Hierarchical multiscale Bayesian algorithm for robust MEG/EEG source reconstruction

Chang Cai[a], Kensuke Sekihara[b,c], Srikantan S Nagarajan[a]

[a]*Department of Radiology and Biomedical Imaging, University of California, San Francisco, CA 94143-0628 e-mail: sri@ucsf.edu*
[b]*Department of Advanced Technology in Medicine, Tokyo Medical and Dental University, 1-5-45 Yushima, Bunkyo-ku, Tokyo 113-8519, Japan*
[c]*Signal Analysis Inc., Hachioji, Tokyo*

Abstract

In this paper, we present a novel hierarchical multiscale Bayesian algorithm for electromagnetic brain imaging using magnetoencephalography (MEG) and electroencephalography (EEG). In particular, we present a solution to the source reconstruction problem for sources that vary in spatial extent. We define sensor data measurements using a generative probabilistic graphical model that is hierarchical across spatial scales of brain regions and voxels. We then derive a novel Bayesian algorithm for probabilistic inference with this graphical model. This algorithm enables robust reconstruction of sources that have different spatial extent, from spatially contiguous clusters of dipoles to isolated dipolar sources. We test new algorithms with several representative benchmarks on both simulated and real brain activities. The source locations and the correct estimation of source time courses used for the simulated data are chosen to test the performance on challenging source configurations. In simulations, performance of the novel algorithm shows superiority to several existing benchmark algorithms. We also demonstrate that the new algorithm is more robust to correlated brain activity present in real MEG and EEG data and is able to resolve distinct and functionally relevant brain areas with real MEG and EEG datasets.

Keywords:
Brain Mapping, Magnetoencephalography, Electroencephalography, Bayesian.

1. Introduction

Mapping of the entire brain's activity in humans is an important undertaking in cognitive neuroscience research that seeks to understand neural mechanisms of complex human behaviors. It also has clinical applications in patients with brain tumors and epilepsy, where functional brain mapping is useful to guide neurosurgical planning, navigation, and resection.

Two techniques currently exist for non-invasive brain mapping of electrophysiological activity in humans: electroencephalography (EEG) and magnetoencephalography (MEG). MEG and EEG are complementary techniques that

measure, respectively, the magnetic field outside the head and the scalp electric potentials produced by electrical activity in neural cell assemblies. Since they directly measure electrical brain activity from neural ensembles, these methods have superior temporal resolution compared to PET or fMRI, thereby enabling studies of the dynamics of neural ensembles that occur at typical time scales on the order of tens of milliseconds.

To estimate brain source activity from EEG or MEG data, source reconstruction algorithms are necessary, which consists of solving a forward problem and an inverse problem. The forward problem computes the scalp potentials and external magnetic fields for a specific set of neural current sources for a given sensor configuration, brain anatomy, head geometries, and volume conductor properties. The inverse problem estimates the parameters of neural sources from MEG and EEG sensor data and makes use of the forward problem computations. The estimation of spatial locations and timing of brain sources is still a challenging problem because it involves solving for unknown brain activity across thousands of voxels from the recordings of just a few hundred sensors. In general, there are no unique solutions to the inverse problem because there are many source configurations that could produce sensor data that can account for the sensor observations. This nonuniqueness is referred to as the ill-posed nature of the inverse problem. Besides handling the ill-posed nature of EEG or MEG imaging, the inverse algorithms have to address the challenge of searching for true source signals while minimizing the many sources of noise that interfere with the true signals. Electrical, thermal and biological noise as well as background room interference can be present.

To overcome these challenges, researchers have proposed many efficient inverse problem algorithms which can broadly be classified into two categories: model-based parametric dipole fitting and whole-brain source imaging methods. Dipole fitting methods assume that a small set of current dipoles can adequately represent an unknown source distribution, which is a direct way to estimate source parameters and has properties of high resolution but low accuracy. This is because solving for dipole parameters requires nonlinear optimization over a high-dimensional parameter space with solutions having great sensitivity to initialization due to the high probability of being a local minima. This is especially a significant problem when multiple dipoles are considered. Furthermore, estimating the number of dipoles remains an intractable problem.

An alternative approach is whole-brain source imaging methods which do not require prior knowledge of the number of sources and can generally avoid the non-linear search in the high dimensional parameter space [1, 2, 3, 4]. These methods apply voxel discretization over a whole brain volume, and assume a source at each voxel and estimate the amplitudes (and orientation) of the sources by minimizing a cost function. Imaging methods can be further classified into two classes: tomographic reconstruction and spatial scanning techniques. Tomographic techniques model the activity at all candidate source locations simultaneously. Tomographic techniques include minimum-norm estimation (MNE) [5, 6], dynamic statistical parametric mapping (dSPM) [7], and standardized low resolution brain electromagnetic tomography (sLORETA) [8]. Some tomo-

graphic techniques promote sparseness in the solution [9, 10], where the majority of the candidate locations do not have significant activity [4, 11, 12, 13]. Empirical evidence shows that a sparse source model can improve the accuracy of the localization in a noisy environment [13]. In contrast, spatial scanning techniques sequentially estimate the time course at every candidate location while suppressing the interference from activity at the other candidate source locations. Some examples of scanning techniques are minimum-variance adaptive beamforming [14, 15, 16, 17] and other variants of beamformers [1].

Most of the source reconstruction algorithms from the above classes can be viewed in Bayesian framework [2]. This perspective is useful because at a high level, the prior distribution, implicitly or explicitly imposed, can be used to differentiate and compare the various source localization methods. Recently, we have developed Champagne, a novel tomographic source reconstruction algorithm that is derived in an empirical Bayesian and incorporates deep theoretical ideas about sparse-source recovery from noisy, constrained measurements. Champagne improves upon existing methods of source reconstruction in terms of reconstruction accuracy, robustness, and computational efficiency [13]. Experiments with preliminary simulated and real data, presented in [18], show that compared to other commonly-used source localization algorithms, Champagne is more robust to correlated sources and noisy data. However, when faced with more complex brain activity patterns that span multiple spatial scales, such as clusters of dipolar sources or mixtures of clusters and isolated dipolar sources, there are still no efficient source reconstruction algorithms.

Here, we present a novel hierarchical multiscale generative model for electromagnetic measurements such as MEG and EEG. This algorithm can be considered as a hierarchical multiscale extension of the Champagne algorithm. We first assume that brain voxels cluster into either anatomically or functionally defined brain regions or parcels with region-level specific variances. Voxel activity is then assumed to have a component arising from regions with additional voxel specific variances to account for variations in voxel activity within a region. The voxel activity is then assumed to be related to sensor data using standard lead-field kernels that are known given the geometry of the sensor measurements and the volume conductor model. We then derive Bayesian algorithm for estimating voxel and region variances from sensor data. We present a novel algorithm with both voxel and region variances, referred to as tree_Champagne. We evaluate its performance in simulations and real-data and compare with existing benchmark algorithms.

2. Methods

This section describes the tree_Champagne algorithm including the probabilistic generative model, estimation of the source and region activity, learning of hyperparameters, and its relation to other Bayesian inference algorithms.

2.1. The probabilistic generative model

We assume that MEG/EEG data have been collected for evoked or induced source activity paradigms, with separate time-windows for evoked or induced source activity and for background brain activity including interference from biological, environmental sources and sensor noise.

The generative model for the sensor data is:

$$\boldsymbol{y}(t) = \sum_{i=1}^{N} \boldsymbol{l}_i \boldsymbol{s}_i(t) + \varepsilon \tag{1}$$

where, $\boldsymbol{y}(t) \in \mathbb{R}^{d_y \times 1}$, is the output data of sensors at time t, d_y is the number of channels measured, N is the number of voxels under consideration and $\boldsymbol{l}_i \in \mathbb{R}^{d_y \times d_c}$ is the lead-field matrix for i-th voxel. The k-th column of $\boldsymbol{l}_i$ represents the signal vector that would be observed at the scalp given a unit current source/dipole at the i-th voxel with a fixed orientation in the k-th direction. It is common to assume $d_c = 2$ (for MEG) or $d_c = 3$ (for EEG), which allows flexible source orientations to be estimated in 2D or 3D space. Multiple methods based on the physical properties of the brain and Maxwell's equations are available for the computation of each $\boldsymbol{l}_i$ [19]. And $\boldsymbol{s}_i(t) \in \mathbb{R}^{d_c \times 1}$ is the ith voxel intensity at time t, which we assume it with d_c orientations. Finally, ε is a noise-plus-interference term where we assume, for simplicity, that the columns are drawn independently from $N(0, \Sigma_\varepsilon)$ with known covariance Σ_ε. Temporal correlations can easily be incorporated if desired using a simple transformation outlined in [20] or using the spatio-temporal framework introduced in [21]. Here, we assume that the noise covariance can be estimated from the baseline and evoked data using a Stimulus-Evoked Factor Analysis, SEFA [22] or variational Bayesian factor analysis (VBFA) model [23].

In our hierarchical framework, we divide the brain into R apriori regions (or tiles) specified either anatomically or functionally [24]. The j-th region contains p_j voxels. As a first step, we assume that the division of regions are assumed to be non-overlapping, where each voxel belongs to exactly one region, but this is not a necessity in the framework. Regional tiling may correspond to a map of anatomical or functional areas, or be constructed by, e.g., dividing the voxels into regions centered at equally-spaced locations throughout the brain [25]. We also assume that each unknown region's activity $\boldsymbol{z}_j(t) \in \mathbb{R}^{d_c \times 1}$ at time t is an equivalent d_c-dimensional neural current dipole, projecting from the j-th region. We then assume that a given voxel's activity arises from the addition of the region's activity and voxel activity that is independent of the region's, as shown below.

$$\boldsymbol{s}_i(t) = \boldsymbol{v}_i(t) + g_j \boldsymbol{z}_j(t) \tag{2}$$

In the equation above, $\boldsymbol{v}_i$ expresses the component that is intrinsic to the i-th voxel and independent from activities of other voxels or the region a voxel belongs to. g_j is the gain matrix between j-th region distribution and voxel $\boldsymbol{s}_i$, here we assume it to be $\frac{1}{p_j}$, where p_j is the number of voxels for j-th region. Then, the source data model in Eq. (1) is expressed such that

$$\boldsymbol{y}(t) = \sum_{i=1}^{N} \boldsymbol{l}_i \boldsymbol{v}_i(t) + \sum_{j=1}^{R} \left(\frac{1}{p_j} \sum_{i \in \omega_j} \boldsymbol{l}_i \right) \boldsymbol{z}_j(t) + \varepsilon = \sum_{i=1}^{N} \boldsymbol{l}_i \boldsymbol{v}_i(t) + \sum_{j=1}^{R} \bar{\boldsymbol{l}}_j \boldsymbol{z}_j(t) + \varepsilon \quad (3)$$

where $\sum_{i \in \omega_j}$ indicates the summation regarding the voxels that belong to the jth region. We then denote the mean lead field over the jth region by $\bar{\boldsymbol{l}}_j$: $\bar{\boldsymbol{l}}_j = 1/p_j \sum_{i \in \omega_j} \boldsymbol{l}_i$ and define an extended (voxel-augumented) lead field matrix $\boldsymbol{H}$ such that

$$\boldsymbol{H} = \left[\boldsymbol{l}_1, \ldots, \boldsymbol{l}_N, \bar{\boldsymbol{l}}_1, \ldots, \bar{\boldsymbol{l}}_R \right] = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_{N+R}] \quad (4)$$

where $\boldsymbol{h}_i = \boldsymbol{l}_i$ for $i = 1, \ldots, N$ and $\boldsymbol{h}_i = \bar{\boldsymbol{l}}_{i-N}$ for $i = N+1, \ldots, N+R$. We also define an extended voxel vector, such that

$$\boldsymbol{x}(t) = \left[\boldsymbol{v}_1^T(t), \ldots, \boldsymbol{v}_N^T(t), \boldsymbol{z}_1^T(t), \ldots, \boldsymbol{z}_R^T(t) \right]^T = \left[\boldsymbol{x}_1^T(t), \ldots, \boldsymbol{x}_{N+R}^T(t) \right]^T \quad (5)$$

where $\boldsymbol{x}_i(t) = \boldsymbol{v}_i(t)$ for $i = 1, \ldots, N$ and $\boldsymbol{x}_i(t) = \boldsymbol{z}_{i-N}(t)$ for $i = N+1, \ldots, N+R$. Eq. (3) can then be rewritten as

$$\boldsymbol{y}(t) = \boldsymbol{H}\boldsymbol{x}(t) + \varepsilon \quad (6)$$

The equation above is the data model used for the derivation of new algorithms.

The data vector $\boldsymbol{y}(t_k)$ is denoted $\boldsymbol{y}_k$ and the extended voxel vector $\boldsymbol{x}(t_k)$ is denoted $\boldsymbol{x}_k$ for simplicity, t_k is the time point at k. We formulate the source reconstruction problem as the spatio-temporal reconstruction, i.e., the voxel time series $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K$ is reconstructed using the sensor time series $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K$. We express the whole time series $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K$ collectively as $\boldsymbol{X} \in \mathbb{R}^{(N+R)d_c \times K}$, and the whole time series $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_K$ as $\boldsymbol{Y} \in \mathbb{R}^{d_y \times K}$.

We then define $\boldsymbol{\Upsilon}_i$ as a prior variance $d_c \times d_c$ matrix of $\boldsymbol{x}_i$ and define $\boldsymbol{\Upsilon}$ as $d_c(N+R) \times d_c(N+R)$ block diagonal matrix expressed as

$$\boldsymbol{\Upsilon} = \begin{bmatrix} \boldsymbol{\Upsilon}_1 & 0 & \cdots & 0 \\ 0 & \boldsymbol{\Upsilon}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{\Upsilon}_{N+R} \end{bmatrix} \quad (7)$$

The prior distribution is expressed as

$$p(\boldsymbol{X}|\boldsymbol{\Upsilon}) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{x}_k|\boldsymbol{0}, \boldsymbol{\Upsilon}) \quad (8)$$

Using the noise assumption that $\varepsilon \sim \mathcal{N}(\varepsilon|0, \boldsymbol{\Sigma}_\varepsilon)$, the conditional probability $p(\boldsymbol{Y}|\boldsymbol{X})$ is expressed as

$$p(\boldsymbol{Y}|\boldsymbol{X}) = \prod_{k=1}^{K} p(\boldsymbol{y}_k|\boldsymbol{x}_k) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{y}_k|\boldsymbol{H}\boldsymbol{x}_k, \boldsymbol{\Sigma}_\varepsilon) \quad (9)$$

Here, the noise covariance $\boldsymbol{\Sigma}_\varepsilon$ can be estimated using SEFA [22] or VBFA [23] and is assumed to be known for simplicity and subsequent considerations.

2.2. Estimation of the source and region activity

To estimate the source distribution $\boldsymbol{X}$, we first derive the posterior distribution $p(\boldsymbol{X}|\,\boldsymbol{Y})$, which is given by

$$p(\boldsymbol{X}|\,\boldsymbol{Y}) = \prod_{k=1}^{K} p(\boldsymbol{x}_k|\boldsymbol{y}_k) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{x}_k|\bar{\boldsymbol{x}}_k, \boldsymbol{\Gamma}^{-1}) \tag{10}$$

where the variance and the mean are obtained as

$$\boldsymbol{\Gamma}^{-1} = \boldsymbol{\Upsilon}^{-1} + \boldsymbol{H}^T \boldsymbol{\Sigma}_{\varepsilon}^{-1} \boldsymbol{H} \tag{11}$$

$$\bar{\boldsymbol{x}}_k = \boldsymbol{\Gamma}^{-1} \boldsymbol{H}^T \boldsymbol{\Sigma}_{\varepsilon}^{-1} \boldsymbol{y}_k \tag{12}$$

The posterior mean can be written in an alternative way, such that

$$\bar{\boldsymbol{x}}_k = \boldsymbol{\Upsilon} \boldsymbol{H}^T \left(\boldsymbol{\Sigma}_{\varepsilon} + \boldsymbol{H} \boldsymbol{\Upsilon} \boldsymbol{H}^T \right)^{-1} \boldsymbol{y}_k = \boldsymbol{\Upsilon} \boldsymbol{H}^T \boldsymbol{\Sigma}_y^{-1} \boldsymbol{y}_k \tag{13}$$

where

$$\boldsymbol{\Sigma}_y = \boldsymbol{\Sigma}_{\varepsilon} + \boldsymbol{H} \boldsymbol{\Upsilon} \boldsymbol{H}^T \tag{14}$$

This $\boldsymbol{\Sigma}_y$ is called the model data covariance matrix. The solution in Eq. (13) can be expressed as

$$\begin{bmatrix} \overline{\boldsymbol{x}}_1(t_k) \\ \overline{\boldsymbol{x}}_2(t_k) \\ \vdots \\ \overline{\boldsymbol{x}}_{N+R}(t_k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Upsilon}_1 & 0 & \cdots & 0 \\ 0 & \boldsymbol{\Upsilon}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{\Upsilon}_{N+R} \end{bmatrix} \begin{bmatrix} \boldsymbol{h}_1^T \\ \boldsymbol{h}_2^T \\ \vdots \\ \boldsymbol{h}_{N+R}^T \end{bmatrix} \boldsymbol{\Sigma}_y^{-1} \boldsymbol{y}_k \tag{15}$$

We can then express the source activity in terms of a spatial filter as shown below:

$$\overline{\boldsymbol{x}}_j(t_k) = \boldsymbol{\Upsilon}_j \boldsymbol{h}_j^T \boldsymbol{\Sigma}_y^{-1} \boldsymbol{y}_k \tag{16}$$

2.3. Learning the hyperparameters $\boldsymbol{\Upsilon}$

The Bayesian estimate of $\boldsymbol{x}_k$ is given as the voxel posterior mean in Eq. (12) or (16). In order to compute $\boldsymbol{x}_k$ in Eq. (16), we need to know the hyperparameter $\boldsymbol{\Upsilon}$. The hyperparameter $\boldsymbol{\Upsilon}$ is obtained by maximizing $p(\boldsymbol{Y}|\boldsymbol{\Upsilon})$ which is called the marginal likelihood [26]. The marginal likelihood $p(\boldsymbol{Y}|\boldsymbol{\Upsilon})$ is expressed as follows (details of the derivation of this function can be found in Appendix A).

$$\begin{aligned} \log p(\boldsymbol{Y}|\boldsymbol{\Upsilon}) = -\frac{1}{K} \sum_{k=1}^{K} \Bigg[& \left(\boldsymbol{y}_k - \boldsymbol{H}\overline{\boldsymbol{x}}_k\right)^T \boldsymbol{\Sigma}_{\varepsilon}^{-1} \left(\boldsymbol{y}_k - \boldsymbol{H}\overline{\boldsymbol{x}}_k\right) + \sum_{j=1}^{N+R} \overline{\boldsymbol{x}}_j^T(t_k) \boldsymbol{\Upsilon}_j^{-1} \overline{\boldsymbol{x}}_j(t_k) \Bigg] \\ & - \log |\boldsymbol{\Sigma}_y| \end{aligned} \tag{17}$$

Although the optimum value of the hyperparameter $\boldsymbol{\Upsilon}$ is obtained by maximizing $\log p(\boldsymbol{Y}|\boldsymbol{\Upsilon})$, maximizing the right-hand side of the equation above is difficult due to the inclusion of the last term $\log|\boldsymbol{\Sigma}_y|$.

Since $\log|\boldsymbol{\Sigma}_y|$ is a concave function with respect to $\boldsymbol{\Upsilon}$ [4], using $d_c \times d_c$ auxiliary-parameter matrices, $\boldsymbol{\Lambda}_j (j = 1, \ldots, N+R)$, the relationship [27] [28],

$$\sum_{j=1}^{N+R} tr\left(\boldsymbol{\Lambda}_j^T \boldsymbol{\Upsilon}_j\right) - \Lambda_0 \geq \log|\boldsymbol{\Sigma}_y| \tag{18}$$

hold where Λ_0 is scalar term. Accordingly, we define an auxiliary cost function $\mathcal{F}(\boldsymbol{\Upsilon}, \boldsymbol{\Lambda})$ such that

$$\begin{aligned}\mathcal{F}(\boldsymbol{\Upsilon}, \boldsymbol{\Lambda}) = -\frac{1}{K}\sum_{k=1}^{K}\left[(\boldsymbol{y}_k - \boldsymbol{H}\overline{\boldsymbol{x}}_k)^T \boldsymbol{\Sigma}_\varepsilon^{-1}(\boldsymbol{y}_k - \boldsymbol{H}\overline{\boldsymbol{x}}_k) + \sum_{j=1}^{N+R}\overline{\boldsymbol{x}}_j^T(t_k)\boldsymbol{\Upsilon}_j^{-1}\overline{\boldsymbol{x}}_j(t_k)\right] \\ - \sum_{j=1}^{N+R} tr\left(\boldsymbol{\Lambda}_j^T \boldsymbol{\Upsilon}_j\right) + \Lambda_0\end{aligned} \tag{19}$$

where,

$$\log p(\boldsymbol{Y}|\boldsymbol{\Upsilon}) \geq \mathcal{F}(\boldsymbol{\Upsilon}, \boldsymbol{\Lambda}) \tag{20}$$

always hold, and increasing $\mathcal{F}(\boldsymbol{\Upsilon}, \boldsymbol{\Lambda})$ with respect to $\boldsymbol{\Upsilon}$ should result in increasing the marginal likelihood $\log p(\boldsymbol{Y}|\boldsymbol{\Upsilon})$. Therefore, the update value of $\boldsymbol{\Upsilon}$ is derived as

$$\hat{\boldsymbol{\Upsilon}} = \arg\max_{\boldsymbol{\Upsilon}} \mathcal{F}(\boldsymbol{\Upsilon}, \boldsymbol{\Lambda}) \tag{21}$$

Update rules can then be derived using

$$\frac{\partial}{\partial \boldsymbol{\Upsilon}_j}\mathcal{F}(\boldsymbol{\Upsilon}_j, \boldsymbol{\Lambda}) = -\boldsymbol{\Upsilon}_j^{-1}\left[\frac{1}{K}\sum_{k=1}^{K}\overline{\boldsymbol{x}}_j(t_k)\overline{\boldsymbol{x}}_j^T(t_k)\right]\boldsymbol{\Upsilon}_j^{-1} + \boldsymbol{\Lambda}_j = \boldsymbol{0} \tag{22}$$

. Setting the right-hand side to zero, we get the equation,

$$\boldsymbol{\Upsilon}_j \boldsymbol{\Lambda}_j \boldsymbol{\Upsilon}_j = \left[\frac{1}{K}\sum_{k=1}^{K}\overline{\boldsymbol{x}}_j(t_k)\overline{\boldsymbol{x}}_j^T(t_k)\right] \tag{23}$$

A positive semi-definite matrix that satisfies Eq. (23), can be derived such that

$$\widehat{\boldsymbol{\Upsilon}}_j = \boldsymbol{\Lambda}_j^{-1/2}\left[\boldsymbol{\Lambda}_j^{1/2}\left[\frac{1}{K}\sum_{k=1}^{K}\overline{\boldsymbol{x}}_j(t_k)\overline{\boldsymbol{x}}_j^T(t_k)\right]\boldsymbol{\Lambda}_j^{1/2}\right]^{1/2}\boldsymbol{\Lambda}_j^{-1/2} \tag{24}$$

Eq. (24) is the update rule for $\boldsymbol{\Upsilon}_j$.

The update rule for $\boldsymbol{\Lambda}_j$ is derived using a fact that the hyper plane $\sum_{j=1}^{N+R} tr\left(\widehat{\boldsymbol{\Lambda}}_j^T \boldsymbol{\Upsilon}_j\right) - \Lambda_0$ forms a tightest upper bound of the concave function $\log|\boldsymbol{\Sigma}_y|$. Such a hyperplane is found as the plane that is tangential to $\log|\boldsymbol{\Sigma}_y|$. Therefore, the update equation for $\boldsymbol{\Lambda}_j$ is derived as

$$\widehat{\boldsymbol{\Lambda}}_j = \frac{\partial}{\partial \boldsymbol{\Upsilon}_j}\log|\boldsymbol{\Sigma}_y| = \boldsymbol{h}_j^T \boldsymbol{\Sigma}_y^{-1}\boldsymbol{h}_j \tag{25}$$

In summary, the hyperparameter $\boldsymbol{\Upsilon}_j$ are estimated by iterating Eq. (16), Eq. (24) and Eq. (25). Each iteration is theoretically guaranteed to increase (or leave unchanged) the cost function $\mathcal{F}(\boldsymbol{\Upsilon}_j, \boldsymbol{\Lambda})$. The per-iteration cost is linear in the number of $N + R$ so the computational cost is relatively modest (it is quadratic in d_y, and cubic in d_c, but these quantities are relatively small). The convergence rate is orders of magnitude faster [4] than Expectation Maximum (EM) based algorithms such as those in [20, 29].

2.4. Algorithm summary

Tree_Champagne is a source reconstruction algorithm based on generative model Eq. (1) and is able to combine sparsity (from voxel level inference) and smoothness (from regional-level inference) to produce optimally smooth and sparse solutions.

Using the updating rules above, we can calculate the variance of both voxels and regions . We denote variance of the voxel intrinsic component as $\boldsymbol{\Upsilon}^V$ and variance of the regions as $\boldsymbol{\Upsilon}^R$, the relationship between $\boldsymbol{\Upsilon}$ and $\boldsymbol{\Upsilon}^V$, $\boldsymbol{\Upsilon}^R$ is

$$\boldsymbol{\Upsilon} = \begin{bmatrix} \boldsymbol{\Upsilon}^V & 0 \\ 0 & \boldsymbol{\Upsilon}^R \end{bmatrix} \tag{26}$$

where

$$\begin{aligned} \boldsymbol{\Upsilon}^V &= \begin{bmatrix} \boldsymbol{\Upsilon}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \boldsymbol{\Upsilon}_N \end{bmatrix} \\ \boldsymbol{\Upsilon}^R &= \begin{bmatrix} \boldsymbol{\Upsilon}_{N+1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \boldsymbol{\Upsilon}_{N+R} \end{bmatrix} \end{aligned} \tag{27}$$

The variance of i-th voxel is treated as the summation of the variance of the i-th voxel's intrinsic component and variance of the region where the i-th voxel belongs to. The time course of tree_Champagne can be expressed as:

$$\hat{\boldsymbol{s}}_i^{tree}(t_k) = \boldsymbol{\Upsilon}_i^{tree} \boldsymbol{h}_i^T \boldsymbol{\Sigma}_y^{-1} \boldsymbol{y}_k \tag{28}$$

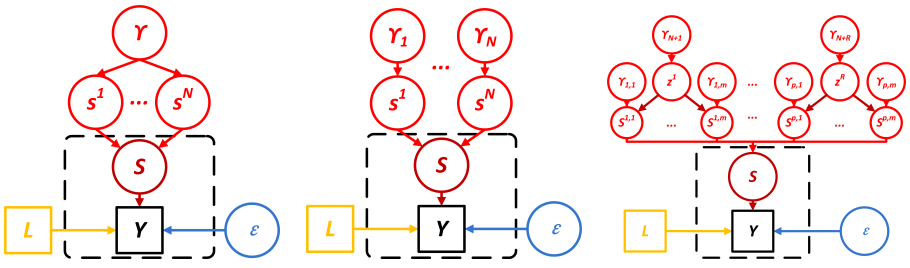
where i-th voxel's variance $\boldsymbol{\Upsilon}_i^{tree}$ is expressed as

$$\boldsymbol{\Upsilon}_i^{tree} = \boldsymbol{\Upsilon}_i^V + \boldsymbol{\Upsilon}_j^R \tag{29}$$

where the i-th voxel belongs to j-th region.

2.5. Algorithm Initialization

Initialization of the parameter updates are described here. First, $\boldsymbol{\Sigma}_\varepsilon$ is learned from the pre-stimulus period using SEFA [22] or VBFA [23] and fixed.



(a) BMN (b) Champagne (c) Tree_Champagne

Figure 1: Graphical models for (a) BMN, (b) Champagne, (c) Tree_Champagne. Variables dependent on time are inside dotted box; Variables independent of time are outside botted box. Variables in circles are unknown and learned from the model, and Variables in squares are known. N is the number of voxels, $\boldsymbol{s}^i$ denotes the ith voxel time course, $\boldsymbol{s}^{i,j}$ is the jth voxel's time course in ith region.

Initialization for $\boldsymbol{\Upsilon}$ is set by running Bayesian Minimum-Norm (BMN) [2], described below, to determine a whole-brain level variance parameter and the variance of all voxels and regions are initialized to this value. The precision and the mean of the posterior distribution $p(\boldsymbol{x}|\boldsymbol{y})$ are computed using Eq. (11) and Eq. (16). The hyperparameter $\boldsymbol{\Upsilon}$ is updated using Eq. (24) and with the values of $\boldsymbol{\Lambda}$ updated using Eq. (25) and $\overline{\boldsymbol{x}}$ obtained earlier. Finally, we calculate the variance of both voxels and regions, the time course of each voxel using Eq. (28) and Eq. (29) with both voxels and regions taken into consideration.

2.6. Relationship of tree_Champagne to BMN and to Champagne

BMN [2] and Champagne are two other Bayesian algorithms for source reconstruction which have close relation to tree_Champagne. The difference among the three algorithms are in the generative model, as can be seen in Figure. 1. For BMN, voxels in source space have a scalar variance υ. Bayesian estimation of this model yields the BMN algorithm which results in smooth widespread activity throughout the brain. In contrast to BMN, each voxel in Champagne has a different prior variance. Bayesian inference of the Champagne model yields very sparse reconstructions [13]. In contrast to both of these algorithms, tree_Champagne uses a source space that is segmented into different regions according to prior anatomy and function. We then assume that each region has its own region's level variance. Additionally, tree_Champagne also includes an intrinsic variance for each voxel independent of the regional variance. This variance partitioning enables it to produce source reconstructions with varying spatial extents, as we will show below.

3. Performance evaluation on simulation and real data

This section describes the performance evaluation of tree_Champagne under different specific complex configurations compared with other four representative benchmark source reconstruction algorithms. Then we evaluate the performance of tree_Champagne using real MEG and EEG datasets with different tasks.

3.1. Benchmark source localization algorithms

Four representative source localization algorithms we chose to compare with the performance of tree_Champagne are: (1) an adaptive spatial filtering method, linearly constrained minimum variance beamformer (referred to as Beamformer) [14, 15, 16, 17], (2) a non-adaptive weighted minimum-norm method, standardized low-resolution tomographic analysis (referred to as sLORETA) [7, 8], and two Bayesian based algorithms - (3) Champagne [13] and (4) MSP [30]. In simulations, for sLORETA we fix the regularization to be 1e-6 times the maximum eigenvalue of the composite lead-field. We did not find much variation in performance when we changed the regularization by 1-2 orders of magnitude. For real MEG data tests, we use BMN [2] for learning the regularization parameter which is then used in conjunction with sLORETA - we call this algorithm of using sLORETA for normalization after BMN as BMN_sLORETA. We found slight improvements in real data using BMN_sLORETA when compared to sLORETA with a fixed regularization that we used in our simulations (see Figure. 10). For Champagne, Beamformer and sLORETA, we use matlab files (nuts_Champagne.m, nuts_LCMV_Vector_Beamformer.m and nuts_sLORETA.m) from NUTMEG [31]. For MSP, we use the exact implementation of MSP as included in standard settings in SPM12 (spm_eeg_invert.m).

3.2. Quantifying performance

In order to evaluate the performance on simulated results, two features are quantified: localization accuracy and time course estimation accuracy. We first examine whether sources are correctly localized, then measure if the source time courses are accurately reconstructed for those source locations. The occurrence of both hits rate and false positives are taken into account for the evaluation. The free-response ROC (FROC) curve is used as it allows for multiple hits and false positives in a single image [32]. The $A^{'}$ metric [33] estimates the area under the FROC curve for one hit rate (H_R) and false positive rate (F_R) pair, or in our case, for each simulation. If the area under the FROC curve is large, then the hit rate is higher compared to the false positive rate.

$$A^{'} = \frac{H_R - F_R}{2} + \frac{1}{2} \tag{30}$$

where H_R is calculated by dividing the number of hits by the true number of seeded sources and F_R is calculated by dividing the number of false positive by the maximum number of false positives for each algorithms. Eq. (30) is

a simple way to compute $A^{'}$ metric in our prior paper [34]. The correlation coefficient between the seed and estimated source time courses for each hit is used to determine the accuracy of the time courses. We then average the correlation coefficients for all the hits for each simulation run, which is denoted as $\bar{R}$. Finally, we combine these two metrics that capture both the accuracy of the location and time courses of the algorithms into a single metric called the Aggregate Performance (AP) [18]. It combines the $A^{'}$, $\bar{R}$, and H_R using the following equation:

$$AP = \frac{1}{2}(A^{'} + H_R\bar{R}) \tag{31}$$

The H_R is used as a weight for $\bar{R}$ since we only compute the correlation coefficient for the sources that are correctly localized. AP ranges from 0 to 1, with higher numbers reflecting better performance.

3.3. MEG simulations

In this paper, we generate data by simulating dipole sources with fixed orientation. Damped sinusoidal time courses are created as voxel source time activity and we then project the voxel activity to the sensors using the forward model generated lead field matrix. The lead field is constructed within the brain volume assuming a single-shell spherical model [19] as implemented in SPM12 (http://www.fil.ion.ucl.ac.uk/spm) at the default spatial resolution of 8196 voxels at approximately 5 mm spacing. The time course is then partitioned into pre- and post-stimulus periods. In the pre-stimulus period (180 samples) there is only noise plus interfering brainc activity, while in the post-stimulus period (300 samples) there is also source activities of interest on top of statistically similarity distributed noise plus interfering brain activity. The noise plus interfering activity consists of actual resting-state sensor recordings collected from a human subject presumed to have only spontaneous brain activity and sensor noise. The voxel level activity is then projected to the sensors through the lead field and the noise/interference is then added to achieve a desired signal to noise ratio. The simulated data has 271 sensor recordings. The locations for the active sources are chosen so that there is some minimum distance between sources (at least 15 mm) and a minimum distance from the center of the head (at least 35 mm) [18].

We could adjust both the signal-to-noise-plus-interference ratio (SNIR) and the correlations between the different voxel time courses (inter-dipole α_{inter}) to examine the algorithm performance on unknown correlated sources and fixed orientation. In this paper, SNIR and correlation between sources are defined the same way as is shown in our prior work [18].

Similar to our prior work, we picked difficult configurations that we have tested for Champagne [18]. Additionally, in this paper, we extend our tests to sources with extended spatial extent, i.e. source clusters and regions with more complex configurations [35]. A voxel source is the point dipolar source and a cluster source is defined as sources with several closely located dipolar sources. A

region is set a priori using atlases by dividing the whole brain into regions defined either anatomically or functionally, such as the Automated Anatomical Labeling (AAL) [24]. We expand configurations with correlations between clusters (inter-clusters, β_{inter}) which define the voxel time courses correlation from different cluster and correlations in clusters (intra-cluster, β_{intra}) which define the voxel time courses correlation among the same cluster. We also tested the effect of activity with both clusters and point sources. In summary, the configurations we tested are as follows:

1 Correlation within cluster - We examine the ability to reconstruct clusters with increasing correlation of sources from the same cluster. We seed 5 clusters with 20 sources for each cluster. The correlation of sources from the same cluster is set as 0.1, 0.3, 0.5, 0.7 and 0.9 - in this situation we set the correlation between clusters as 0.25.

2 Correlation between clusters - We examine the influence of correlation between clusters for the novel algorithm. We seeded 5 clusters with 20 sources per cluster. We set the correlation between clusters as 0.1, 0.3, 0.5, 0.7 and 0.9 - the source time courses within each cluster is set to have an intra-cluster correlation coefficient of 0.5.

3 Number of clusters - We test the ability to localize distributed clusters by simulating different numbers of clusters. We seed 1, 4, 7, 10, 13, 16 clusters with 20 sources for each cluster. These clusters correspond to 20, 80, 140, 200, 260 and 320 voxels having nonzero activity. The placement of the cluster center is seeded randomly and cluster consists of sources seeded within the 19 nearest neighboring voxels.

4 Effect of clusters' size - We assess the robustness to localize distributed sources with different cluster sizes. We seed 5 clusters with 10, 16, 22, 28, 34 and 40 active dipoles per cluster, which correspond to 50, 80, 110, 140, 170 and 200 active voxels.

5 Number of regions - Since our novel algorithm is based on the distribution of voxels into regions, we also test the influence of different sizes of the region divisions. Here, we set the number of regions as 8, 9, 32, 95, 108, 116, 285 and 291 to evaluate performance of the tree_Champagne algorithm. For these simulations, we fix the activity as arising from 5 clusters with 20 sources for each cluster.

6 Mixed conditions (clusters and sources) - We extend the previous cluster analysis experiments to investigate the effect of having both cluster and dipole activity. We choose to set the number of clusters from 1, 4, 7, 11, 14 to 17 with additional activity from 5 dipoles. Subsequently, we set the number of clusters as 5 and vary the number of dipoles from 1, 4, 7, 10, 13, 16 to 19.

If not indicated otherwise, each of the experiments is conducted with the following settings: the source time courses within each cluster have an intra-cluster correlation coefficient of $\beta_{intra} = 0.5$ and an inter-dipole correlation coefficient of $\beta_{inter} = 0.25$. We make the correlations within the clusters higher than between clusters because nearby voxels are more plausibly correlated than voxels at a distance. For clusters, we are both interested in whether a cluster is localized and whether the extent of cluster is accurately reconstructed. To assess the localization of clusters, we use the $A^{'}$ metric. The $A^{'}$ metric is calculated for clusters by testing if there is a local peak within the known extent of the cluster. To assess the accuracy of the extent of clusters, we calculate the fraction of seeded voxels with power in or above 10th percentile of all voxels. At the same time, the power of localized peaks should be at least 0.1 percentile of the maximum power.

The results obtained using simulated data are averaged over 50 simulations for each of six configurations with SNIR=0 or 10 dB and we plot these averaged results with standard error bars. We show the plots of mean AP, our Aggregate Performance metric. We also show examples of localization results from single simulations, which complement our aggregate results.

3.4. EEG simulations

We also test the novel algorithm on simulated EEG data using a scalar lead-field computed for a three-shell spherical model in SPM12 (http://www.fil.ion.ucl.ac.uk/spm) at the default resolution resulting in 8196 voxels at approximately 5 mm spacing. The simulated EEG data has 120 sensor recordings. With this lead-field, EEG data is simulated in the same way as the MEG data, as described above. We repeat the detection of multiple clusters and mixed conditions (clusters and sources) experiments for EEG simulations.

3.5. Real datasets

All the MEG data here was acquired in the Biomagnetic Imaging Laboratory at University of California, San Francisco (UCSF) with a CTF Omega 2000 whole-head MEG system from VSM MedTech (Coquitlam, BC, Canada) with 1200 Hz sampling rate. The lead field for each subject is calculated in NUTMEG [31] using a single-sphere head model (two spherical orientation lead fields) and an 8 mm voxel grid. Each column is normalized to have a norm of unity. The data is digitally filtered from 1 to 160 Hz to remove artifacts and the DC offset is removed.

We ran tree_Champagne and all of the benchmark algorithms on five real MEG data sets: 1. Somatosensory Evoked Fields (SEF); 2. Auditory Evoked Fields (AEF); 3. Audio-Visual Evoked fields; 4. Face-processing task; 5. Interictal spike data from patients with epilepsy spikes. The first four data sets have been reported in our prior publications using the Champagne algorithm, and details about these datasets can be found in [18, 13]. Novel data included in this paper are interictal spikes from seven patients with epilepsy. These spikes were identified by trained MEG technologists in the Biomagnetic Imaging Laboratory, and the peak time-point was localized using dipole fitting method. For

Champagne and tree_Champagne, we choose a pre-spike window from -350 ms to -250 ms as a baseline control period and the post-spikes window is from -50 ms to 50 ms where the spikes time is at 0 ms time point.

The EEG data (128-channel ActiveTwo system) was downloaded from the SPM website (http://www.fil.ion.ucl.ac.uk/spm/data/mmfaces) and the lead field was calculated in SPM8 using a three-shell spherical model at the coarse resolution. The EEG data paradigm involves randomized presentation of at least 86 faces and 86 scrambled faces, here we subtract the averaged scrambled-faces data to the averaged faces data to study the differential response to faces versus scrambled faces [36], and the power is plotted on a 3-D brain. The EEG data has been reported in our prior publication using the Champagne algorithm, and details about our analyses of this dataset can be found in [18].

4. Results

4.1. MEG simulations

Figure 2 shows a representative example of localization results for an MEG simulation with 3 clusters at SNIR = 10 dB, compared with the ground truth. Champagne can find all three clusters but it estimates activity that is more focal than the true spatial extent of the sources. Tree_Champagne is also able to localize three clusters with estimates that are more spatially distributed than Champagne. Beamformer is unable to find the three clusters correctly. sLORETA can find all three clusters correctly but produces blurred and diffuse solutions. In contrast, MSP can find all three clusters but reconstructions are smoother than ground truth and also estimates additional sources that are not present in the simulations.

The performance for a second special case where 3 regions of the model are specified to be active is shown in Figure 3. Only tree_Champagne is able to reconstruct the correct active region, showing the extended activity corresponding to each region. Champagne localizes the active region but treats the regions' activity as if they are arising from several point sources. In contrast, sLORETA, Beamformer and MSP do not accurately estimate the regions' active and show blurred and inaccurate reconstructions.

Figure 4 shows an example of the steps that go into the aggregate performance metric calculation. With the increase of correlation in clusters, we first calculate Hit Rate (subplot A) and False Rate (subplot B) using the method from our prior work [18]. Then, the correlation between hit sources and seeded time series is obtained as shown in subplot C. At last, we calculate the $A^{'}$ matric and Aggregate performance using Eqs. (30) and (31). Aggregate performance across 50 simulations for each of 6 configurations is reported. For subsequent performance evaluation figures we only show the AP metric.

4.1.1. Influence of the Correlation within each Cluster

The sensitivity to performance as a result of increasing the correlation within each cluster on both 10 dB and 0 dB is presented at the first row of Figure 5.

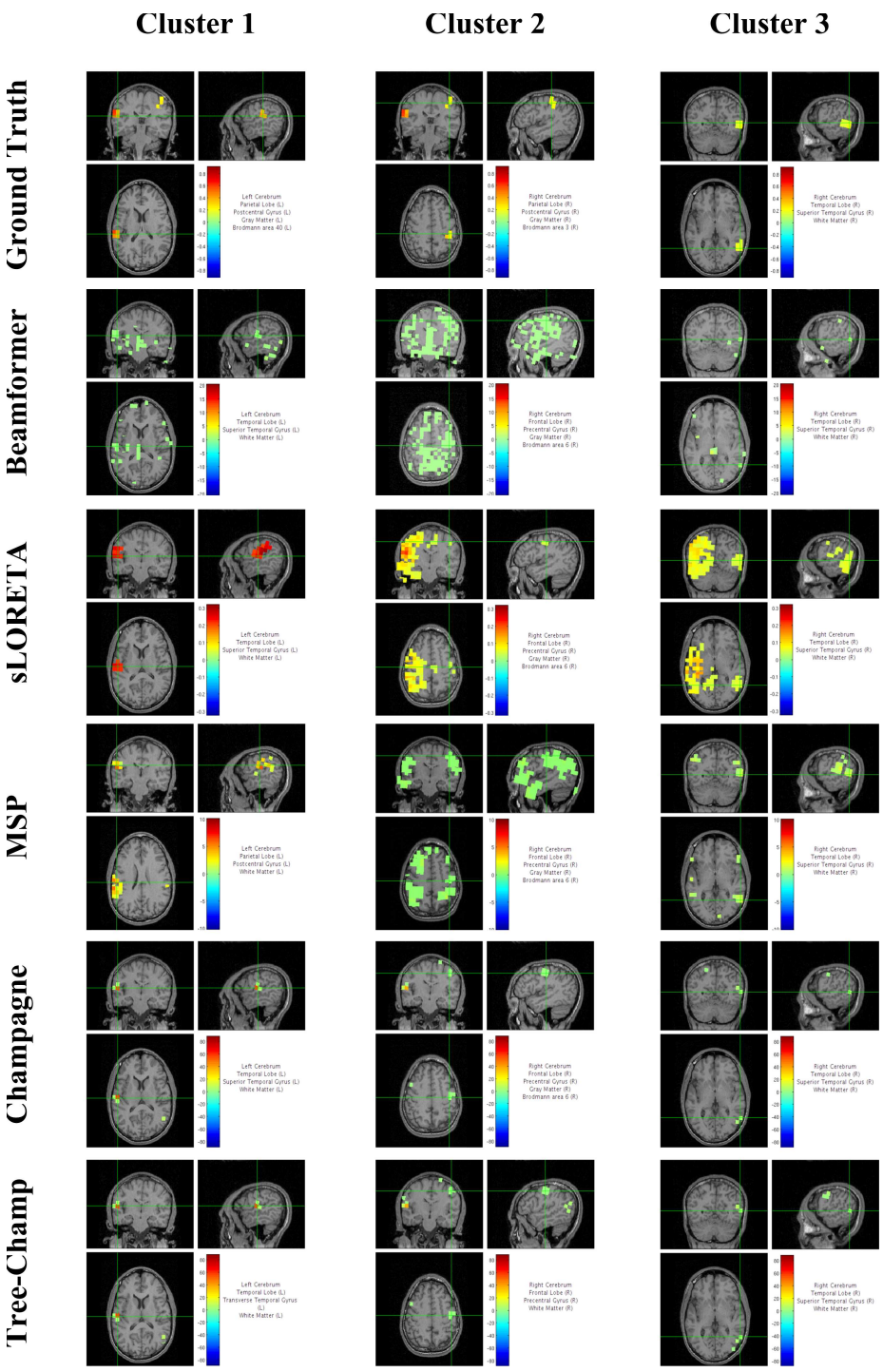


Figure 2: Example of the localization results for simulated MEG data with 3 clusters at SNIR=10. The activity power is normalized by the lead-field value at each voxel. The ground truth is shown for comparison.

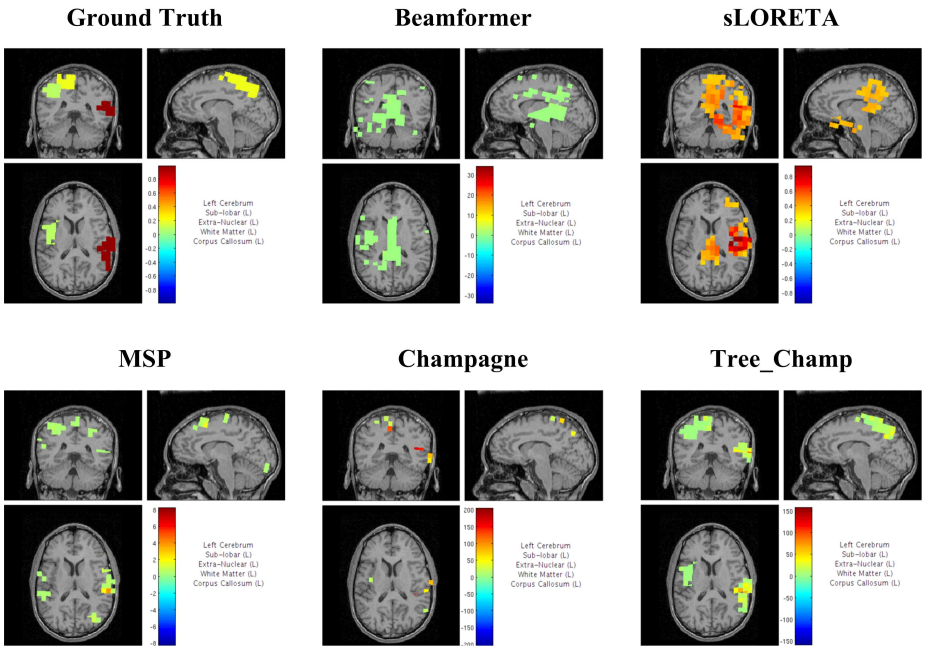


Figure 3: Example of the localization results for simulated MEG data with 3 regions active at 10 dB. The activity power is normalized by the lead-field value at each voxel. The ground truth is shown for comparison.

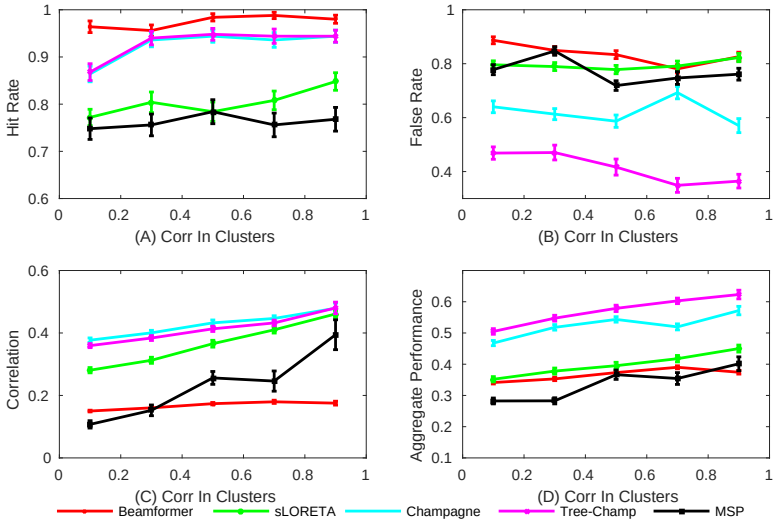


Figure 4: Example of the Aggregate Performance metric calculation with increasing correlation in clusters from 0.1 to 1 at 10 dB for 50 simulations: (A) Averaged Hit Rate for all algorithms; (B) Averaged False Rate for all algorithms; (C) Averaged correlations for all hit sources; (D) Averaged Aggregate Performance scores for all algorithms.

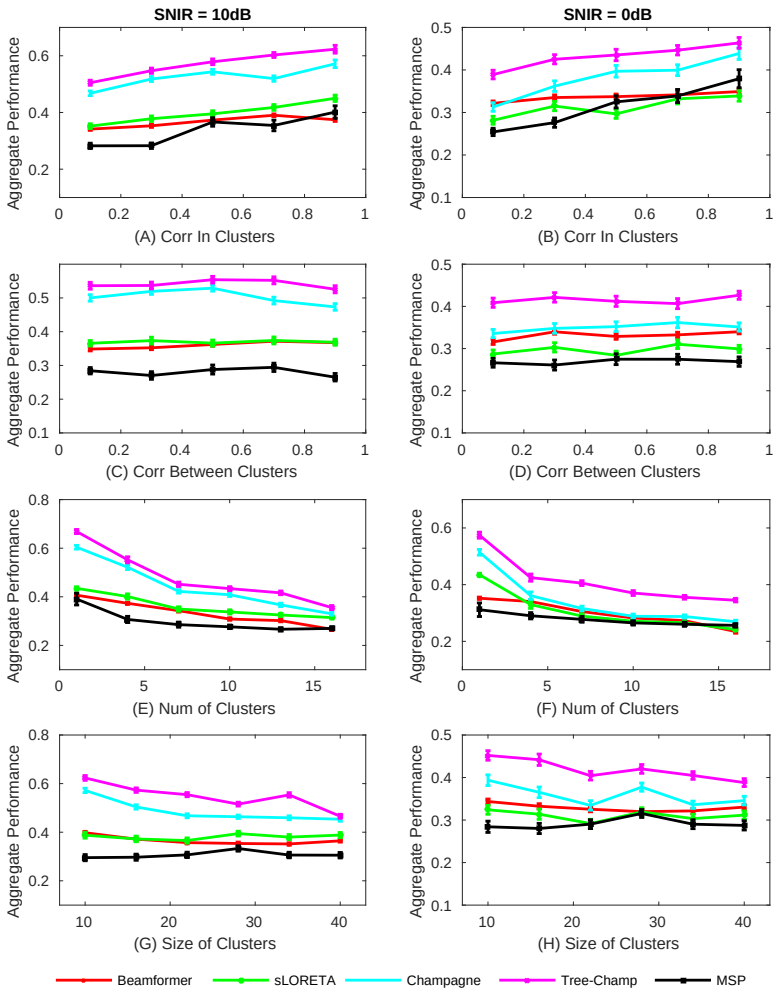


Figure 5: Simulation results of Aggregate Performance with four different configurations at 10 dB and 0 dB: (A) and (B) show results for increasing dipoles time courses correlation from the same cluster; (C) and (D) show results for increasing correlation between clusters; (E) and (F) show results for increasing the number of clusters; (G) and (H) show results for variations in the sizes of the clusters.

From the AP plot, for both SNIR = 10 dB and SNIR = 0 dB, tree_Champagne outperforms all benchmarks. Champagne is not as good as tree_Champagne but is much better than other benchmarks. The benchmark algorithms perform somewhat similarly for these simulations, but their performance is not as good as tree_Chamagne and Champagne. Nevertheless, increasing the correlation in each cluster also improves the performance of all algorithms.

4.1.2. Influence of the Correlation between Clusters

The second row of Figure 5 shows the influence of increasing the correlation between clusters on algorithm performances. Increasing the correlation between clusters has little influence on the performance of all algorithms at both 10 dB and 0 dB. Based on the AP metric, it is clear that tree_Champagne outperforms all benchmarks. Although Champagne is not as good as tree_Champagne, but it is the best among all benchmarks when compared to Beamformer, sLORETA and MSP.

4.1.3. Influence of the number of clusters

In the third row of Figure 5, we plot the number of clusters versus AP metric at SNIR levels of 10 dB and 0 dB. All algorithms have the same trend at both 10 dB and 0 dB, with the increase number of clusters, the AP score decreases. Again, tree_Champagne outperforms all benchmark algorithms. Champagne is not as good as tree_Champagne but better than others. For benchmarks, at 10 dB, sLORETA shows higher AP score than Beamformer and MSP. While at 0 dB, all benchmarks performs at a similar level when the number of clusters is more than 4.

4.1.4. Effect of Clusters' size

The results of all methods at both 10 dB and 0 dB in response to increasing clusters' size are presented in the last row of Figure 5. Performances of all algorithms do not show much change when the clusters' size increases. From the AP plot, tree_Champagne outperforms all benchmarks. Again, Champagne is very close to tree_Champagne with superior performance when compared to Beamformer, sLORETA and MSP.

4.1.5. Effects of Increasing the Number of Regions in the Generative Model

The first row of Figure 6 shows the influence to localization methods by increasing the number of regions. The whole source space is segmented into different size regions and tested at 10 dB and 0 dB. Although this should only influence the performance of tree_Champagne algorithm, we also show performance for the benchmarks for these specific simulation data instantiations using the same performance metrics. The intra-cluster correlation is at 0.5 and the inter-clusters correlation is 0.25. The results are averaged over 50 simulations each with 5 clusters seeded with 20 sources for each cluster, and the error bars show the standard error. As we can see in the AP metric, when increasing the number of the regions, despite some changes in the performance of tree_Champagne, it is superior to the benchmark algorithms.

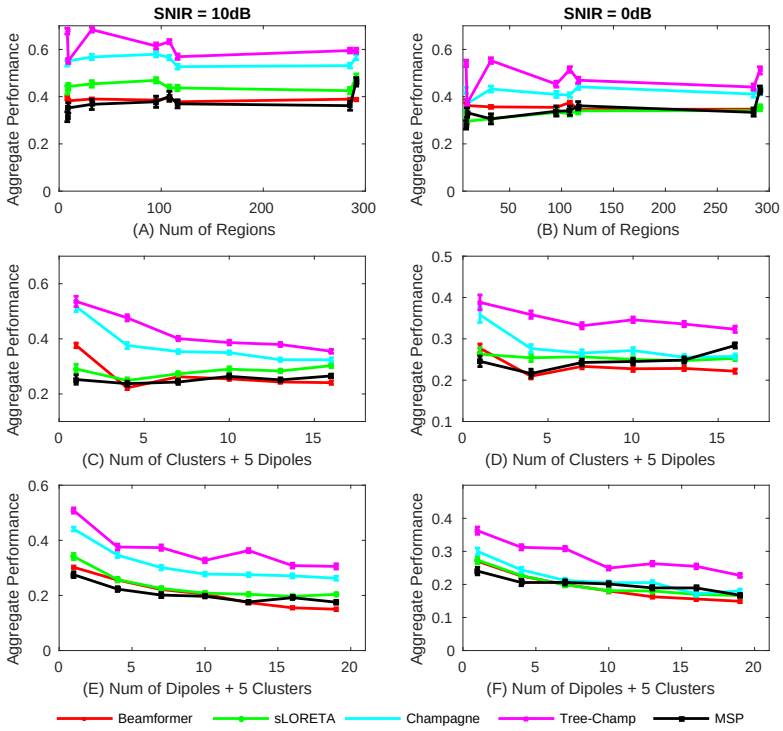


Figure 6: Aggregate Performance with three different configurations: (A) and (B) show results for increasing the brain's regions at 10 dB and 0 dB; (C) and (D) show the performance of all algorithms with fixed 5 dipoles while increasing the number of clusters at 10 dB and 0 dB; (E) and (F) show results with fixed 5 clusters but increasing the number of dipoles at 10 dB and 0 dB.

4.1.6. Performance for Mixed Source Configurations

A single representative simulation experiment with 2 clusters and 2 dipoles at 10 dB is presented in Figure 7, where the ground truth is shown on the first row for comparison. Champagne and tree_Champagne can localize all clusters and the dipoles. Beamformer, sLORETA and MSP can localize almost all activities but with very diffuse reconstructions and some false positive activity estimates.

In order to evaluate the performance of source localization algorithms for configuration with both clusters and dipoles, we first fix the number of dipoles as 5 and increase the number of clusters, then we fix the number of clusters as 5 and increase the number of dipoles. The final results are plotted in the second and third rows of Figure 6. As we can see in the AP value with fixed number of dipoles and increase number of clusters, the performance of all algorithms decreases. Tree_Champagne outperforms the benchmarks at both 10 dB and 0 dB. Champagne shows better performance than other benchmarks at 10 dB but is close to others at 0 dB. We then fix the number of clusters and increase the number of dipoles, and the performance of all algorithms decline as the number of dipoles increases. Tree_Champagne still produces the highest scores among all source localization algorithms both at 10 dB and 0 dB.

4.2. EEG simulations

In Figure 8 we show EEG simulation results at 10 dB. According to our tests, the performance of all algorithms have a similar trend with SNIR equals to 10 dB or 0 dB. The left column shows the results of A Prime Metric and the right column is the Aggregate Performance score. Across both the A Prime and Aggregate Performance metrics, tree_Champagne outperforms all benchmarks for all three different configurations.

In simulations, according to the evaluation function used in the paper, the performance of tree_Champagne is much better than Champagne, especially for clusters localization. Tree_Champagne is also more accurate than Champagne at estimating the spatial extent of cluster sources. As is shown in Figure 9, when we compare the radius of estimated size of clusters for Champagne and tree_Champagne, the latter is better at estimating the spatial extent of the cluster.

4.3. Summary for simulations

As we can see from the simulation results and analysis above, both at 10 dB or 0 dB, tree_Champagne outperforms all the benchmark source reconstruction algorithms. Next, we extend the evaluation of the performance using real MEG and EEG data.

4.4. Results of real data

This section shows the evaluation for our algorithms using real MEG and EEG data, which contains five different MEG datasets and one EEG dataset: Somatosensory Evoked Field Paradigm, Auditory Evoked Field, Audio-Visual task, Face-processing task for MEG, Epileptic spikes data for MEG and Face-Processing task for EEG.

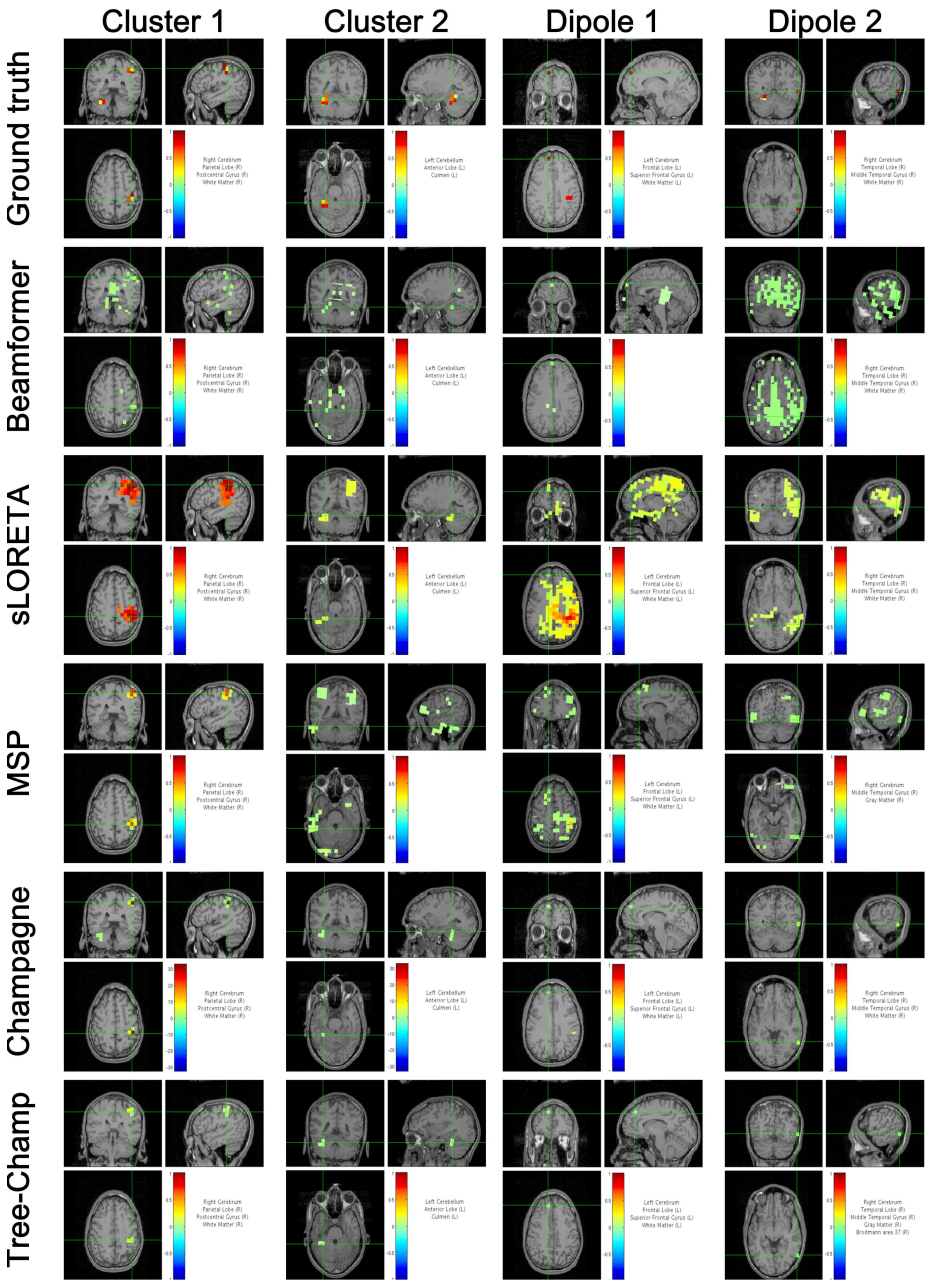


Figure 7: Example of the localization results for 2 clusters and 2 dipoles at SNIR = 10 dB. The activity power is normalized by the lead-field value at each voxel. The ground truth is shown for comparison.

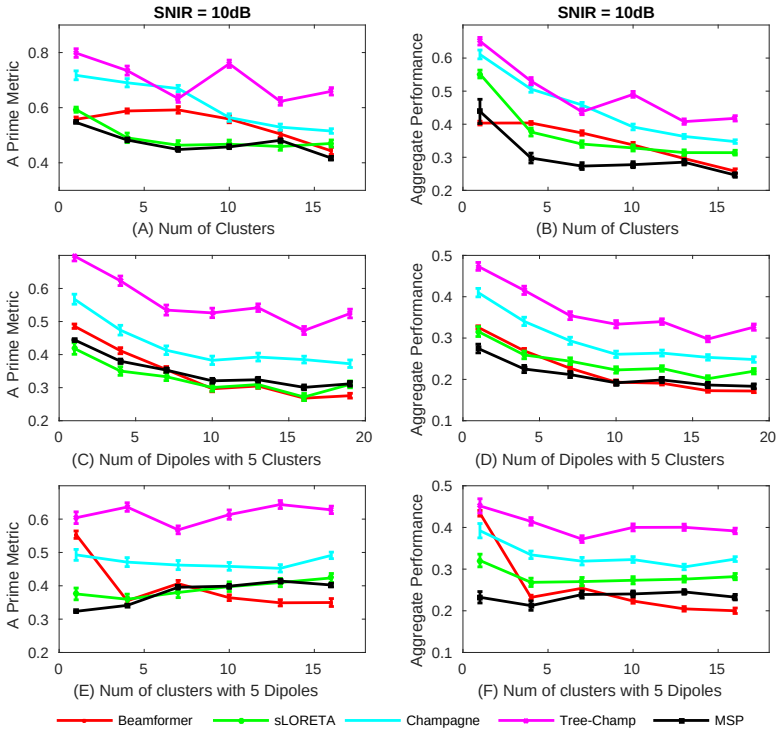


Figure 8: EEG simulation results of the A Prime Metric (left column) and Aggregate Performance (right column) with three different configurations at 10 dB: (A) and (B) show results for increasing number of clusters; (C) and (D) show results with fixed 5 clusters and increasing the number of dipoles; (E) and (F) show results with fixed 5 dipoles while increasing the number of clusters.

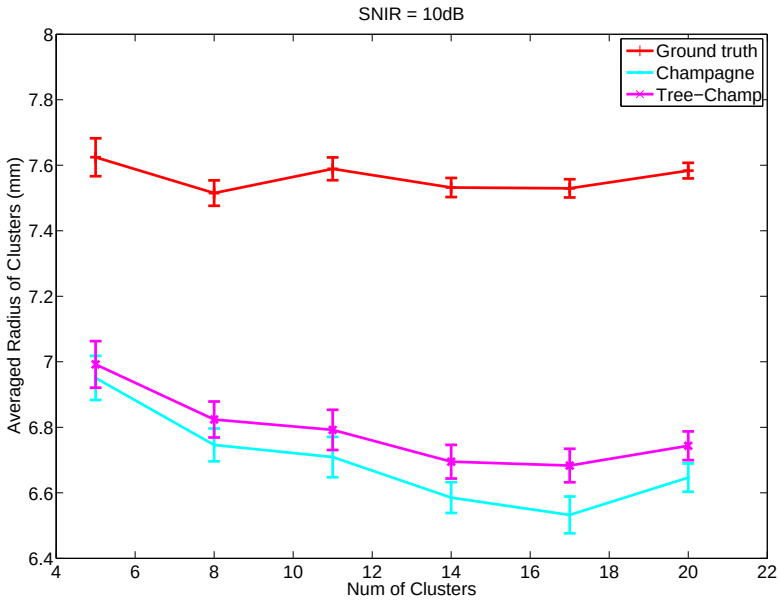


Figure 9: Averaged Radius of Clusters with EEG simulations for Champagne, tree_Champagne. The Ground Truth is shown for comparison.

4.4.1. Somatosensory Evoked Field Paradigm

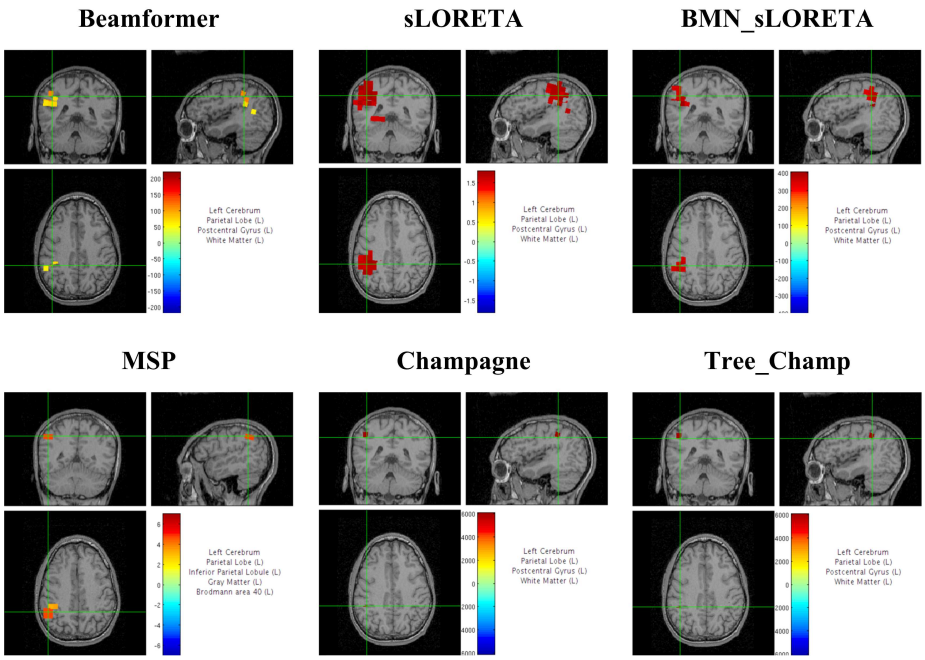


Figure 10: Sensory Evoked Field localization results. The activity power is normalized by the lead-field value at each voxel. All six algorithms localize to somatosensory cortical areas, where Champagne and tree_Champagne are the most focal. BMN_sLORETA also performs well on the localization. Here we set the threshold for tree_Champagne and Champagne much lower than other benchmarks.

Figure 10 shows the results of the somatosensory evoked field response due to somatosensory stimuli presented to a subject's right index finger, average derived from a total of 240 trials. A peak is typically seen ∼50 ms after stimulation in the contralateral (in this case, the left) somatosensory cortical area for the hand, i.e., dorsal region of the postcentral gyrus. MSP, Champagne and tree_Champagne can localize this activation to the correct area of somatosensory cortex with focal reconstructions. Here, we show performance in three benchmarks - Beamformer, sLORETA with a fixed regularization, and BMN_sLORETA. While benchmarks are also able to localize somatosensory cortex, these reconstructions are more diffuse especially for sLORETA with a fixed regularization.

4.4.2. Auditory Evoked Fields

The localization results for AEF data from three subjects are shown in figure 11. The power of at each voxel in a 50-75 ms window around M100 peak is plotted for every algorithm. Both Champagne and tree_Champagne are able

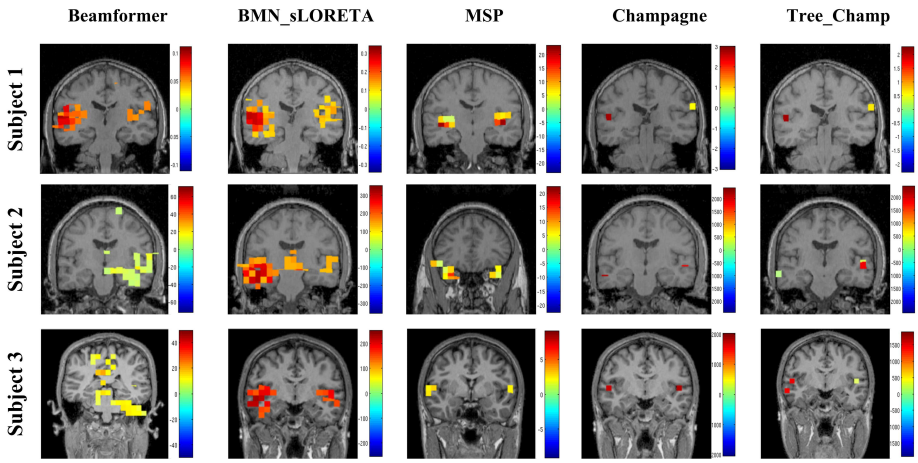


Figure 11: Auditory Evoked Field results for three subjects. The activity power is normalized by the lead-field value at each voxel. The results from both Champagne and tree_Champagne are shown in the last two columns, which outperform the other benchmark algorithms shown in the first to three columns.

to consistently localize bilateral auditory activity for all subjects (shown in the last two columns in Figure 11). The activity is in Heschl's gyrus, which is the location of primary auditory cortex. Champagne and tree_Champagne perform similarly for all subjects. Beamformer can find the two auditory cortices only in one subject, whereas for the rest of the subjects the activations are mostly biased towards the centra of the head; This suggests that the correlation of bilateral auditory cortical activity really impacts the performance of Beamformer. BMN_sLORETA is able to find the auditory activity for almost every subject, but the results are diffuse and with additional spurious activities (not seen on the slices shown). MSP can localize bilateral auditory activity but with some location bias and more diffuse activation.

4.4.3. Audio-Visual Evoked Fields

Figure 12 shows results of the audio-visual evoked fields for tree_Champagne. In subplot (A) and (B) we show the brain activations associated with the auditory stimulus. Tree_Champagne is able to localize bilateral auditory activity in Heschl's gyrus in the window around the M100 peak, shown in the first row of Figure 12. The two auditory sources have the maximum power in the window around the M100 peak. We show the early visual response in the second row of Figure 12. Tree_Champagne is able to localize a source in the medial, occipital gyrus with a peak around 150 ms. We plot the power in the window around this peak and the time course of the source marked with the cross hairs. Our novel algorithm can localize a later visual response with a time course that has power extending past 150 ms, which is similar to the results that we have obtained with Champagne [18].

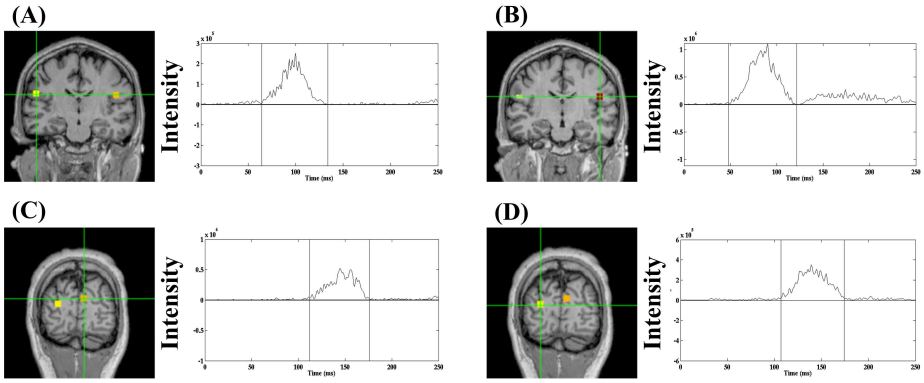


Figure 12: Audio-Visual data localization results from tree_Champagne. The activity power is normalized by the lead-field value at each voxel. Tree_Champagne is able to localize a bilateral auditory response at 100 ms after the simultaneous presentation of tones and a visual stimulus. For bilateral auditory activity, the results of locations and time courses are shown in (A), (B). Tree_Champagne can localize an early visual response at 150 ms after the simultaneous presentation of tones and visual stimulus shown in (C) and (D).

4.4.4. Face-processing task: MEG

Localization of Face-processing task (MEG) in response to faces are shown in Figure 13. We see an early visual cortical response to the presentation of the face visual stimulus in medial occipital cortex and later visual cortical response more lateral to the early response shown in the first row of Figure 13. Subsequently, tree_Champagne is able to localize the bilateral activation in the fusiform gyrus with peaks around 170 ms [36, 37]. Performance of benchmarks algorithms on this dataset can be found in [18].

4.4.5. Face-Processing task: EEG

In Figure. 14, we present the results from using novel algorithm and benchmarks on the face-processing task EEG data set. Figure 14 shows the average power, M100 peak power and M170 peak power at different rows separately. We see that tree_Champagne is able to localize the brain activity with sparse peaks at visual areas and fusiform gyrus. However the benchmarks produce the brain activity with either wrong location or blurred solutions. Even though the threshold we use is 1% of the maximum activation of the image for tree_Chamapgne and 10% of the maximum activation of the image for benchmarks, our novel algorithm gives us more sparse and accurate results.

4.4.6. Epilepsy Spikes

The localization results for epilepsy spikes data from seven patients are shown in Figure 15. The best time point dipole fitting for each spike is shown in the left-most column for reference. As we can see, both Champagne and

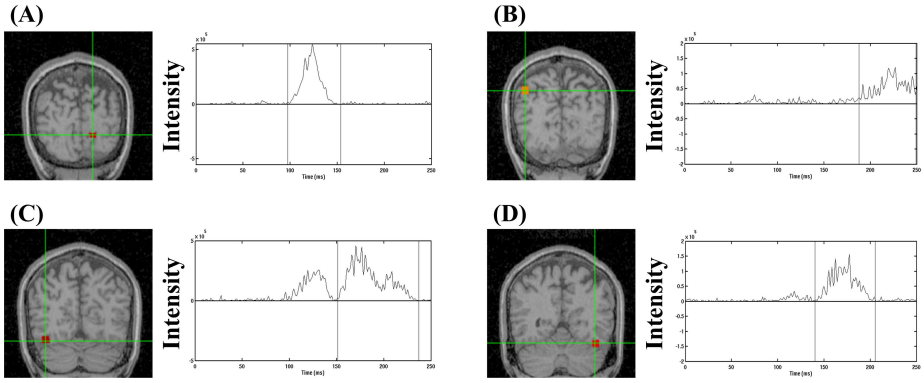


Figure 13: Face-processing task (MEG) localization results for tree_Champagne. The activity power is normalized by the lead-field value at each voxel. Tree_Champagne can localize an early visual response around 100 ms after the presentation of a face stimulus, results with time courses shown in subplot (A). A later visual response around 200 ms after the presentation of a face stimulus are shown in subplot (B). The novel algorithm can localize the bilateral activation in fusiform gyrus that is thought to be in FFA, shown in (C) and (D). The peak for the brain activity is around 170 ms after the presentation of a face stimulus, and the time courses are shown next to brain activity figures in subplots (C) and (D).

tree_Champagne are able to localize almost all spikes for all subjects (shown in the forth to fifth columns in Figure 15). Champagne and tree_Champagne perform similarly for all subjects. For other benchmark algorithms, Beamformer can localize the spike for each subject, but localization results are only reasonable for subject 5 since the rest are either diffuse or have many spurious activations which are stronger than the true location of the spikes. BMN_sLORETA performs better than Beamformer, but shows more diffuse results when compared to Champagne and tree_Champagne. Since default MSP settings were optimized for scalar lead-fields but these data included vector lead-fields, we did not run MSP on these data.

5. Discussion

This paper derives a novel hierarchical multiple spatial scale Bayesian algorithm, tree_Champagne, for electromagnetic brain imaging using magnetoencephalography (MEG) and electroencephalography (EEG) with comparisons to existing benchmark algorithms. The novel algorithm is based on a principled cost function that maximizes the marginal likelihood of the data with fast, convergent update rules. The multiscale formulation enables tree_Champagne to optimally combine smoothness (from regional-level inference) and sparsity (from voxel level inference). Results show significant theoretical and empirical

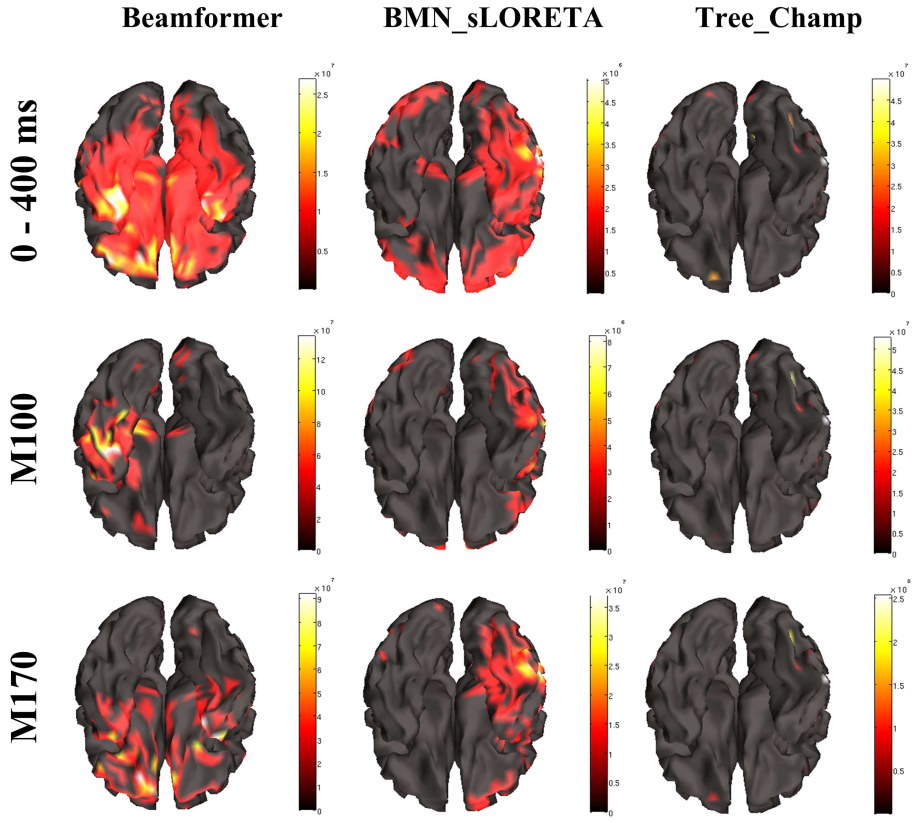


Figure 14: Results for face processing (EEG) from novel algorithm and benchmarks. The first row is the average power mapping from 0 ms to 400 ms, the second and third rows are for peak power activity at 100 ms and 170 ms separately. Thresholds is 1% of the maximum activation of the image for tree_Chamapgne and 10% of the maximum activation of the image for benchmarks. The activity power is normalized by the lead-field value at each voxel.

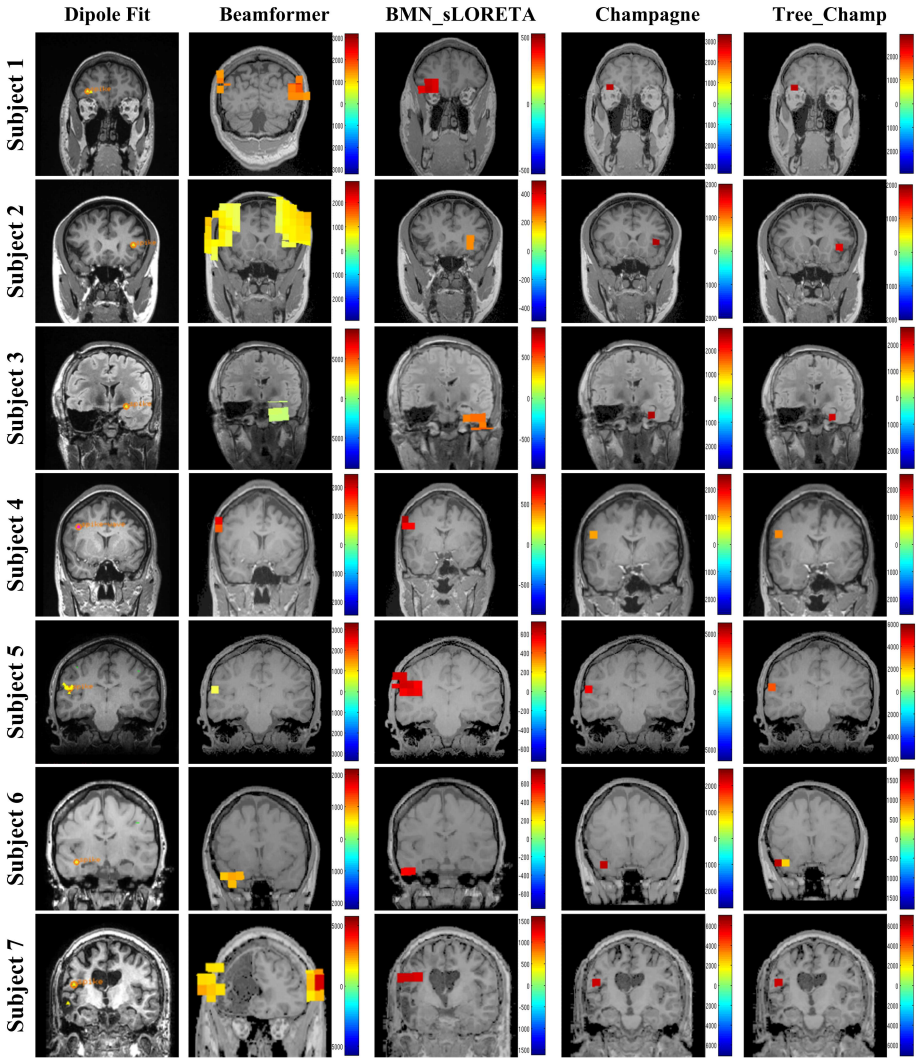


Figure 15: Epilepsy Spikes results for 7 subjects. The results of best time point dipole fitting are shown in the left-most column, the results of benchmarks are shown from second to forth columns, the novel algorithm's results are shown in the last column. The activity power is normalized by the lead-field value at each voxel.

advantages over many existing methods. The algorithm readily handles multiple correlated sources and is appropriate for sources that have variable spatial extent ranging from isolated dipoles and extended clusters of dipoles, situations that commonly arise even with simple cognitive neuroscience tasks.

The experiments with simulated data exemplify that tree_Champagne provides robust localization and time course estimation with complex source configurations and noisy data for both MEG and EEG simulations with correlated sensor data. Tree_Champagne outperforms existing benchmarks with highly correlated sources even at high levels of interference at 0 dB. We also found that with increasing of the number of clusters and the size of clusters, tree_Champagne performs much better than the benchmark algorithms. Notably, tree_Champagne performance better than Champagne since tree_Champagne shows more extended activity for clusters. For more complex configurations with simultaneous clusters and dipoles activity for both MEG and EEG, tree_Champagne is also able to accurately localize the simulated activity and significantly outperforms benchmark algorithms.

Experiments with real data highlight the source localization abilities of the novel algorithm. It is difficult to evaluate localization accuracy with real data since the ground truth is not known. For this reason, we have chosen real data sets that have well-established patterns of brain activity; AEF, audio-visual, and face-processing data. For all these real data, the tree_Champagne algorithm performs superiorly compared to benchmarks and improves upon our prior work on Champagne. Additionally, here we examine a novel dataset of interictal spikes from patients with intractable epilepsy. For these data, tree_Champagne is able to successfully localize all spikes for all subjects.

In this paper, the novel algorithm mainly models and addresses issues related to incorporating priors for spatial-smoothness of sources activity. We extend our prior framework of Champagne to include this spatial smoothness using regional variances. Other researchers have taken different approaches for incorporating priors on spatial-smoothness for sources reconstruction. Knösche [38] has proposed a functional similarity as priors for the reconstruction of distributed source current densities from EEG: patchLORETA1, which uses both topological neighborhood and prior information to define smoothness and patchLORETA2, which neglects topological neighborhood [39]. Alternatively, fMRI-Informed Regional Estimation (FIRE) [40] utilizes information from fMRI in EEG/MEG source reconstruction which takes advantage of the spatial alignment between the neural and vascular activities, while allowing for substantial differences in their dynamics.

The region-based variance model in tree_Champagne is different from other multiscale or hierarchical approaches in several ways [40, 41, 42, 43]. First, algorithms like the Multiple Sparse Priors (MSP) also evaluated here [20] impose spatial kernel smoothness across voxels based on the adjacency matrix and only include regional level variances with no voxel-level variances. Second, in contrast to these algorithms, we do not use variational approximations to factorize the posterior variances at the region-level and voxel-levels, which allows for the posterior voxel and regional variances to be correlated. Finally, we do not

use greedy algorithms like those proposed in Babadi et al. and Friston et al. [43, 41], which are highly sensitive to initialization and have the possibility of sub-optimal solutions. In contrast to using these update rules that are based on approximate likelihood maximization using restricted maximum likelihood based, cost functions that have slower convergence rates, tree_Champagne uses faster update rules based on convex-bounds on true marginal likelihood of the data[4]. Given the similarities between the proposed generative model, MSP and related hierarchical algorithms, inclusion of proposed implementation ideas into these frameworks may minimize observed differences in results.

Notably, however, the algorithms described in this paper do not incorporate temporal smoothness constraints and this represents the future directions for our work. Various forms of temporal prior information or constraints can be unified within the framework of covariance component estimation. We are currently investigating the use of temporal-smoothness priors in the form of basis functions [44] and in the form of autoregressive smoothness priors, which also model spatiotemporal correlations in the background noise and can potentially improve performance. The best example of such an effort is the Bayesian Electromagnetic Spatio-Temporal Imaging of Extended Sources (BESTIES) [45] algorithm, which is built upon a Bayesian framework that determines the spatio-temporal smoothness of source activities in a fully data-driven fashion is based on a Markov Random Field (MRF), which can precisely capture local cortical interactions, employed to characterize the spatial smoothness of source activities, and importantly the temporal dynamics of which are modeled by a set of temporal basis functions (TBFs). Jean Daunizeau et al.[42] also introduced a Bayesian framework to incorporate distinct temporal and spatial constraints on the solution and to estimate both parameters and hyperparameters of the model. A full multivariate autoregressive (MAR) model formulates directed interactions (i.e., effective connectivity) between sources. The observation process of MEG data, the source dynamics, and a series of the priors are combined into a Bayesian framework using a state-space representation. By formulating the source dynamics in the context of MEG source reconstruction, and unifying the estimations of source amplitudes and interactions, the effective connectivity without requiring the selection of regions of interest can be identified [39]. We derive inspiration for our future work from these approaches. Our next steps will focus on incorporating temporal-smoothness and effective connectivity prior on our novel algorithms, which hold promise for improving upon an already robust source localization algorithm.

Acknowledgment

The authors would like to thank Susanne Honma, Danielle Mizuiri and Anne Findlay for collecting much of the MEG data in the Biomagnetic Imaging Laboratory. We would also like to thank Julia Owen for the support of MEG data simulation and evaluation, Hagai Attias for inspiration and early discussions, and Inez Raharjo for editing. This work was supported by NIH grants

R01EB022717, R01DC013979, UCOP MRP-17-454755, and a gift from Ricoh Company Ltd..

Appendix A. Derivation of the marginal likelihood function

Here, we derive the expression for the marginal likelihood function shown in Eq. (17). We make the use of the form [2] (*pp.*244)

$$\begin{aligned}\log p(\boldsymbol{Y}|\boldsymbol{\Upsilon}) &= E_{p(\mathbf{X}|\mathbf{Y})}\left[\frac{\log p(\mathbf{Y},\mathbf{X}|\boldsymbol{\Upsilon})}{p(\mathbf{X}|\mathbf{Y})}\right] = \int d\mathbf{X} p(\mathbf{X}|\mathbf{Y}) \log[\frac{p(\mathbf{Y},\mathbf{X}|\boldsymbol{\Upsilon})}{p(\mathbf{X}|\mathbf{Y})}] \\ &= E_{p(\mathbf{X}|\mathbf{Y})}\left[\log p(\mathbf{Y}|\mathbf{X})\right] + E_{p(\mathbf{X}|\mathbf{Y})}\left[\log p(\mathbf{X}|\boldsymbol{\Upsilon})\right] + \mathcal{H}\left(p(\mathbf{X}|\mathbf{Y})\right)\end{aligned} \tag{A.1}$$

Substitution of equations Eq. (8), Eq. (9) and Eq. (10) into Eq. (A.1) results in the relationship

$$\begin{aligned}\log p(\boldsymbol{Y}|\boldsymbol{\Upsilon}) &= \log|\boldsymbol{\Sigma}_\varepsilon| - E_{p(\mathbf{X}|\mathbf{Y})}\left[\textstyle\sum_{k=1}^{K}(\boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{x}_k)^T \boldsymbol{\Sigma}_\varepsilon^{-1}(\boldsymbol{y}_k - \boldsymbol{H}\boldsymbol{x}_k)\right] \\ &\quad + \log|\boldsymbol{\Upsilon}| - E_{p(\mathbf{X}|\mathbf{Y})}\left[\textstyle\sum_{k=1}^{K} \boldsymbol{x}_k^T \boldsymbol{\Upsilon} \boldsymbol{x}_k\right] - \log|\boldsymbol{\Gamma}| \\ &= -\frac{1}{K}\sum_{k=1}^{K}\left[(\boldsymbol{y}_k - \boldsymbol{H}\overline{\boldsymbol{x}}_k)^T \boldsymbol{\Sigma}_\varepsilon^{-1}(\boldsymbol{y}_k - \boldsymbol{H}\overline{\boldsymbol{x}}_k) + \sum_{j=1}^{N+R}\overline{\boldsymbol{x}}_j^T(t_k)\boldsymbol{\Upsilon}_j^{-1}\overline{\boldsymbol{x}}_j(t_k)\right] \\ &\quad + \log\left[\frac{|\boldsymbol{\Sigma}_\varepsilon||\boldsymbol{\Upsilon}|}{|\boldsymbol{\Gamma}|}\right]\end{aligned} \tag{A.2}$$

using equation 4.28 from book [2] (*pp.*55), we get equation Eq. (17).

References

[1] K. Sekihara and S. S. Nagarajan. *Adaptive spatial filters for electromagnetic brain imaging*. Springer-Verlag, Berlin, Heidelber, 2008.

[2] Kensuke Sekihara and Srikatan S Nagarajan. *Electromagnetic brain imaging: A Bayesian perspective*. Springer-Verlag, Berlin, Heidelber, 2015.

[3] Wei Wu, Srikantan Nagarajan, and Zhe Chen. Bayesian machine learning: EEG/MEG signal processing measurements. *IEEE Signal Processing Magazine*, 33(1):14–36, 2016.

[4] David Wipf and Srikantan Nagarajan. A unified bayesian framework for MEG/EEG source imaging. *NeuroImage*, 44(3):947–966, 2009.

[5] M. S. Hämäläinen and R. J. Ilmoniemi. Interpreting measured magnetic fields of the brain: Estimates of current distributions. Technical Report TKK-F-A559, Helsinki University of Technology, 1984.

[6] M. S. Hämäläinen and R. J. Ilmoniemi. Interpreting magnetic fields of the brain: minimum norm estimates. *Med. & Biol. Eng. & Comput.*, 32:35–42, 1994.

[7] A. M. Dale, A. K. Liu, B. R. Fischl, R. L. Buckner, J. W. Belliveau, J. D. Lewine, and E. Halgren. Dynamic statistical parametric mapping: Combining fMRI and MEG for high-resolution imaging of cortical activity. *Neuron*, 26:55–67, 2000.

[8] R. D. Pascual-Marqui. Standardized low resolution brain electromagnetic tomography (sLORETA): technical details. *Methods and Findings in Experimental and Clinical Pharmacology*, 24:5–12, 2002.

[9] K. Matsuura and Y. Okabe. Selective minimum-norm solution of the biomagnetic inverse problem. *IEEE Transactions on Biomedical Engineering*, 42(6):608–615, June 1995.

[10] Kimmo Uutela, M Hämäläinen, and Erkki Somersalo. Visualization of magnetoencephalographic data using minimum current estimates. *NeuroImage*, 10(2):173–180, 1999.

[11] J. M. Zumer, H. T. Attias, K. Sekihara, and S. S. Nagarajan. A probabilistic algorithm integrating source localization and noise suppression for MEG and EEG data. *NeuroImage*, 37:102–15, 2007.

[12] Johanna M Zumer, Hagai T Attias, Kensuke Sekihara, and Srikantan S Nagarajan. Probabilistic algorithms for MEG/EEG source reconstruction using temporal basis functions learned from data. *NeuroImage*, 41(3):924–940, 2008.

[13] David P Wipf, Julia P Owen, Hagai T Attias, Kensuke Sekihara, and Srikantan S Nagarajan. Robust bayesian estimation of the location, orientation, and time course of multiple correlated neural sources using MEG. *NeuroImage*, 49(1):641–655, 2010.

[14] S. E. Robinson and D. F. Rose. Current source image estimation by spatially filtered MEG. In M. Hoke et al., editors, *Biomagnetism Clinical Aspects*, pages 761–765. Elsevier Science Publishers, 1992.

[15] M. E. Spencer, R. M. Leahy, J. C. Mosher, and P. S. Lewis. Adaptive filters for monitoring localized brain activity from surface potential time series. In *Conference Record for 26th Annual Asilomer Conference on Signals, Systems, and Computers*, pages 156–161, November 1992.

[16] B. D. Van Veen, W. Van Drongelen, M. Yuchtman, and A. Suzuki. Localization of brain electrical activity via linearly constrained minimum variance spatial filtering. *IEEE Trans. Biomed. Eng.*, 44:867–880, 1997.

[17] K. Sekihara and B. Scholz. Generalized Wiener estimation of three-dimensional current distribution from biomagnetic measurements. In C. J. Aine et al., editors, *Biomag 96: Proceedings of the Tenth International Conference on Biomagnetism*, pages 338–341, New York, 1996. Springer-Verlag.

[18] Julia P Owen, David P Wipf, Hagai T Attias, Kensuke Sekihara, and Srikantan S Nagarajan. Performance evaluation of the champagne source reconstruction algorithm on simulated and real M/EEG data. *Neuroimage*, 60(1):305–323, 2012.

[19] Hans Hallez, Bart Vanrumste, Roberta Grech, Joseph Muscat, Wim De Clercq, Anneleen Vergult, Yves D'Asseler, Kenneth P. Camilleri, Simon G. Fabri, Sabine Van Huffel, and Ignace Lemahieu. Review on solving the forward problem in eeg source analysis. *Journal of NeuroEngineering and Rehabilitation*, 4(1):46, Nov 2007.

[20] Karl Friston, Lee Harrison, Jean Daunizeau, Stefan Kiebel, Christophe Phillips, Nelson Trujillo-Barreto, Richard Henson, Guillaume Flandin, and Jrmie Mattout. Multiple sparse priors for the M/EEG inverse problem. *NeuroImage*, 39(3):1104 – 1120, 2008.

[21] Andrew Bolstad, Barry Van Veen, and Robert Nowak. Spacectime event sparse penalization for magneto-/electroencephalography. *NeuroImage*, 46(4):1066 – 1081, 2009.

[22] Srikantan S Nagarajan, Hagai T Attias, Kenneth E Hild, and Kensuke Sekihara. A probabilistic algorithm for robust interference suppression in bioelectromagnetic sensor data. *Statistics in medicine*, 26(21):3886–3910, 2007.

[23] S. S. Nagarajan, H. Attias, K. E. Hild 2nd, and K. Sekihara. A probabilistic algorithm for robust interference suppression in bioelectromagnetic sensor data. *Stat. Med.*, 20:3886–3910, 2007.

[24] Nathalie Tzourio-Mazoyer, Brigitte Landeau, Dimitri Papathanassiou, Fabrice Crivello, Olivier Etard, Nicolas Delcroix, Bernard Mazoyer, and Marc Joliot. Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. *Neuroimage*, 15(1):273–289, 2002.

[25] Hagai T Attias. M/EEG imaging by learning mean norms in brain tiles. In *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*, pages 548–551. IEEE, 2013.

[26] D. P. Wipf, J. P. Owen, H. T. Attias, K. Sekihara, and S. S. Nagarajan. Robust Bayesian estimation of the location, orientation, and time course of multiple correlated neural sources using MEG. *NeuroImage*, 49:641–655, 2010.

[27] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[28] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[29] Masa-aki Sato, Taku Yoshioka, Shigeki Kajihara, Keisuke Toyama, Naokazu Goda, Kenji Doya, and Mitsuo Kawato. Hierarchical bayesian estimation for MEG inverse problem. *NeuroImage*, 23(3):806–826, 2004.

[30] Karl Friston, Lee Harrison, Jean Daunizeau, Stefan Kiebel, Christophe Phillips, Nelson Trujillo-Barreto, Richard Henson, Guillaume Flandin, and Jérémie Mattout. Multiple sparse priors for the M/EEG inverse problem. *NeuroImage*, 39(3):1104–1120, 2008.

[31] Sarang S Dalal, JM Zumer, V Agrawal, KE Hild, Kensuke Sekihara, and SS Nagarajan. NUTMEG: a neuromagnetic source reconstruction toolbox. *Neurology & clinical neurophysiology: NCN*, 2004:52, 2004.

[32] Felix Darvas, D Pantazis, E Kucukaltun-Yildirim, and RM Leahy. Mapping human brain function with MEG and EEG: methods and validation. *NeuroImage*, 23:S289–S299, 2004.

[33] Joan G Snodgrass and June Corwin. Pragmatics of measuring recognition memory: applications to dementia and amnesia. *Journal of Experimental Psychology: General*, 117(1):34, 1988.

[34] K Sekihara. Computing Resolution for Neuromagnetic Imaging Systems. *J Comput Eng Inf Technol 5: 3. doi: http://dx. doi. org/10.4172/2324*, 9307:2, 2016.

[35] Erik Edwards, Srikantan S Nagarajan, Sarang S Dalal, Ryan T Canolty, Heidi E Kirsch, Nicholas M Barbaro, and Robert T Knight. Spatiotemporal imaging of cortical activation during verb generation and picture naming. *Neuroimage*, 50(1):291–301, 2010.

[36] Richard N Henson, Guillaume Flandin, Karl J Friston, and Jeremie Mattout. A Parametric Empirical Bayesian framework for fMRI-constrained MEG/EEG source reconstruction. *Human brain mapping*, 31(10):1512–1531, 2010.

[37] Nancy Kanwisher, Josh McDermott, and Marvin M Chun. The fusiform face area: a module in human extrastriate cortex specialized for face perception. *Journal of neuroscience*, 17(11):4302–4311, 1997.

[38] Thomas R Knösche, Markus Gräser, and Alfred Anwander. Prior knowledge on cortex organization in the reconstruction of source current densities from EEG. *Neuroimage*, 67:7–24, 2013.

[39] Makoto Fukushima, Okito Yamashita, Thomas R Knösche, and Masa-aki Sato. MEG source reconstruction based on identification of directed source interactions on whole-brain anatomical networks. *NeuroImage*, 105:408–427, 2015.

[40] Wanmei Ou, Aapo Nummenmaa, Jyrki Ahveninen, John W Belliveau, Matti S Hämäläinen, and Polina Golland. Multimodal functional imaging using fmri-informed regional EEG/MEG source estimation. *Neuroimage*, 52(1):97–108, 2010.

[41] Karl Friston, Carlton Chu, Janaina Mourão-Miranda, Oliver Hulme, Geraint Rees, Will Penny, and John Ashburner. Bayesian decoding of brain images. *Neuroimage*, 39(1):181–205, 2008.

[42] Jean Daunizeau, Jérémie Mattout, Diego Clonda, Bernard Goulard, Habib Benali, and J-M Lina. Bayesian spatio-temporal approach for EEG source reconstruction: conciliating ECD and distributed models. *IEEE Transactions on Biomedical Engineering*, 53(3):503–516, 2006.

[43] Behtash Babadi, Gabriel Obregon-Henao, Camilo Lamus, Matti S Hämäläinen, Emery N Brown, and Patrick L Purdon. A subspace pursuit-based iterative greedy hierarchical solution to the neuromagnetic inverse problem. *NeuroImage*, 87:427–443, 2014.

[44] Johanna M Zumer, Hagai T Attias, Kensuke Sekihara, and Srikantan S Nagarajan. Probabilistic algorithms for MEG/EEG source reconstruction using temporal basis functions learned from data. *NeuroImage*, 41(3):924–940, 2008.

[45] Ke Liu, Zhu Liang Yu, Wei Wu, Zhenghui Gu, Yuanqing Li, and Srikantan Nagarajan. Bayesian electromagnetic spatio-temporal imaging of extended sources with Markov Random Field and temporal basis expansion. *NeuroImage*, 139:385–404, 2016.