse rDCM adopted the model of Fig. 9(b) because it was impossible to estimate the \mathbf{B} matrix. The only difference between the two models is whether there is an input modulation.

The parameters in the tested algorithms are set as follows:

- sparse rDCM: The initial values μ_0 and Σ_0 are set using the standard prior values used by DCM10 in SPM8; the filter cutoff frequency is calculated by the algorithm itself; the Bernoulli prior probability p_0^i is obtained by maximizing the negative free energy, specifically, at the non-zero position of the \mathbf{C} matrix, p_0^i is set to 0.5, and the rest are 0; the value of p_0^i corresponding to each position in the \mathbf{A} matrix is set to 0.5, except for self-connection which is set to 1.
- GSD: The setting of p_0^i is optimized via (23), specifically, the p_0^i at the non-zero positions of the \mathbf{C} matrix and the \mathbf{B} matrix is set to 0.8, and the rest are 0; the p_0^i corresponding to each position in the \mathbf{A} matrix, except for the self-connection which is set to 1, the rest are all set to 0.8; the setting of Σ_0 uses the standard prior value used by DCM10 in SPM8; μ_0 and the cutoff frequency in the frequency domain filtering have two settings: (1) optimized via Eq.(23), which are 0.8 and 60 (high-pass cutoff frequency) respectively; (2) μ_0 uses the standard neural prior used by DCM10 in SPM8, and the cutoff frequency is calculated by the sparse rDCM algorithm.
- DCM: All parameters adopt the settings of DCM12.5 of SPM12.

In the results of the measured data, in addition to showing the regression results of each connectivity parameter in the models, we also give the results of the fitting error, that is, the difference between the fitted BOLD signal and the true BOLD signal. The fitted BOLD signals of GSD and sparse rDCM are expressed as the inverse fourier digital transform of the neural connectivity parameters and the design matrix, and the fitted BOLD signal of DCM is obtained by SPM12.

5.3. Measured data 2: face fMRI dataset

The face fMRI dataset is commonly used to study the face repetition suppression (RS) effect (Lee et al., 2022; Ewbank et al., 2013). The data can be accessed at <https://www.openfmri.org/dataset/ds000117/>, with detailed information in reference (Wakeman and Henson, 2015). This experiment selected fMRI data from 16 subjects, aged 23–37 years. Data were collected using a 3 Tesla MRI scanner (Siemens, Erlangen). The MP-RAGE sequence was used to obtain T1-weighted structural images with a resolution of $1 \times 1 \times 1$ mm, and the EPI sequence (TR = 2000 ms, TE = 30 ms, flip angle = 78° , voxel size ranging from $3 \times 3 \times 3.75$ mm to $3 \times 3 \times 4.05$ mm) was used to obtain functional images. Each subject participated in 9 experiments, and 210 functional images were collected in each experimental run. The experiment included two conditions: (I) repetition (initial stimulus, immediate repetition or delayed repetition), and (II) different faces (familiar faces, unfamiliar faces or scrambled faces).

After smoothing, spatially normalizing, realigning, slice time correction, and spatial normalization, we performed a GLM analysis, in which 9 regressors were defined: (1) initial familiar face, (2) immediately repeated familiar face, (3) delayed repeated familiar face, (4) initial unfamiliar face, (5) immediately repeated unfamiliar face, (6) delayed repeated unfamiliar face, (7) initial scrambled face, (8) immediately repeated scrambled face, and (9) delayed repeated scrambled face. The results were basically consistent with the literature (Lee et al., 2022). This experiment mainly focused on the right occipital face area (OFA) and right fusiform face area (FFA) because these two areas

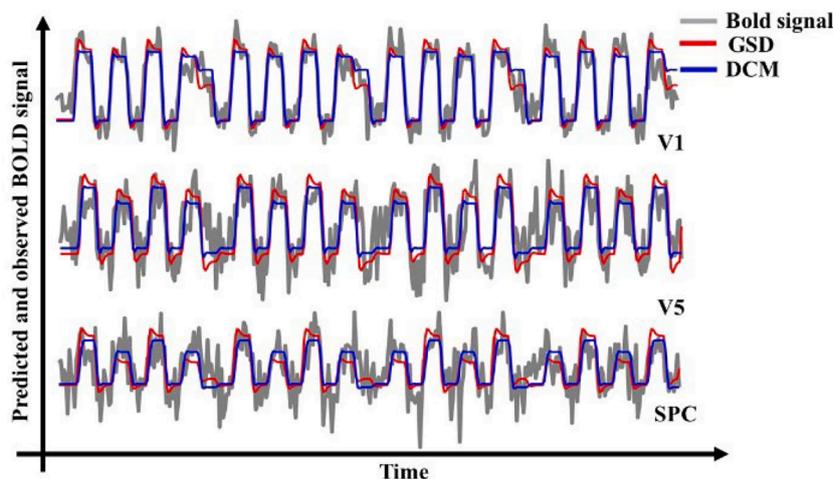


Fig. 19. Fitting curve of GSD and DCM.

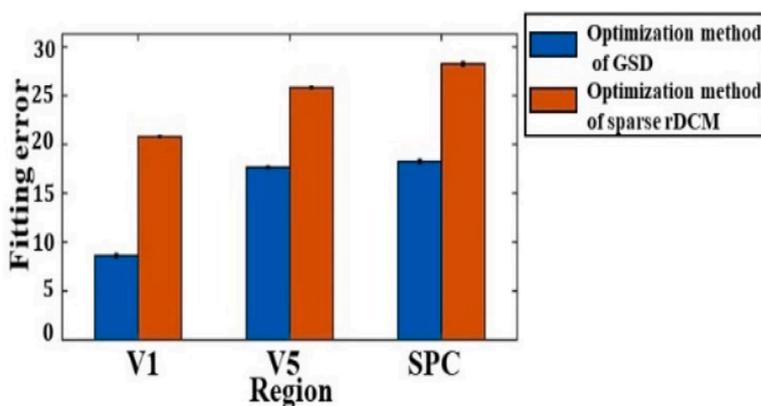


Fig. 20. GSD fitting error under two hyperparameter settings.

showed higher activation levels than other areas. We defined two small spheres with centers of $[39, -82, -10]$ and $[42, -46, -19]$ and a radius of 10 mm, respectively, and extracted the corresponding BOLD signals from them. Finally, we followed the method of (Lee et al., 2022) and recoded the nine regressors defined in the above GLM into three experimental effects: (1) all stimuli (contrast vector = $[(\text{Friston}, 2011)$ (Friston, 2011) (Friston, 2011)], the order is the same as the order of the regressors in the GLM), (2) immediate repetition (contrast vector = $[0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0]$) and (3) delayed repetition (contrast vector = $[0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1]$).

For the analysis of DCM, we assumed three connectivity patterns, as shown in Fig. 10. In these patterns, the endogenous connectivity is a fully connected network, and the driving input allows all stimuli to directly drive the activity of the brain region, so there are complete \mathbf{A} matrices and \mathbf{C} matrices. The input modulation (\mathbf{B} matrix) considers: (a) both immediate repetition and delayed repetition modulate the self-connection of FFA, and delayed repetition and immediate repetition also modulate the forward connectivity (OFA to FFA) and backward connectivity (FFA to OFA) between OFA and FFA respectively; (b) both immediate repetition and delayed repetition modulate the backward connectivity between OFA and FFA, (c) immediate repetition and delayed repetition do not have any modulation effect.

The parameters in the tested algorithms are set as follows:

- GSD: The setting of p_0^i is optimized via (23). Specifically, at the non-zero positions of the \mathbf{C} matrix and the \mathbf{B} matrix, p_0^i is set to 0.85, and

the rest are 0. The values of p_0^i corresponding to each position in the \mathbf{A} matrix are all set to 0.85 except for the self-connection which is set to 1. The setting of Σ_0 uses the standard prior value used by DCM10 in SPM8. The cutoff frequency in the frequency domain filter and μ_0 are optimized via (23), which are 0.1 and 400 (high-pass cutoff frequency) respectively.

- DCM: All parameters are set using SPM12's DCM12.5.

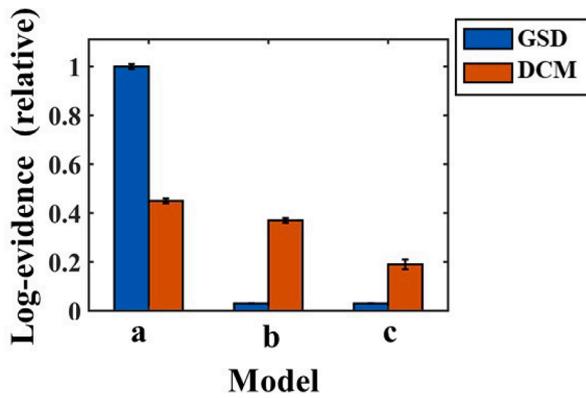
Note that the sparse rDCM was not tested on this dataset because it cannot regress the \mathbf{B} matrix, which is essential for this analysis.

Finally, we used GSD and DCM to regress the three connectivity models shown in Fig. 10. After selecting the best model, we summarized and analyzed the regression results for the 16 subjects.

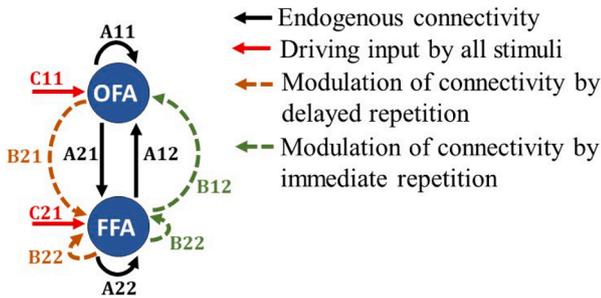
6. Result analysis

6.1. Simulation data

The results presented in this section are averages of 20 independent calculations, with the specific numerical values provided in the supplementary materials. Figs. 11 and 12 illustrate the estimated heat maps and fitted curves of GSD and sparse rDCM for the S50 network with a threshold of 10^{-2} . The fitted BOLD signals of GSD and sparse rDCM are represented as the inverse Fourier digital transform of the product of the neural connectivity parameters and the design matrix. Due to the extensive number of brain regions involved in the simulation data, only the fitted BOLD signals of two brain regions are displayed in Fig. 12.



(a) Relative model evidence



(b) The model selected by GSD and DCM algorithm

Fig. 21. Relative model evidence and model selection of GSD and DCM.

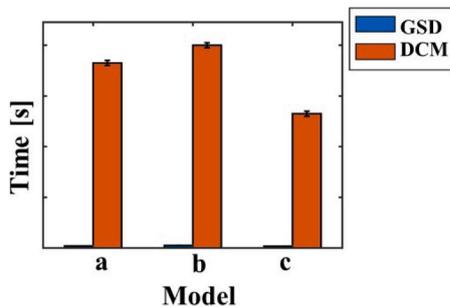


Fig. 22. Regression time of GSD and DCM.

From the estimated heat map and fitting curve, both algorithms exhibit similar performance.

To further analyze the differences in regression performance, Figs. 13–15 show the performance results of GSD and sparse rDCM for regressing the S50 network at thresholds of 10^{-5} , 10^{-3} , and 10^{-2} . In terms of specificity and sensitivity, the performance of both algorithms is relatively close, except when SNR=1. However, in terms of precision, the GSD algorithm consistently outperforms sparse rDCM at all SNR values. Notably, when SNR=100, the precision of sparse rDCM decreases, while the precision of GSD continues to increase. Additionally, as the threshold increases from 10^{-5} to 10^{-2} , the precision of both algorithms improves, with GSD maintaining higher precision than sparse rDCM. It is also worth noting that the computation time of GSD is significantly lower than that of sparse rDCM, approximately 1/3 of the latter.

Fig. 16 displays the regression results of the DCM, GSD, and sparse rDCM algorithms for the S20 network under SNR=3. As described in the experimental setup section, DCM regression for 50 nodes exceeds the

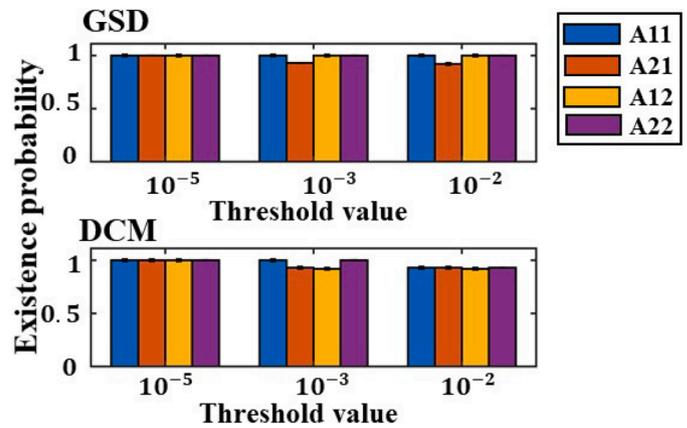


Fig. 23. The probability of the existence of endogenous connections (A matrix) at different thresholds (a) The probability of delayed repetition modulation (B_1 matrix) at different thresholds (b) The probability of immediate repetition modulation (B_2 matrix) at different thresholds.

memory capacity of the computing host used, so results are only provided for the S20 network. Since results under other SNR values are similar, only those for SNR=3 are presented. From Fig. 16, it is evident that the precision and specificity of DCM are lower than those of GSD and sparse rDCM, and its computation time is significantly higher. Additionally, although the specificity and other performance metrics of the GSD algorithm are close to those of sparse rDCM, GSD's computation time remains lower, consistent with the S50 network results in Fig. 13.

6.2. Measured data 1: attention and motion dataset

The results presented here are averages of 20 independent calculations. Fig. 17 shows the regression time for the three models. GSD's regression time is <1 second, sparse rDCM's is about 1–2 s, and DCM's is >12 s, with GSD being the fastest.

Fig. 18 shows the regression results of each connectivity parameter in GSD, DCM, and sparse rDCM. For endogenous connections, all three brain regions in the models exhibit self-connection. However, GSD and DCM show excitation effects, while sparse rDCM shows inhibitory effects. Both GSD and DCM reveal forward (SPC to V5) and backward (V5 to SPC) connections between SPC and V5, whereas sparse rDCM only shows backward connections. Neither DCM nor sparse rDCM show connections between V1 and SPC, and the connectivity strength from V1 to SPC in GSD is only on the order of 10^{-2} , which can be considered negligible. All three models identify the backward connectivity between V1 and V5 (V5 to V1), while only DCM finds the forward connectivity (V1 to V5), albeit at the order of 10^{-2} . Regarding input modulation, GSD and DCM exhibit the same modulation effect, while sparse rDCM shows none. For driving input, both GSD and DCM have photic stimulation input in the V1 area, whereas sparse rDCM does not. Overall, GSD and DCM yield similar regression results, while sparse rDCM differs significantly, with smaller amplitudes.

Fig. 19 shows the fitting curves of GSD and DCM, which are close to the true curves. Sparse rDCM's fitting curve, being quite different from the true curve, is not shown. Finally, Fig. 20 displays the fitting errors of GSD under two hyperparameter optimization settings. The first uses optimization method (23), and the second uses the sparse rDCM method. The fitting errors in the three ROIs are smaller with the first method.

6.3. Measured data 2: face fMRI dataset

The results here are averages of 20 independent calculations for 16 subjects. Fig. 21(a) shows the relative model evidence for the three connectivity models, calculated by negative free energy, as per SPM12. The relative model evidence for each model is the average of 16 subjects,

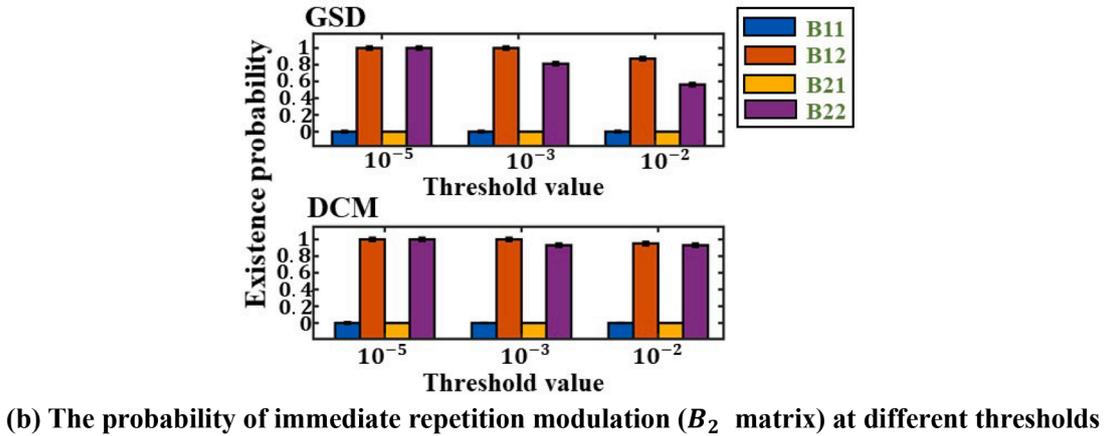
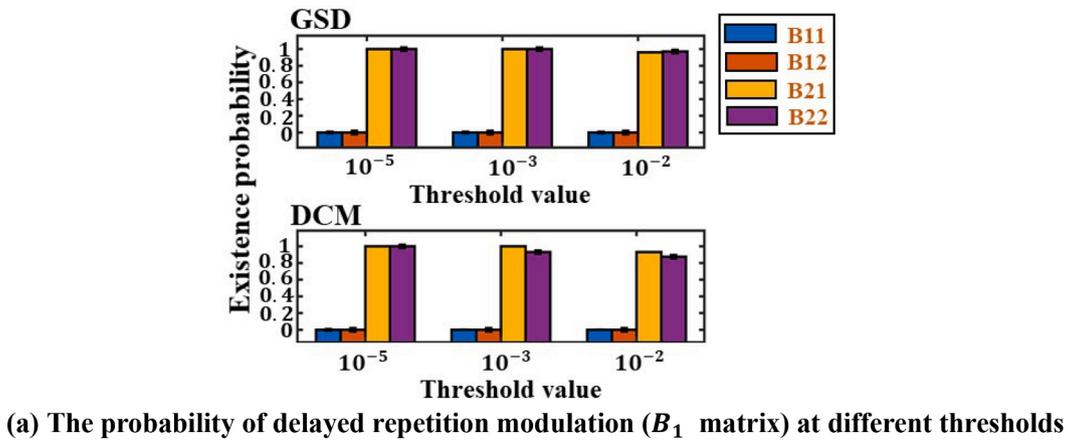


Fig. 24. The existence probability of input modulation at different thresholds.

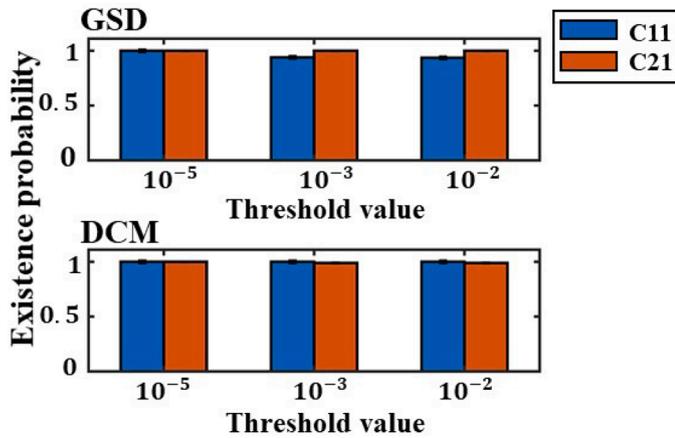


Fig. 25. The existence probability of driving input (C matrix) at different thresholds.

with the three connectivity models shown in Fig. 10. Both GSD and DCM select model (a) as the winning model. Fig. 22 demonstrates that GSD's computation time for the three models is much shorter than DCM's. Fig. 21(b) shows the final regression results. GSD and DCM both find that immediate and delayed repetition modulate the self-connection of FFA, and modulate the forward and backward connections between OFA and FFA, respectively. This conclusion is based on the probability of connectivity existence exceeding 0.95 for 16 subjects.

Figs. 23–25 display the probability of each connectivity parameter existing in the winning model of GSD and DCM at various thresholds (10^{-5} , 10^{-3} , 10^{-2}). Regardless of the threshold, the probabilities for

GSD and DCM remain high, except for the probability of connectivity B22 in the immediate repetition at the 10^{-2} threshold, which decreases.

7. Discussion

In this paper, we propose an algorithm that infers effective connectivity in more brain regions by combining sparse variational inference, converting complex-number equations to real-number ones, GLM filtering, frequency domain filtering, and other processing methods. The goal is to balance model interpretability and computational complexity, thereby improving both the computational speed and regression performance of frequency domain DCM. Testing on one set of simulation data and two sets of measured data shows that the proposed GSD algorithm not only reduces computational complexity but also demonstrates strong model interpretability.

From the experimental results, the computational speed of the GSD algorithm has significantly improved. One reason is that it performs real-number operations on the complex linear regression equation during parameter estimation, avoiding redundant complex operations typical of other frequency domain DCMs. Another reason is the convolution of the HRF on both sides of the original DCM equation, transforming the original model into a linear regression. This eliminates the need for GSD to estimate hidden neuron activity. In the simulation data set, GSD was compared with a sparse rDCM capable of inferring effective connectivity in large-scale brain networks. The regression speed of GSD is approximately 70 % faster than sparse rDCM and nearly 98 % faster than traditional DCM under various SNR or threshold settings. In the attention and motion dataset, GSD's regression speed is about 50 % faster than sparse rDCM and about 95 % faster than traditional DCM. In the face dataset, GSD's regression speed is about 90 % faster than traditional DCM. Although the improvement in regression speed for

measured data is not as large as in the simulation data, this indicates that GSD's speed advantage becomes more pronounced with an increasing number of brain regions, showing its potential for large-scale brain network research.

In terms of model interpretability, GSD did not sacrifice performance to improve computational speed. The traditional DCM algorithm was used as a baseline to illustrate GSD's efficiency. In the simulation data, we synthesized S50 and S20 network data. The S20 network, with fewer brain regions, allowed the traditional DCM algorithm to be tested, while the S50 network's large computational workload meant traditional DCM results were not provided. Results for the S50 network show that GSD maintains high specificity and sensitivity, accurately regressing the network structure. For the S20 network, GSD improves specificity and precision compared to traditional DCM and has relatively low estimation error. In both simulations, GSD outperforms sparse rDCM in estimation error and precision.

Although GSD showed efficient regression speed and strong model interpretability in simulation data, the simulation did not account for input modulation, so further testing was done on two groups of measured data affected by input modulation. In the attention and motion data, GSD's regression results for endogenous connectivity (**A** matrix) and input modulation (**B** matrix) were consistent with traditional DCM, showing the same effects (excitation or inhibition) and similar connectivity strengths. GSD and DCM differ slightly in endogenous connectivity details, but connectivity strengths in these areas are low, only reaching the order of 10^{-2} . Sparse rDCM, on the other hand, deletes most connections except self-connection and shows lower retained connectivity strength, likely due to its omission of input modulation to increase regression speed, thereby reducing model interpretability.

In the face data experiment, we focused on the effect of input modulation on brain area connectivity. Since sparse rDCM does not account for this effect, it was excluded from the comparison. From the model selection results, both GSD and traditional DCM identified the same winning models, and the analysis results for 16 subjects were consistent. The results show that both immediate and delayed repetition modulate the self-connection of FFA. Additionally, delayed repetition and immediate repetition modulate the forward and backward connectivity between OFA and FFA, respectively. When the threshold is 10^{-5} , the probability of these connections existing is almost 1. Moreover, the analysis results are consistent with the findings of the literature (Lee et al., 2022) from which the data is derived. Notably, the analysis method in (Lee et al., 2022) combines traditional DCM with the new parametric empirical Bayes (PEB) method, further validating GSD's regression reliability. It should be noted that literature (Lee et al., 2022) focused on four effects (face perception, immediate repetition, delayed repetition, and face recognition) while studying two ROIs (OFA and FFA). In contrast, this paper's experiment focuses on the two repetition effects because our primary interest is exploring the face RS effect, not the face perception or recognition effect. The regression results for face perception are more easily influenced by the number or position of extracted voxels in the ROI.

GLM filtering and hyperparameter optimization also play crucial roles in GSD's efficiency. The experimental results show that these methods significantly improve GSD's performance. In the S50 network, GSD maintains high sensitivity and specificity while achieving high regression speed. Its estimation error and precision remain high across various signal-to-noise ratio (SNR) environments, demonstrating strong noise resistance. Specifically, in a high SNR environment (SNR=100), GSD's precision is better. Analysis shows that GSD's precision is proportional to the SNR, which aligns with expected norms. In contrast, sparse rDCM's precision drops sharply at SNR=100, showing irregular fluctuations. This phenomenon, also observed in literature (Frässle et al., 2018), suggests that sparse rDCM's filtering technology fails in high SNR environments, highlighting GSD's robustness due to GLM filtering and parameter optimization.

For the attention and motion data, hyperparameters were set using optimization functions and empirical methods. Results showed that optimization functions resulted in lower fitting errors, indicating that empirical methods, being relatively fixed, lack flexibility for different situations. Specifically, empirical methods only perform low-pass filtering, whereas BOLD signal noise exists across various frequency bands. The limitation makes it difficult to filter out some noise effectively, affecting both fitting results and model selection. In the face dataset, while both GSD and traditional DCM selected the same winning model, the relative model evidence for GSD's winning model was significantly different from other models, unlike traditional DCM. This suggests that GSD may have a stronger model selection ability.

In summary, based on the above experimental results, the GSD algorithm proposed in this paper improves calculation speed in brain functional network regression without sacrificing model interpretability, achieving a balance between model complexity and interpretability. GSD effectively analyzes brain networks affected by input modulation and processes data from large-scale networks and complex experimental designs efficiently.

Despite GSD's strong performance in two measured datasets, there is room for further optimization. First, GSD does not process random effects in multi-subject analysis, and combining it with PEB to create a hierarchical model, as in reference (Lee et al., 2022), could improve its performance. Second, GSD could adopt a more flexible approach to setting the prior existence probability of brain region connections, as the sparsity of effective connectivity varies between regions. Finally, while GSD is inclined to analyze task-state fMRI data, it can theoretically analyze resting-state data, which can be explored further.

CRediT authorship contribution statement

Haifeng Wu: Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization. **Xinhang Hu:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Yu Zeng:** Investigation, Formal analysis.

Declaration of competing interest

The authors declare no competing financial interests or personal relationships that could have influenced the work reported in this paper.

Data and code availability statement

We validated the GSD algorithm using three public fMRI datasets: simulated Smith small-world network data, attention and motion measured data, and face recognition repetition effect measured data. The generation method of simulated Smith small-world network data has been integrated into the uploaded code. The acquisition addresses of the other two measured data sets are:

1. <http://www.fil.ion.ucl.ac.uk/spm/data/attention/>,
2. <https://www.openfmri.org/dataset/ds000117/>.

The code used in this study to test GSD performance has been publicly released. The copyright for the code belongs to the School of Electrical & Information Technology, Yunnan Minzu University. For any special requests, please contact Prof. Haifeng Wu at whf5469@gmail.com. The code is available at: <https://github.com/monk5469/GSD>.

Acknowledgements

The work was supported by funding from the National Natural Science Foundation of China (Grant numbers: 62161052), the Yunnan Provincial Department of Education Scientific Research Fund (2024Y432).

Ethics statement

The two public datasets (the attention and motion dataset and the face fMRI dataset) used in this work comply with relevant ethical and moral standards and have been approved by the Yunnan Minzu University Artificial Intelligence and Engineering Ethics Committee, under approval number 20240001.

Data and code availability statement

The method for generating the simulation data is included in the

code we have uploaded. The attention and motion dataset and the face fMRI dataset can be accessed at <http://www.fil.ion.ucl.ac.uk/spm/data/attention/> and <https://www.openfmri.org/dataset/ds000117/>.

The code used in this study is publicly available at <https://github.com/monk5469/GSD>. The copyright for the code belongs to the School of Electrical & Information Technology, Yunnan Minzu University. For inquiries, please contact Prof. Haifeng Wu at whf5469@gmail.com.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at [doi:10.1016/j.neuroimage.2024.120954](https://doi.org/10.1016/j.neuroimage.2024.120954).

Appendix

A.1 Derivation of the linear regression equation

Below we give the specific derivation process of converting the neural state equation into the linear regression equation of the observed state. In particular, the approximate processing process of the bilinear term (\mathbf{B} matrix) is given.

Convolution of HRF on both sides of (1) gives:

$$\dot{\mathbf{y}}_t = \mathbf{A}\boldsymbol{\psi}_t + \sum_{j=1}^K \mathbf{u}_t^j \mathbf{B}^j \mathbf{Z}_t \otimes h_t + \mathbf{C}\mathbf{U}_t \otimes h_t \quad (\text{A1-1})$$

From GLM (Worsley and Friston, 1995), i.e. (11), introducing the brain region symbol r , we can get

$$[\dot{\gamma}_t^{(1)} \dot{\gamma}_t^{(2)} \dots \dot{\gamma}_t^{(R)}] = \mathbf{x}_t^g [\boldsymbol{\beta}^{(1)} \boldsymbol{\beta}^{(2)} \dots \boldsymbol{\beta}^{(R)}] = (\mathbf{U}_t^T \otimes h_t) \mathbf{b} \quad (\text{A1-2})$$

where $\mathbf{b} = [\boldsymbol{\beta}^{(1)} \boldsymbol{\beta}^{(2)} \dots \boldsymbol{\beta}^{(R)}]$. Then using the convolution model, we can get:

$$[\dot{\gamma}_t^{(1)} \dot{\gamma}_t^{(2)} \dots \dot{\gamma}_t^{(R)}] = \mathbf{Z}_t^T \otimes h_t \quad (\text{A1-3})$$

From Eq.(A1-2) and Eq.(A1-3), we can know

$$\mathbf{Z}_t = \mathbf{b}^T \mathbf{U}_t \quad (\text{A1-3A})$$

Substituting Eq.(A1-3) into the third term on the left side of Eq.(A1-1), we have

$$\sum_{j=1}^K \mathbf{B}^j \mathbf{y}_t^j = \sum_{j=1}^K \mathbf{u}_t^j \mathbf{B}^j \mathbf{b}^T \mathbf{U}_t \otimes h_t$$

Then

$$\mathbf{y}_t^j = \mathbf{u}_t^j \mathbf{b}^T \mathbf{U}_t \otimes h_t$$

A.2 Transformation process of linear regression equation after GLM filtering

In the following, we explain in detail the differences and connections between the linear regression equation before and after GLM filtering.

Substituting the Fourier transform of (12) into the left side of (10), we have:

$$\dot{\mathbf{Y}}^o = \dot{\mathbf{Y}}^g + \dot{\boldsymbol{\Xi}} \quad (\text{A2-1})$$

where $\dot{\mathbf{Y}}^g = [\dot{\gamma}_n] \in \mathbb{C}^{N \times 1}$, $\dot{\gamma}_n$ is the Fourier transform of the GLM output derivative $\dot{\gamma}_t$; $\dot{\boldsymbol{\Xi}} = [\dot{\xi}_n] \in \mathbb{C}^{N \times 1}$, $\dot{\xi}_n$ is the Fourier transform of the derivative of ξ_n . Substituting the Fourier transform of (12) into the right side of (10), we have:

$$\mathbf{X}^o \boldsymbol{\theta} = \mathbf{X}^g \boldsymbol{\theta} + \boldsymbol{\Xi}^o \quad (\text{A2-2})$$

where

$$\mathbf{X}^g = \begin{bmatrix} \boldsymbol{\Psi}_1^g & \boldsymbol{\Psi}_2^g & \dots & \boldsymbol{\Psi}_N^g \\ \mathbf{Y}_1^1 & \mathbf{Y}_2^1 & \dots & \mathbf{Y}_N^1 \\ \mathbf{Y}_1^2 & \mathbf{Y}_2^2 & \dots & \mathbf{Y}_N^2 \\ \vdots & \vdots & \dots & \vdots \\ \mathbf{Y}_1^K & \mathbf{Y}_2^K & \dots & \mathbf{Y}_N^K \\ \mathbf{Y}'_1 & \mathbf{Y}'_2 & \dots & \mathbf{Y}'_N \end{bmatrix}^T \in \mathbb{C}^{N \times (R+RK+K)} \quad (\text{A2-3})$$

$$\Psi_n^g = [\Gamma_n^{(r)}] \in \mathbb{C}^{R \times 1}$$

$$\Xi^o = \begin{bmatrix} \Xi_1 & \Xi_2 & \dots & \Xi_N \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}^T \theta \in \mathbb{C}^{(R+RK+K) \times 1} \quad (\text{A2-3A})$$

$$\Xi_n = [\Xi_n^{(r)}] \in \mathbb{C}^{R \times 1}$$

$\Xi_n^{(r)}$ is the fourier transform of the derivative of $\xi_n^{(r)}$. Since Eq.(A2-1) and Eq.(A2-2) are equal, then

$$\hat{Y}^g = X^g \theta + \Xi^g$$

where $\Xi^g = \Xi^o - \hat{\Xi}$.

Data availability

The data and code supporting the findings of this study are available in the "Data and Code Availability Statement" file, which is attached with this submission.

References

- Bönstrup, M., Schulz, R., Feldheim, J., et al., 2016. Dynamic causal modelling of EEG and fMRI to characterize network architectures in a simple motor task. *Neuroimage* 124, 498–508. <https://doi.org/10.1016/j.neuroimage.2015.08.052>. Part A.
- Büchel, C., Friston, K.J., 1991. Modulation of connectivity in visual pathways by attention: cortical interactions evaluated with structural equation modelling and fMRI. *Cereb. Cortex* 7 (8), 768–778. <https://doi.org/10.1093/cercor/7.8.768>, 1997.
- Bishop, C.M., Nasrabadi, N.M., 2006. *Pattern Recognition and Machine learning*. Springer, New York.
- David O, Friston K J, 2003. A neural mass model for MEG/EEG: coupling and neuronal dynamics. *Neuroimage* 20 (3), 1743–1755. <https://doi.org/10.1016/j.neuroimage.2003.07.015>.
- Ewbank, M.P., Henson, R.N., Rowe, J.B., et al., 2013. Different neural mechanisms within occipitotemporal cortex underlie repetition suppression across same and different-size faces. *Cereb. Cortex* 23 (5), 1073–1084. <https://doi.org/10.1093/cercor/bhs070>.
- Frässle, S., Lomakina, E.I., Razi, A., et al., 2017. Regression DCM for fMRI. *Neuroimage* 155, 406–421. <https://doi.org/10.1016/j.neuroimage.2017.02.090>.
- Frässle, S., Lomakina, E.I., Kasper, L., et al., 2018. A generative model of whole-brain effective connectivity. *Neuroimage* 179, 505–529. <https://doi.org/10.1016/j.neuroimage.2018.05.058>.
- Frässle, S., Manjaly, Z.M., Do, C.T., et al., 2021. Whole-brain estimates of directed connectivity for human connectomics. *Neuroimage* 225, 117491. <https://doi.org/10.1016/j.neuroimage.2020.117491>.
- Frank, L.R., Buxton, R.B., Wong, E.C., 2001. Estimation of respiration-induced noise fluctuations from undersampled multislice fMRI data. *Magnetic Resonance in Medicine: an Official Journal of the International Society for Magnetic Resonance in Medicine* 45 (4), 635–644. <https://doi.org/10.1002/mrm.1086>.
- Friston, K., Mattout, J.E., 2007. Variational free energy and the Laplace approximation. *Neuroimage* 34 (1), 220–234. <https://doi.org/10.1016/j.neuroimage.2006.08.035>.
- Friston, K.J., Buechel, C., Fink, G.R., et al., 1997. Psychophysiological and modulatory interactions in neuroimaging. *Neuroimage* 6 (3), 218–229. <https://doi.org/10.1006/nimg.1997.0291>.
- Friston, K.J., Penny, W., Phillips, C., et al., 2002a. Classical and Bayesian inference in neuroimaging: theory. *Neuroimage* 16 (2), 465–483. <https://doi.org/10.1006/nimg.2002.1090>.
- Friston, K.J., Glaser, D.E., Henson, R.N., et al., 2002b. Classical and Bayesian inference in neuroimaging: applications. *Neuroimage* 16 (2), 484–512. <https://doi.org/10.1006/nimg.2002.1091>.
- Friston, K.J., Harrison, L., Penny, W., 2003. Dynamic causal modelling. *Neuroimage* 19 (4), 1273–1302. [https://doi.org/10.1016/S1053-8119\(03\)00202-7](https://doi.org/10.1016/S1053-8119(03)00202-7).
- Friston, K., Moran, R., Seth, A.K., 2013. Analysing connectivity with Granger causality and dynamic causal modelling. *Curr. Opin. Neurobiol.* 23 (2), 172–178. <https://doi.org/10.1016/j.conb.2012.11.010>.
- Friston, K. J., Frith C D, Turner R, et al., 1995. Characterizing evoked hemodynamics with fMRI. *Neuroimage* 2 (2), 157–165. <https://doi.org/10.1006/nimg.1995.1018>.
- Friston, K. J., Kahan, J., Biswal, B., et al., 2014. A DCM for resting state fMRI. *Neuroimage* 94, 396–407. <https://doi.org/10.1016/j.neuroimage.2013.12.009>.
- Friston, K.J., 2002. Bayesian estimation of dynamical systems: an application to fMRI. *Neuroimage* 16 (2), 513–530. <https://doi.org/10.1006/nimg.2001.1044>.
- Friston, K.J., 2011. Functional and effective connectivity: a review. *Brain Connect* 1 (1), 13–36. <https://doi.org/10.1089/brain.2011.0008>.

- Højen-Sørensen, P.A., Winther, O., HANSEN, L.K., 2002. Mean-field approaches to independent component analysis. *Neural Comput.* 14 (4), 889–918. <https://ieeexplore.ieee.org/abstract/document/6789937>.
- Hamilton J.D. Chapter 50 State-space models. *Handbook of Econometrics*, 1994, 4: 3039–3080. [https://doi.org/10.1016/S1573-4412\(05\)80019-4](https://doi.org/10.1016/S1573-4412(05)80019-4).
- Harrison, L., Penny, W.D., Friston, K., 2003. Multivariate autoregressive modeling of fMRI time series. *Neuroimage* 19 (4), 1477–1491. [https://doi.org/10.1016/S1053-8119\(03\)00160-5](https://doi.org/10.1016/S1053-8119(03)00160-5).
- Kiebel, S.J., Garrido, M.I., Moran, R.J., et al., 2008. Dynamic causal modelling for EEG and MEG. *Cogn Neurodyn* 2, 121–136. <https://doi.org/10.1007/s11571-008-9038-0>.
- Klingner, C.M., Brodoehl, S., Huonker, R., et al., 2015. Parallel processing of somatosensory information: evidence from dynamic causal modeling of MEG data. *Neuroimage* 118, 193–198. <https://doi.org/10.1016/j.neuroimage.2015.06.028>.
- Lee, S.M., Tibon, R., Zeidman, P., et al., 2022. Effects of face repetition on ventral visual stream connectivity using dynamic causal modelling of fMRI data. *Neuroimage* 264, 119708. <https://doi.org/10.1016/j.neuroimage.2022.119708>.
- Liang, Q., Li, J., Zheng, S., et al., 2022. Dynamic causal modelling of hierarchical planning. *Neuroimage* 258, 119384. <https://doi.org/10.1016/j.neuroimage.2022.119384>.
- Lin Tiger, W., Chen, Y.S., Bukhari, Q., et al., 2020. Differential Covariance: A New Method to Estimate Functional Connectivity in fMRI. *Neural Comput.* 32 (12), 2389–2421. https://doi.org/10.1162/neco_a_01323.
- Lindley, D.V., 1956. On a Measure of the Information Provided by an Experiment. *Ann. Math. Statist.* 4, 986–1005. <http://gdmtest.u-ga.fr/item/1177728069/>.
- McIntosh, A., Gonzalez-Lima, F., 1994. Structural equation modeling and its application to network analysis in functional brain imaging. *Hum. Brain Mapp.* 2 (1–2), 2–22. <https://doi.org/10.1002/hbm.460020104>.
- Moon, T.K., 1996. The expectation-maximization algorithm. *IEEE Signal Process. Mag.* 13 (6), 47–60. <https://ieeexplore.ieee.org/abstract/document/543975>.
- Novelli, L., Friston, K., Razi, A., 2024. Spectral dynamic causal modeling: a didactic introduction and its relationship with functional connectivity. *Network Neurosci.* 8 (1), 178–202. https://doi.org/10.1162/netn_a_00348.
- Ostwald, D., Starke, L., 2016. Probabilistic delay differential equation modeling of event-related potentials. *Neuroimage* 136, 227–257. <https://doi.org/10.1016/j.neuroimage.2016.04.025>.
- Ou, Y., Dai, P., Zhou, X., et al., 2022. A strategy of model space search for dynamic causal modeling in task fMRI data exploratory analysis. *Phys. Eng. Sci. Med.* 45 (3), 867–882. <https://doi.org/10.1007/s13246-022-01156-w>.
- Poline, J.B., Brett, M., 2012. The general linear model and fMRI: does love last forever? *Neuroimage* 62 (2), 871–880. <https://doi.org/10.1016/j.neuroimage.2012.01.133>.
- Prando, G., Zorzi, M., Bertoldo, A., et al., 2020. Sparse DCM for whole-brain effective connectivity from resting-state fMRI data. *Neuroimage* 208, 116367. <https://doi.org/10.1016/j.neuroimage.2019.116367>.
- Razi, A., Seghier, M.L., Zhou, Y., et al., 2017. Large-scale DCMs for resting-state fMRI. *Network Neuroscience* 1 (3), 222–241. https://doi.org/10.1162/NETN_a_00015.
- Rigoux, L., Daunizeau, J., 2015. Dynamic causal modelling of brain-behaviour relationships. *Neuroimage* 117, 202–221. <https://doi.org/10.1016/j.neuroimage.2015.05.041>.
- Sahoo, D., Honnorat, N., Davatzikos, C., 2019. Sparse low-dimensional causal modeling for the analysis of brain function[C]/Medical Imaging 2019. Image Processing. SPIE, Bellingham, WA, pp. 701–707. <https://doi.org/10.1117/12.2512542>.
- Sato, J.R., Takahashi, D.Y., Arcuri, S.M., et al., 2009. Frequency domain connectivity identification: an application of partial directed coherence in fMRI. *Hum. Brain Mapp.* 30 (2), 452–461. <https://doi.org/10.1002/hbm.20513>.
- Seghier, M.L., Friston, K.J., 2013. Network discovery with large DCMs. *Neuroimage* 68, 181–191. <https://doi.org/10.1016/j.neuroimage.2012.12.005>.
- Shi, Y., Li, Y., 2024. The effective connectivity analysis of fMRI based on asymmetric detection of transfer brain entropy. *Cereb. Cortex* 34 (3). <https://doi.org/10.1093/cercor/bhae070>.
- Smith, S.M., Miller, K.L., Salimi-Khorshidi, G., et al., 2011. Network modelling methods for FMRI. *Neuroimage* 54 (2), 875–891. <https://doi.org/10.1016/j.neuroimage.2010.08.063>.

- Sotero, R.C., Sanchez-Bornot, J.M., 2024. Shaharabi-Farahani I. Parameter estimation in brain dynamics models from resting-state fMRI data using physics-informed neural networks[J/OL]. *Biorxiv*. <https://doi.org/10.1101/2024.02.27.582428>.
- Stephan, K.E., Roebroeck, A., 2012. A short history of causal modeling of fMRI data. *Neuroimage* 62 (2), 856–863. <https://doi.org/10.1016/j.neuroimage.2012.01.034>.
- Stephan, K.E., Kasper, L., Harrison, L.M., et al., 2008. Nonlinear dynamic causal models for fMRI. *Neuroimage* 42 (2), 649–662. <https://doi.org/10.1016/j.neuroimage.2008.04.262>.
- Wakeman, D.G., Henson, R.N., 2015. A multi-subject, multi-modal human neuroimaging dataset. *Sci. Data* 2 (1), 1–10. <https://doi.org/10.1038/sdata.2015.1>.
- Wang, W., Zhang, G., Yang, L., et al., 2019. Revisiting signal processing with spectrogram analysis on EEG, ECG and speech signals. *Future Generat. Comput. Syst.* 98, 227–232. <https://doi.org/10.1016/j.future.2018.12.060>.
- Worsley, K.J., Friston, K.J., 1995. Analysis of fMRI time-series revisited—Again. *Neuroimage* 2 (3), 173–181. <https://doi.org/10.1006/nimg.1995.1023>.
- Zetterberg, L.H., Kristiansson, L., Mossberg, K., 1978. Performance of a model for a local neuron population. *Biol. Cybern.* 31, 15–26. <https://doi.org/10.1007/BF00337367>.
- Zhou, S.K., Greenspan, H., Shen, D., 2024. *Deep Learning For Medical Image analysis[M]*, 2nd ed. Academic Press, San Diego.