STATISTICS IN MEDICINE
Statist. Med. (in press)
Published online in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/sim.2941

A probabilistic algorithm for robust interference suppression in bioelectromagnetic sensor data

Srikantan S. Nagarajan[1,*,†], Hagai T. Attias[2], Kenneth E. Hild II[1] and Kensuke Sekihara[3]

[1]*Department of Radiology, University of California at San Francisco, 513 Parnassus Avenue 5362, San Francisco, CA 94122, U.S.A.*
[2]*Golden Metallic Inc., San Francisco, CA 94147, U.S.A.*
[3]*Department of Electronic Systems and Engineering, Tokyo Metropolitan Institute of Technology, 191-0065, Japan*

SUMMARY

Magnetoencephalography (MEG) and electroencephalography (EEG) sensor measurements are often contaminated by several interferences such as background activity from outside the regions of interest, by biological and non-biological artifacts, and by sensor noise. Here, we introduce a probabilistic graphical model and inference algorithm based on variational-Bayes expectation-maximization for estimation of activity of interest through interference suppression. The algorithm exploits the fact that electromagnetic recording data can often be partitioned into baseline periods, when only interferences are present, and active time periods, when activity of interest is present in addition to interferences. This algorithm is found to be robust and efficient and significantly superior to many other existing approaches on real and simulated data. Copyright © 2007 John Wiley & Sons, Ltd.

KEY WORDS: magnetoencephalography; electroencephalography; graphical models

1. INTRODUCTION

Bioelectromagnetic data are obtained by measuring electric and magnetic fields, which arise in biological tissues using a sensor array. This paper is focused on electromagnetic fields arising from the brain, but the techniques presented here apply to other biological systems, such as the heart. For brain tissues, electroencephalography (EEG) data are obtained by measuring electric fields using an electrode array placed on the scalp, and magnetoencephalography (MEG) data

*Correspondence to: Srikantan S. Nagarajan, Department of Radiology, University of California at San Francisco, 513 Parnassus Avenue 5362, San Francisco, CA 94143-0628, U.S.A.
†E-mail: srikantan.nagarajan@radiology.ucsf.edu, sri@radiology.ucsf.edu

Contract/grant sponsor: NIH; contract/grant numbers: R01DC004855, R01DC006435

Received 19 April 2007
Accepted 23 April 2007

are obtained by measuring magnetic fields using a SQUID array surrounding the head. Among existing techniques for non-invasive mapping of brain functions, both MEG and EEG have the highest temporal resolution. Both are used by basic neuroscientists in studies of brain functions. They are also used by clinicians, most commonly in patients suffering from brain tumors and epilepsy. In brain tumor patients, MEG is used to map the cognitive function of the tumor area and of neighboring areas, in order to guide neurosurgical planning, navigation, and tumor resection. Similarly, in epilepsy patients, MEG and EEG are often used to map where epileptic activity originates and to map the cognitive function of brain regions surrounding epileptogenic zones.

However, current techniques for functional brain mapping using MEG and EEG suffer from important shortcomings. The data captured by the sensor array arise not only from signal from brain sources located in areas of interest, but also from other sources, termed interference sources. These include sources in other brain areas, such as spontaneous brain activity, biological sources outside the brain, such as eye blinks, and non-biological sources, such as power lines. Signals from interference sources overlap with those from the brain sources of interest, making it difficult to accurately reconstruct the activity of desired brain areas. The task of removing interference signals from the sensor data is termed interference suppression.

This paper focuses on the stimulus-evoked experimental paradigm, which is extremely popular in EEG and MEG studies. In this paradigm, a stimulus is presented to the subject at a series of equally spaced time points. Each presentation produces activity in a set of brain sources, which generates an electromagnetic field captured by the sensor array. Those data constitute the stimulus-evoked response, and analyzing them can help to gain some insight into the mechanism used by the brain to process the stimulus and similar sensory inputs. Perhaps the most important use of stimulus evoked responses is to identify the brain locations of the sources evoked by the stimulus. Unfortunately, the presence of interference often results in very inaccurate estimates of those locations.

Many approaches to the problem of interference suppression in stimulus-evoked responses have been taken, with varying degrees of success. One common method is using a large number of stimulus presentations (100–200), also termed trials, and averaging the response across trials. The underlying assumption there is that interference signals in different trials are statistically independent, whereas evoked signals are not. Hence, averaging over sufficiently many trials would minimize interference and reveal the clean-evoked response. However, the required large number of trials results in several drawbacks. Since subjects can typically tolerate only 1–2 h of recordings in the sensor array, the number of stimulus conditions that can be obtained within an experiment is limited. Furthermore, although the evoked response may vary a little across a small number of trials, it could be non-stationary over a large number of trials. In such cases, averaging would yield an inaccurate estimate of the evoked response. Moreover, many rapid brain processes that occur within the course of single trials or a small number of trials cannot be examined by averaging across many trials.

Data-driven approaches such as principal component analysis (PCA), Wiener filtering, matched filtering, and more recently, independent component analysis (ICA), have also been used for interference suppression [1–3]. Some disadvantages of such approaches include the need to make subjective choices when running them, such as setting the threshold in PCA and selecting relevant components in ICA. An important drawback of most of those methods is their inability to exploit the pre-/post-stimulus partition of the data (see below). In the experiments section, we demonstrate that the new technique presented here significantly outperforms those methods.

DOI: 10.1002/sim

This paper presents a new technique for interference suppression in stimulus-evoked EEG/MEG data. Our approach to this problem is formulated in the framework of probabilistic graphical models with hidden variables, which has been developed considerably during the last decade in the fields of machine learning and statistics. In this approach, we describe the observed sensor data in terms of three types of unobserved signals, arising from evoked sources, interference sources, and sensor noise. Those signals are described in our model by hidden variables with their own probability distribution and depend on the sources *via* an appropriate probability distribution, derived from the physics of the problem. The model exploits the fact that the data are partitioned into two periods: pre-stimulus period, where the data include just the response of interference and sensor noise sources, and post-stimulus period, where the data also include the response of evoked sources. Combining those distributions, we obtain a probabilistic model for the sensor data. We present a variational Bayesian expectation-maximization (VB-EM) algorithm that infers the model parameters from data. VB-EM is an extension of standard EM that has two major advantages: (1) it automatically infers the optimal number of interference and evoked sources required to explain the sensor data and (2) it computes a full posterior distribution over model parameters, rather than a point estimate, which effectively prevents overfitting.

The paper is organized as follows. The probabilistic graphical model, termed partitioned factor analysis (PFA), is defined in mathematical terms in the next section. Section 3 presents the VB-EM algorithm for inferring this model from data. Section 4 provides an estimator for the clean-evoked response, i.e. the contribution of the evoked sources alone to the sensor data, using the model to remove the contribution of the interference sources. This section also presents an automatically regularized estimator of the correlation matrix of the clean-evoked response. Section 5 demonstrates, using real and simulated data, that the algorithm provides interference-robust estimates of the time course of the stimulus-evoked response. Section 6 concludes with a discussion of our results and of extensions to PFA.

2. PFA PROBABILISTIC GRAPHICAL MODEL

This section presents the PFA probabilistic graphical model, which is the focus of this paper. The PFA model describes observed EEG/MEG sensor data in terms of three types of underlying, unobserved signals: (1) signals arising from stimulus-evoked sources; (2) signals arising from interference sources; and (3) sensor noise signals. The model is inferred from data by an algorithm presented in the next section. Following inference, the model is used to separate the evoked source signals from those of the interference sources and from sensor noise, thus providing a clean version of the evoked response. In addition, it produces a regularized correlation matrix of the clean-evoked response, which facilitates localization.

Let y_{in} denote the signal recorded by sensor $i = 1 : M_y$ at time $n = 1 : N$. We assume that these signals arise from M_x evoked factors and M_u interference factors that are combined linearly. Let x_{jn} denote the signal of evoked factor $j = 1 : M_x$, and let u_{jn} denote the signal of interference factor $j = 1 : M_u$, both at time n. We use the term factor rather than source for a reason explained below. Let A_{ij} denote the evoked mixing matrix, and let B_{ij} denote the interference mixing matrix. Those matrices contain the coefficients of the linear combination of the factors that produces the data. They are analogous to the factor loading matrix in the factor analysis model. Let v_{in} denote

DOI: 10.1002/sim

the noise signal on sensor i. Mathematically

$$y_{in} = \sum_{j=1}^{M_x} A_{ij} x_{jn} + \sum_{j=1}^{M_u} B_{ij} u_{jn} + v_{in} \tag{1}$$

We use an evoked stimulus paradigm, where a stimulus is presented at a specific time, termed the stimulus onset time. The stimulus onset time is defined as $n = N_0 + 1$. The period preceding the onset $n = 1 : N_0$ is termed pre-stimulus period, and the period following the onset $n = N_0 + 1 : N$ is termed post-stimulus period. We assume that the evoked factors are active only post-stimulus and satisfy $x_{jn} = 0$ before its onset. Hence, using vector notations

$$y_n = \begin{cases} Bu_n + v_n, & n = 1 : N_0 \\ Ax_n + Bu_n + v_n, & n = N_0 + 1 : N \end{cases} \tag{2}$$

To turn (2) into a probabilistic model, each signal must be modelled by a probability distribution. Here, each evoked factor is modelled by a Gaussian distribution[‡] with zero mean and unit precision

$$p(x_{jn}) = \mathcal{N}(x_{jn} \mid 0, 1) \tag{3}$$

We model the factors as mutually statistically independent, hence

$$p(x_n) = \prod_{j=1}^{M_x} p(x_{jn}) = \mathcal{N}(x_n \mid 0, I) \tag{4}$$

For interference signals, we also employ a Gaussian model. Each interference factor is modelled by a zero-mean Gaussian distribution with unit precision, $p(u_{jn}) = \mathcal{N}(u_{jn} \mid 0, 1)$. PFA describes the factors as independent:

$$p(u_n) = \prod_{j=1}^{M_u} p(u_{jn}) = \mathcal{N}(u_n \mid 0, I) \tag{5}$$

The sensor noise is modelled by a zero-mean Gaussian distribution with a diagonal precision matrix λ,

$$p(v_n) = \mathcal{N}(v_n \mid 0, \lambda) \tag{6}$$

From (2) we obtain $p(y_n \mid x_n, u_n) = p(v_n)$, where we substitute $v_n = y_n - Ax_n - Bu_n$ with $x_n = 0$ for $n = 1 : N_0$. Hence, we obtain the distribution of the sensor signals conditioned on the evoked and interference factors,

$$p(y_n \mid x_n, u_n, A, B) = \begin{cases} \mathcal{N}(y_n \mid Bu_n, \lambda), & n = 1 : N_0 \\ \mathcal{N}(y_n \mid Ax_n + Bu_n, \lambda), & n = N_0 + 1 : N \end{cases} \tag{7}$$

[‡]A Gaussian distribution over a random vector z with mean μ and precision matrix Λ is defined by

$$\mathcal{N}(x \mid \mu, \Lambda) = \left| \frac{\Lambda}{2\pi} \right|^{1/2} \exp[-\tfrac{1}{2}(x-\mu)^{\mathrm{T}} \Lambda (x-\mu)]$$

The precision matrix is defined as the inverse of the covariance matrix.

DOI: 10.1002/sim

PFA also makes an i.i.d. assumption, meaning the signals at different time points are independent. Hence,

$$p(y \mid x, u, A, B) = \prod_{n=1}^{N} p(y_n \mid x_n, u_n, A, B)$$
$$p(x) = \prod_{n=N_0+1}^{N} p(x_n) \tag{8}$$
$$p(u) = \prod_{n=1}^{N} p(u_n)$$

where y, x, u denote collectively the signals y_n, x_n, u_n at all time points. The i.i.d. assumption is made for simplicity, and implies that the algorithm presented below can exploit the spatial statistics of the data but not their temporal statistics.

To complete the definition of PFA, we must specify prior distributions over the model parameters. For the noise precision matrix λ, we choose a flat prior, $p(\lambda) = \text{const}$. For the mixing matrices A, B, we use a conjugate prior. A prior distribution is termed conjugate w.r.t. a model when its functional form is identical to that of the posterior distribution (see the discussion below equation (A15)). We choose a prior where all matrix elements are independent zero-mean Gaussians

$$p(A) = \prod_{ij} \mathcal{N}(A_{ij} \mid 0, \lambda_i \alpha_j)$$
$$p(B) = \prod_{ij} \mathcal{N}(B_{ij} \mid 0, \lambda_i \beta_j) \tag{9}$$

and the precision of the ijth matrix element is proportional to the noise precision λ_i on sensor i. It is the λ dependence which makes this prior conjugate. (It can be shown that in the limit of zero sensor noise $\lambda \to \infty$; the impact of the prior on the posterior mean of A, B would vanish in the absence of this dependence, which would be undesirable.) The proportionality constants α_j and β_j constitute the parameters of the prior, a.k.a. hyperparameters. Equations (8), (9) together with equations (4), (5), (7) fully define the PFA model.

3. INFERRING THE PFA MODEL FROM DATA: A VB-EM ALGORITHM

This section presents an algorithm that infers the PFA model from data. PFA is a probabilistic model with hidden variables, since the evoked and interference factors are not directly observable. We use an extended version of the expectation maximization (EM) algorithm to infer the model from data. This version is termed VB-EM.

Standard EM computes the most likely parameter value given the observed data, a.k.a. the maximum *a posteriori* (MAP) estimate. In contrast, VB-EM considers all possible parameter values, and computes the probability of each value conditioned on the observed data. VB-EM therefore treats hidden variables and parameters on equal footing by computing posterior distributions for both quantities. One may, however, choose to compute a posterior only over one set of model parameters, while computing just a MAP estimate for the other set.

VB-EM is an iterative algorithm, where each iteration consists of an E-step and an M-step. The E-step computes the sufficient statistics (SS) of the hidden variables, and the M-step computes

DOI: 10.1002/sim

the SS of the parameters. (SS of an unobserved variable are quantities that define its posterior distribution.) The algorithm is iterated to convergence, which is guaranteed.

The VB-EM algorithm has several advantages compared with standard EM. It is more robust to overfitting, which can be a significant problem when working with high-dimensional but relatively short time series, as we do in this paper. It produces automatically regularized estimators, such as for the evoked response correlation matrix, whereas standard EM produces under-conditioned ones. In addition, the variance of the posterior distribution it computes (essentially the estimator's variance or squared error) provides a measure of the range of parameter values compatible with the data.

We now describe the VB-EM algorithm for the PFA model. A full derivation is provided in Appendix A.

3.1. E-step

The E-step of VB-EM computes the SS for the hidden variables conditioned on the data. For the pre-stimulus period $n = 1 : N_0$, the hidden variables are the interference factors u_n. Compute their posterior mean $\bar{u}_n$ and covariance Φ by

$$\begin{aligned} \bar{u}_n &= \Phi \bar{B}^{\mathrm{T}} \lambda y_n \\ \Phi &= (\bar{B}^{\mathrm{T}} \lambda \bar{B} + I + M_y \Psi_{BB})^{-1} \end{aligned} \tag{10}$$

where $\bar{B}$ are Ψ_{BB} are computed in the M-step by equations (15)–(17). $\bar{B}$ is the posterior mean of the interference mixing matrix, and Ψ_{BB} is related to its posterior covariance (specifically, the posterior covariance of the ith row of B is Ψ_{BB}/λ_i; see Appendix A).

For the post-stimulus period $n = N_0 + 1 : N$, the hidden variables include the evoked and interference factors x_n, u_n. To simplify the notation, we combine the evoked and interference factors into a single vector, and their mixing matrices into a single matrix. Let $L' = M_x + M_u$ be the combined number of evoked and interference factors. Let A' denote the $M_y \times L'$ matrix containing A and B, and let x'_n denote the $L' \times 1$ vector containing x_n and u_n

$$x'_n = \begin{pmatrix} x_n \\ u_n \end{pmatrix}, \quad A' = (A \;\; B) \tag{11}$$

The SS are computed as follows. At time n, compute the posterior means $\bar{x}_n$ and $\bar{u}_n$ of the evoked and interference factors, and their posterior covariance Γ, by

$$\begin{aligned} \bar{x}'_n &= \Gamma \bar{A}'^{\mathrm{T}} \lambda y_n \\ \Gamma &= (\bar{A}'^{\mathrm{T}} \lambda \bar{A}' + I + M_y \Psi)^{-1} \end{aligned} \tag{12}$$

Here, as in (11), we have combined the posterior means of the factors into a single vector $\bar{x}'_n$, and the posterior means of the mixing matrices into a single matrix $\bar{A}'$,

$$\bar{x}'_n = \begin{pmatrix} \bar{x}_n \\ \bar{u}_n \end{pmatrix}, \quad \bar{A}' = (\bar{A} \;\; \bar{B}) \tag{13}$$

where $\bar{A}$, $\bar{B}$, Ψ are computed in the M-step by equations (15)–(17). As explained in Appendix A, Ψ/λ_i is the posterior covariance of row i of A'.

DOI: 10.1002/sim

The covariances Γ_{xx} and Γ_{uu} of the evoked and interference factors, and their cross-covariance Γ_{xu}, are obtained by appropriately dividing Γ into quadrants

$$\Gamma = \begin{pmatrix} \Gamma_{xx} & \Gamma_{xu} \\ \Gamma_{xu}^{\mathrm{T}} & \Gamma_{uu} \end{pmatrix} \tag{14}$$

where Γ_{xx} is the top left $M_x \times M_x$ block of Γ, Γ_{xu} is the top right $M_x \times M_u$ block, and Γ_{uu} is the bottom right $M_u \times M_u$ block. These covariances are used in the M-step.

3.2. M-step

The M-step of VB-EM computes the SS for the model parameters conditioned on the data. We divide the parameters into two sets. The first set includes the mixing matrices A, B, for which we compute full posterior distributions. The second set includes the noise precision λ and the hyperparameters matrices α, β, for which we compute MAP estimates.

Compute the posterior means of the mixing matrices by

$$\begin{aligned} \bar{A} &= R_{yx}\Psi \\ \bar{B} &= R_{yu}\Psi \end{aligned} \tag{15}$$

where

$$\Psi = \begin{pmatrix} R_{xx} + \alpha & R_{xu} \\ R_{xu}^{\mathrm{T}} & R_{uu} + \beta \end{pmatrix}^{-1} \tag{16}$$

The quantities $R_{yx}, R_{yu}, R_{xx}, R_{xu}, R_{uu}$ are posterior correlations between the factors and the data and among the factors themselves, and are computed below. α, β are diagonal matrices with the hyperparameters α_j, β_j on the diagonal.

The covariances Ψ_{AA} and Ψ_{BB} corresponding to the evoked and interference mixing matrix (see Appendix A), and Ψ_{AB} corresponding to their cross-covariance, are obtained by appropriately dividing Ψ into quadrants

$$\Psi = \begin{pmatrix} \Psi_{AA} & \Psi_{AB} \\ \Psi_{AB}^{\mathrm{T}} & \Psi_{BB} \end{pmatrix} \tag{17}$$

where Ψ_{AA} is the top left $L \times L$ block of Ψ, Ψ_{AB} is the top right $L \times M$ block, and Ψ_{BB} is the bottom right $M \times M$ block.

Next, use those covariances to update the hyperparameter matrices α, β by

$$\begin{aligned} \alpha^{-1} &= \mathrm{diag}\left(\frac{1}{M_y}\bar{A}^{\mathrm{T}}\lambda\bar{A} + \Psi_{AA}\right) \\ \beta^{-1} &= \mathrm{diag}\left(\frac{1}{M_y}\bar{B}^{\mathrm{T}}\lambda\bar{B} + \Psi_{BB}\right) \end{aligned} \tag{18}$$

and to update the noise precision matrix λ by

$$\lambda^{-1} = \frac{1}{N}\mathrm{diag}(R_{yy} - \bar{A}R_{yx}^{\mathrm{T}} - \bar{B}R_{yu}^{\mathrm{T}}) \tag{19}$$

DOI: 10.1002/sim

3.2.1. Posterior means and correlations of the factors. Here we compute the posterior correlations, used above, between the factors and the data and among the factors themselves. Let $\bar{x}_n = \langle x_n \rangle$ and $\bar{u}_n = \langle u_n \rangle$ denote the posterior mean of the evoked and interference factors. During the pre-stimulus period $n = 1 : N_0$, $\bar{x}_n = 0$ and $\bar{u}_n$ is given by (10). During the post-stimulus period $n = N_0 + 1 : N$, they are given by (12), (13).

Let $R_{yx} = \sum_n \langle y_n x_n^{\mathrm{T}} \rangle$ and $R_{yu} = \sum_n \langle y_n u_n^{\mathrm{T}} \rangle$ denote the data-evoked and data-interference posterior correlations. Then

$$\begin{aligned} R_{yx} &= \sum_{n=N_0+1}^{N} y_n \bar{x}_n^{\mathrm{T}} \\ R_{yu} &= \sum_{n=1}^{N} y_n \bar{u}_n^{\mathrm{T}} \end{aligned} \tag{20}$$

Let $R_{xx} = \sum_n \langle x_n x_n^{\mathrm{T}} \rangle$, $R_{xu} = \sum_n \langle x_n u_n^{\mathrm{T}} \rangle$, and $R_{uu} = \sum_n \langle u_n u_n^{\mathrm{T}} \rangle$ denote the evoked–evoked, evoked–interference, and interference–interference posterior correlations. Then

$$\begin{aligned} R_{xx} &= \sum_{n=N_0+1}^{N} (\bar{x}_n \bar{x}_n^{\mathrm{T}} + \Gamma_{xx}) \\ R_{xu} &= \sum_{n=N_0+1}^{N} (\bar{x}_n \bar{u}_n^{\mathrm{T}} + \Gamma_{xu}) \\ R_{uu} &= \sum_{n=1}^{N_0} (\bar{u}_n \bar{u}_n^{\mathrm{T}} + \Phi) + \sum_{n=N_0+1}^{N} (\bar{u}_n \bar{u}_n^{\mathrm{T}} + \Gamma_{uu}) \end{aligned} \tag{21}$$

using the factors covariances (14).

Finally, let R_{yy} denote the data–data correlation

$$R_{yy} = \sum_{n=1}^{N} y_n y_n^{\mathrm{T}} \tag{22}$$

4. ESTIMATING CLEAN-EVOKED RESPONSE AND ITS CORRELATION MATRIX

In this section, we present two sets of estimators computed by the PFA model after inferring it from data. The first estimator computes the clean-evoked response. The second estimator computes a well-conditioned correlation matrix for the signals obtained by the first estimator.

Let z_{in} denote the combined contribution from all evoked factors to sensor signal i. Then

$$z_{in} = \sum_{j=1}^{M_x} A_{ij} x_{jn} \tag{23}$$

Let $\bar{z}_{in}$ denote the estimators of z_{in}. This means that $\bar{z}_{in} = \langle z_{in} \rangle$, where the average is w.r.t. the posterior over A, x. Computing this estimate amounts to obtaining a clean version of the combined contribution of the evoked factors, removing contributions from interference factors and sensor

DOI: 10.1002/sim

noise. We obtain

$$\bar{z}_{in} = \sum_{j=1}^{M_x} \bar{A}_{ij} \bar{x}_{jn} \tag{24}$$

Next, consider the correlation matrix of the evoked response, which is a required input for localization algorithms such as beamforming. Let C denote the correlation of the combined contribution from all evoked factors. Then

$$C = \sum_{n=N_0+1}^{N} z_n z_n^{\mathrm{T}} \tag{25}$$

Let $\bar{C}$ denote the estimator of C. This means, as above, that $\bar{C} = \langle C \rangle$. We obtain

$$C = \bar{A} R_{xx} \bar{A}^{\mathrm{T}} + \lambda^{-1} \operatorname{Tr}(R_{xx} \Psi_{AA}) \tag{26}$$

We point out an important fact about the estimated correlation matrix $\bar{C}$. It is always well conditioned, due to the diagonal Ψ_{AA} term. Hence, the VB-EM approach automatically produces regularized correlation matrix. Note that the correlation matrix obtained directly from the signal estimates, $\sum_n \bar{z}_n \bar{z}_n^{\mathrm{T}}$, is under-conditioned.

5. MODEL-ORDER SELECTION, INITIALIZATION AND COMPLEXITY

One advantage of the algorithm presented here is that it offers a principled method of model-order selection. Model-order selection in PFA algorithm refers to the choice of M_x and M_u. The MAP estimates of the hyperparameters of the mixing matrices can be used to estimate the number of factors by thresholding. Alternatively, we can compute the maximum of the posterior over model structure $q(M_x, M_u|y)$, which is equivalent to maximizing the marginal log likelihood $\log p(y|M_x, M_u)$. The marginal log likelihood obtained by integrating over all hidden variables is also referred to as the evidence. The evidence penalizes complexity and corresponds to the Bayesian information criterion (BIC) and the minimum description length (MDL) for infinite data [4]. It can be shown that the evidence is lower bounded by a free energy objective function $\mathscr{F}$, as defined in equations (A5) and (A6). Therefore, after computing $\mathscr{F}$ for different model orders M_x and M_u, we can choose

$$\bar{M}_x, \bar{M}_u = \underset{M_x, M_u}{\operatorname{argmax}} \mathscr{F}(M_x, M_u)$$

Although, the proposed algorithm is fairly robust to initialization, the specific initializations of the parameters that we use in the Results section are as follows. We initialize the mixing matrix B to the dominant eigen-vectors of the data obtained in the pre-stimulus period. The evoked factor mixing matrix A is initialized as the dominant eigen-vectors of the post-stimulus data after pre-whitening with the pre-stimulus data covariance. λ is initialized to be uniform across sensors and equal to the inverse of the least-significant eigenvalue of the pre-stimulus data. α and β are initially assumed to be identity matrices.

For each iteration of the algorithm, the computational complexity of estimation of the PFA graphical model is $O(N * (M_x + M_u) * M_y)$. So, PFA is linear in the number of time samples, sensors and factors.

DOI: 10.1002/sim

6. RESULTS

6.1. Simulations

Figure 1 shows an example of performance for the proposed interference suppression algorithm on simulated data. The top row shows simulated noisy MEG data created assuming three brain sources and 25 interference sources and 275 sensors. The middle row shows the true signal that is present in the post-stimulus period within the noisy MEG data. The bottom row shows the estimated signal extracted by PFA. When the true signal y^* is known, denoising performance can be quantified using the output signal-to-noise/interference ratio (SNIR)

$$\text{SNIR} = \frac{1}{M_y} \sum_{m=1}^{M_y} 10 \log_{10} \frac{\sum_{n=N_0}^{N} y^{*2}_{m,n}}{\sum_{n=N_0}^{N} (y^*_{m,n} - \bar{y}_{m,n})^2} \text{ (dB)}$$

For the example shown, the input SNIR is -13 dB and the output SNIR is -2 dB.

In more extensive simulations, we compare interference suppression performance for the proposed probabilistic algorithm with five other standard methods used in practice—PCA [5], Wiener Filtering [2], ICA using TDSEP [6] and/or FastICA [7]), and trial averaging. TDSEP and FastICA were chosen as the representative ICA methods based on their low computational complexity. Furthermore, when there are more than about 50 sensors, as is typically for high-resolution EEG, MEG, or magnetocardiography (MCG) systems, TDSEP and FastICA do not require additional dimensionality reduction. We report the better results between these two ICA algorithms. With the exception of the trial mean, all the above interference suppression methods are spatial filtering methods that apply a linear transformation that is applied to the observed data to obtain an estimate of the underlying signal.

The proposed algorithm, and the comparison methods mentioned above, could be applied either to concatenated single-trial data or to trial-averaged data. For interference suppression performance,

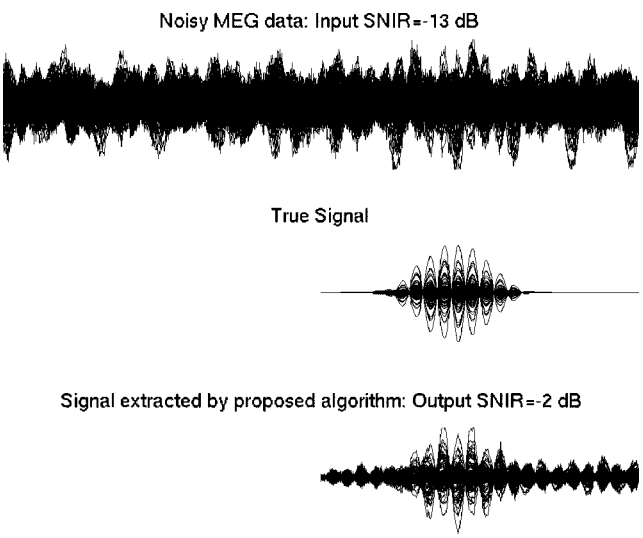

Figure 1. Example of performance of the proposed algorithm.

DOI: 10.1002/sim

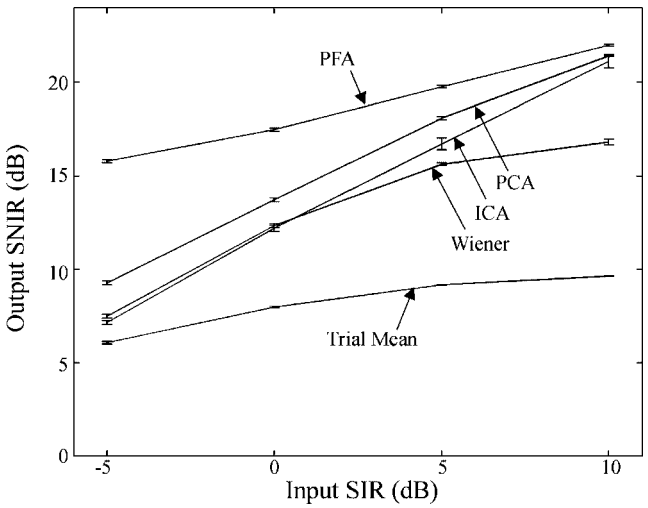

Figure 2. Output SNIR as a function of the input SIR for 10 trials and input SNR = 0 dB.

we first apply each method to the trial-averaged data so that we can directly compare it with the trial mean. In some cases (as noted), we also apply the interference suppression on single-trial data and then compute the trial average.

For the simulation results below, there are 1000 data points per trial (the first 63% of which corresponds to the inactive period), $M_y = 132$ sensors, $M_x = 2$ factors, $M_s = 2$ sources, and $M_u = 1000$ interference signals. Results shown represent the mean over 10 Monte Carlo repetitions and the error bars are used to indicate one standard error of the mean. The input signal-to-interference ratio (SIR) is the ratio of the power of the factors to the power of the interferences (measured in sensor space). Likewise, the input signal-to-noise ratios (SNR) is the ratio of the power of the factors to the power of the additive noise. The number of factors, M_x, must be specified for all denoising methods except the trial mean. The proposed method must also be supplied with a known number of interference signals, M_u. To simplify the comparisons, it is assumed that the number of factors is the true number and the number of interference signals $M_u = 50$.

Figure 2 shows the interference suppression performance as a function of the input SIR. Only The input SNR is held constant at 0 dB and the number of trials is 10. All of the methods perform better than the trial mean. PFA performs the best across all input SIR. The performances of both PCA and Wiener approach that of PFA as the input SIR increases.

Figure 3 shows the interference suppression performance as a function of the number of trials. The input SIR and input SNR are held constant at -5 and 0 dB, respectively. In this figure the trial mean outperforms TDSEP and PFA outperforms the other four methods.

6.2. Model-order selection

We demonstrate robustness to model-order selection using the PFA criterion with simulated data. Figures 4 and 5 show the results of using the PFA criterion, the evidence under the variational approximation, as a function of model order. For these two figures there are $M_x^* = 2$ sources, $M_u^* = 10$ interferences, 1000 data per trial, and 10 trials of data. Figure 4 plots the PFA criterion

DOI: 10.1002/sim

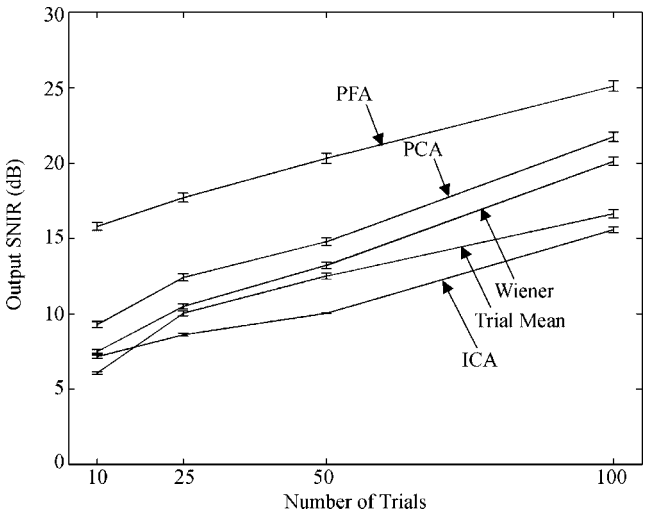

Figure 3. Output SNIR as a function of the number of trials for input SIR $= -5$ dB and input SNR $= 0$ dB.

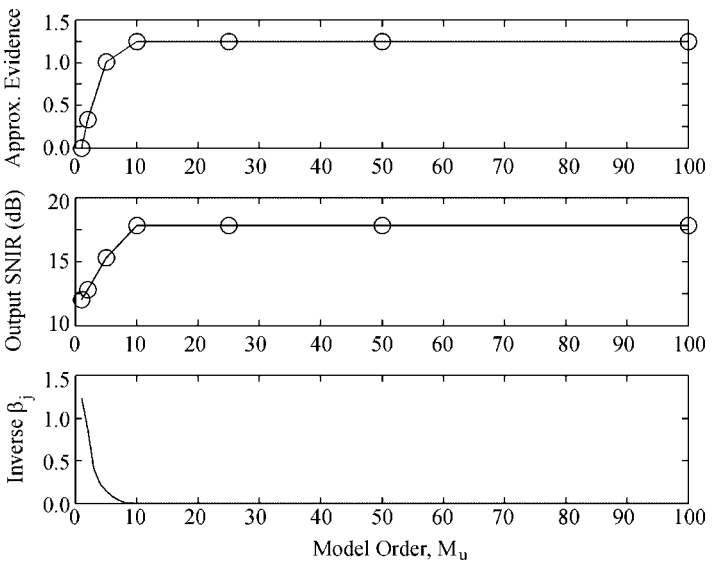

Figure 4. Performance as a function of M_u, where M_x is assumed to be 2. The true values of M_x and M_u are 2 and 10, respectively.

as a function of M_u, where it is assumed that $M_x = 2$. Also shown are the plots of output SNIR and the amplitude of the estimated (inverse) hyperparameters corresponding to the scaled variance of the columns of the mixing matrix. Note that the output SNIR asymptotes for higher model orders because the columns of the mixing matrices comprise elements with values near zero. The PFA criterion matches the output SNIR quite well (note that the precise value of the PFA criterion

DOI: 10.1002/sim

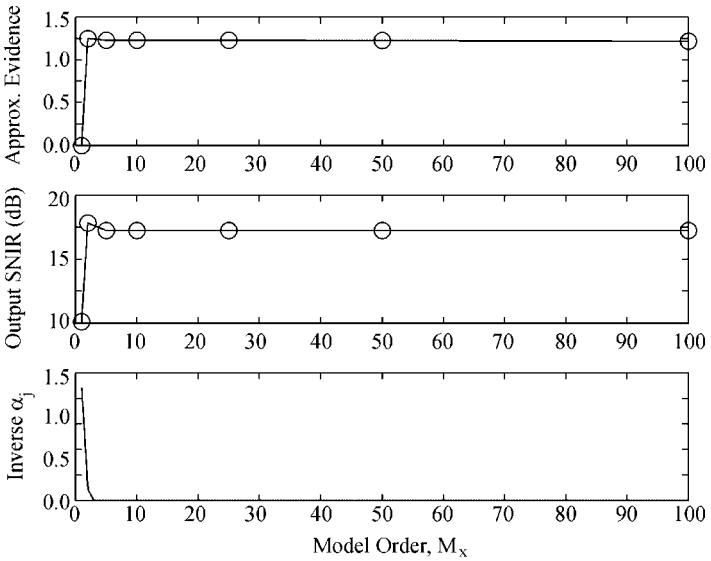

Figure 5. Performance as a function of M_x, where M_u is assumed to be 15. The true values of M_x and M_u are 2 and 10, respectively.

can be obtained for real data, whereas the precise value of the output SNIR cannot). Figure 5 plots the PFA criterion as a function of M_x, where it is assumed that $M_u = 15$. The plot of the PFA criterion *versus* model-order peaks at the correct value of $\bar{M}_x = 2$, where the output SNIR also peaks. Moreover, increasing the specified model order beyond the true model order does not contribute to significant deterioration in performance, hence our use of the term 'robust interference suppression'.

6.3. PFA as preprocessing for ICA

The stimulus-evoked factors in PFA can be subsequently separated using ICA algorithms. Here, we compare the performance on source separation using ICA after preprocessing with the proposed and comparative algorithms. Source extraction performance is measured using the output source-to-distortion ratio (SDR), where the distortion for source estimate m includes noise, interference, and all sources except one. For the case of no permutations, the SDR is defined by

$$\mathrm{SDR} = \frac{1}{M_s} \sum_{m=1}^{M_s} \mathrm{SDR}_m \ \text{(dB)}$$

where

$$\mathrm{SDR}_m = 10 \log_{10} \frac{1}{M_s} \sum_{m'=1}^{M_s} \left(\frac{1}{2 - \frac{2}{N - N_0} \left| \sum_{n=N_0+1}^{N} s_{m,n} \bar{s}_{m',n} \right|} \right) \ \text{(dB)}$$

DOI: 10.1002/sim

$s_n = W^{-1}x_n$ is the true source vector at time n, and both $s_{m,n}$ and $\bar{s}_{m,n}$ are normalized to have unit variance. The definition above is easily extended to account for any possible permutation. This metric reflects the performance of both the interference suppression/dimension reduction algorithm and the ICA algorithm. The interference suppression method accounts for all differences in SDR performance below, since, for each experiment, the same ICA algorithm is used. In general, we found that TDSEP performed better than FastICA for denoising and FastICA performed better than TDSEP for source extraction. For TDSEP and FastICA, the source subspace is automatically determined by selecting the components that have the largest ratio of active power to inactive power. The first component is given by

$$\bar{m}_1 = \underset{m}{\operatorname{argmax}} \frac{\sum_{n=N_0}^{N} \bar{s}_{m,n}^2}{\sum_{n=1}^{N_0-1} \bar{s}_{m,n}^2}$$

and the subsequent $M_x - 1$ components are found in a similar manner.

Figure 6 shows the source extraction performance as a function of the input SIR. The input SNR is 0 dB and the number of trials is 10. The non-ICA denoising methods are used to reduce the dimensionality of the data from 132 to 2 prior to applying the ICA algorithm, which in this case is FastICA. Also shown are the results for FastICA when no dimension reduction method is used. PFA produces the best overall results and is the least sensitive to input SIR. The results reported here for FastICA (with no dimension reduction) indicate that denoising/dimension reduction preprocessing is advantageous when the input SIR is low ($<$10 dB).

Figure 7 shows the source extraction performance as a function of the number of trials. As before, the input SIR and input SNR are -5 and 0 dB, respectively, and the results of using FastICA with no dimension reduction are included. PFA (combined with FastICA) performs the best and FastICA (with no dimension reduction) performs the worst. These results indicate that 10 trials of 1000 data points per trial is already sufficiently large so that no improvement in separation

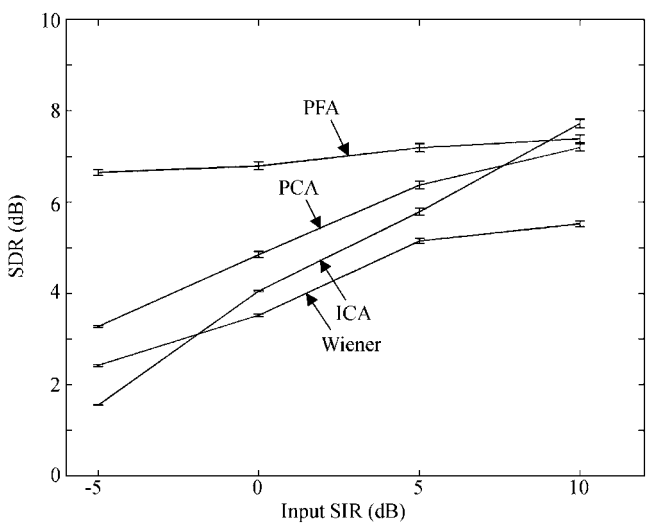

Figure 6. Output SDR as a function of the input SIR for 10 trials and input SNR = 0 dB.

DOI: 10.1002/sim

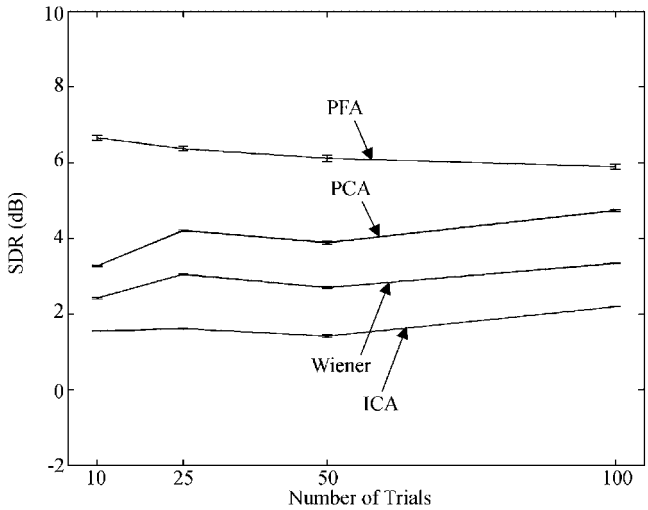

Figure 7. Output SDR as a function of the number of trials for input SIR $= -5$ dB and input SNR $= 0$ dB.

performance is obtained by additional increases in data length. This is not expected to be the case if the input SIR and/or input SNR are increased.

6.4. Real data

An example of performance of the proposed interference suppression algorithm on auditory-evoked magnetic fields measured across the whole head obtained from a 275-channel sensor array in response to a 1 kHz tone pip is shown in Figure 8. For these data, $M_y = 274$, the data length is 720 samples per trial (170 pre-stim), and there are 109 trials. Averaged data from 20 trial averages are noisy as shown in the top left for select channels. The output of PFA is shown in the middle left. Also shown is the response obtained from averaging 109 trials. It can be seen that the response from 20 trials does not resemble the 109 trial average (shown in the bottom left) suggesting trial-to-trial variability or non-stationarity in the evoked response over 109 trials. The right column shows waveforms for the noisy input (thin lines) and interference-suppressed output (thick lines) for selected individual channel waveforms.

Figure 9 shows the denoising for a different MEG data set. Here, we examine the stimulus-evoked response to a somatosensory stimulus with $M_y = 274$ and the data length is 361 samples per trial. In this figure, 0 ms corresponds to N_0, which is the onset of the stimulus. We assume that M_x is 2, M_u is 50. For comparison, PCA denoising on the 10-trial average is also shown, as well as the average across 525 trials. It can be seen that PFA performs adequate interference suppression of the evoked response.

Quantifying performance of interference suppression with real data is difficult because output SNIR and SDR can be easily computed only for simulated data since y^*, s are not known for real data. The output SNIR can, however, be used with real data if $y_n^* = Ax_n$ can be approximated. In this latter example, the average response obtained from 525 trials appears to be more similar to the response to 10 trials, suggesting stationarity in the evoked response. Furthermore, five principal components explain 97% of the total energy of the trial-averaged data. Therefore, for this real data

DOI: 10.1002/sim

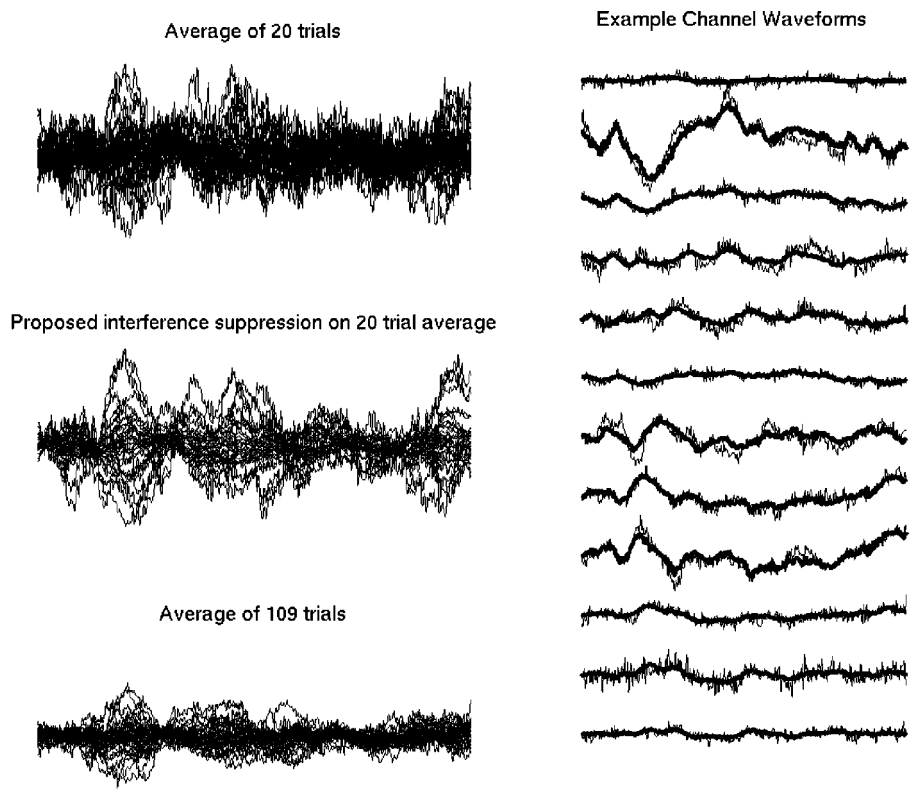

Figure 8. Example of performance on real auditory-evoked magnetic field data.

we replace y_n^* with the sensor signals due to the five principal components of the trial-averaged sensor data obtained from 525 trials.

Figure 10 shows denoising performance as a function of the number of trials using the above-mentioned procedure. None of the estimated sources produced by the ICA method resembled the desired signals even when the number of trials was increased to 50. The performance of ICA denoising depends on being able to correctly select the M_x sources and the results show poor performance of ICA on this data. Results for PFA, PCA, and Wiener are better than those produced by the trial mean (when the trial mean uses the same number of trials). PFA performs the best of these methods, although PCA performs almost identically when the number of trials equals or exceeds 30.

Figures 11 and 12 show the results of model-order selection for the auditory MEG data set shown above. For these two figures, only the 20 trials of data are used. Figure 11 plots the evidence as a function of M_u, where it is assumed that $M_x = 2$. Figure 12 plots the evidence as a function of M_x, where it is assumed that $M_u = 25$. Also shown are the plots of the amplitude of the associated inverse hyperparameters. The model orders that maximize the PFA criterion are $\bar{M}_u = 25$ and $\bar{M}_x = 5$. It can be seen that the evidence peaks for small model orders and that the posterior estimates of many of the inverse hyperparameters are zero, thereby demonstrating the built-in model-order robustness of the PFA inference algorithm.

DOI: 10.1002/sim

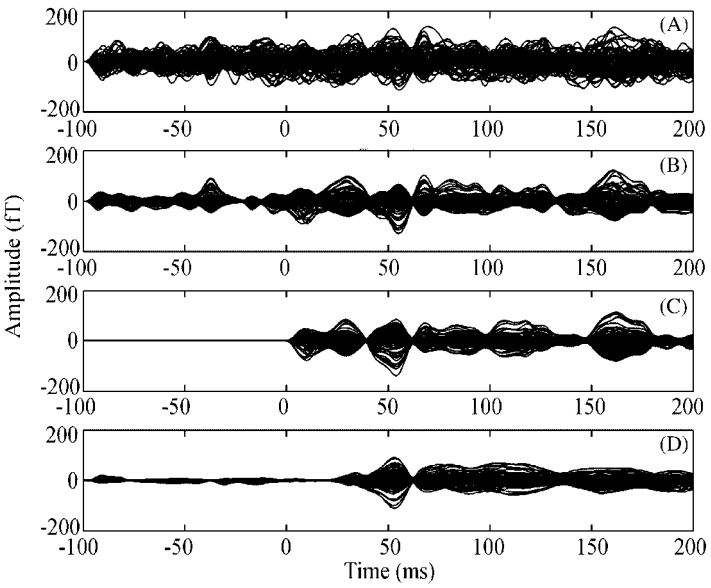

Figure 9. Time series after interference suppression of real MEG data from somatosensory cortex. (A) Trial Mean (10 trials); (B) PCA (10 trials); (C) PFA (10 trials); and (D) Trial mean (525 trials).

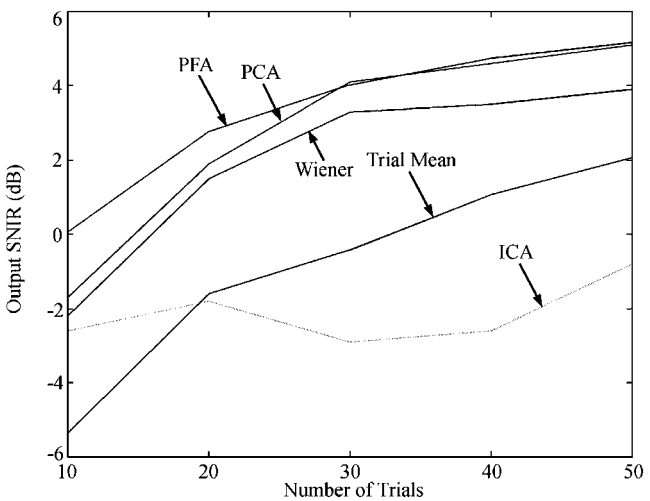

Figure 10. Output SNIR as a function of the number of trials for real MEG data.

Figure 13 shows the interference suppression of real EEG data, where $M_y = 119$, the data length is 720 samples per trial (170 in the pre-stimulus period), the number of trials is 120, and the data were the response to an auditory 1 kHz tone. Results for the trial mean are shown for both 10 and 120 trials. Notice that data contain a large 60 Hz contribution (further examination reveals that

DOI: 10.1002/sim

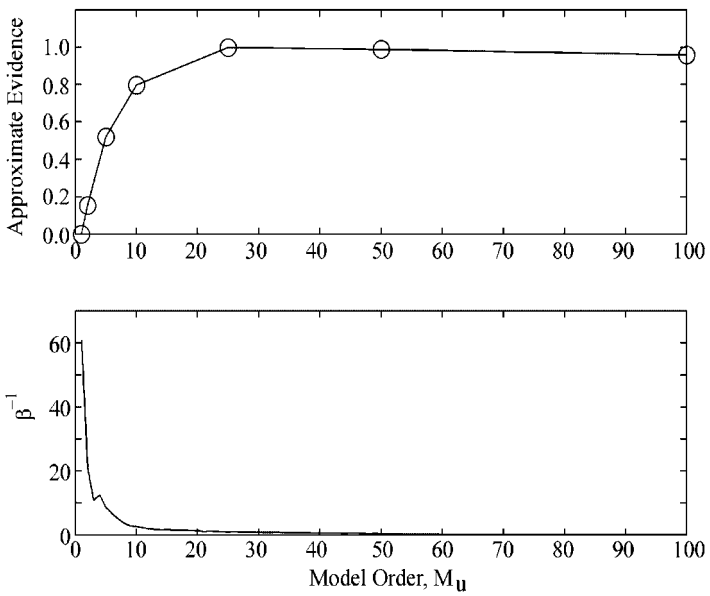

Figure 11. Model-order selection as a function of M_u, where M_x is assumed to be 2.

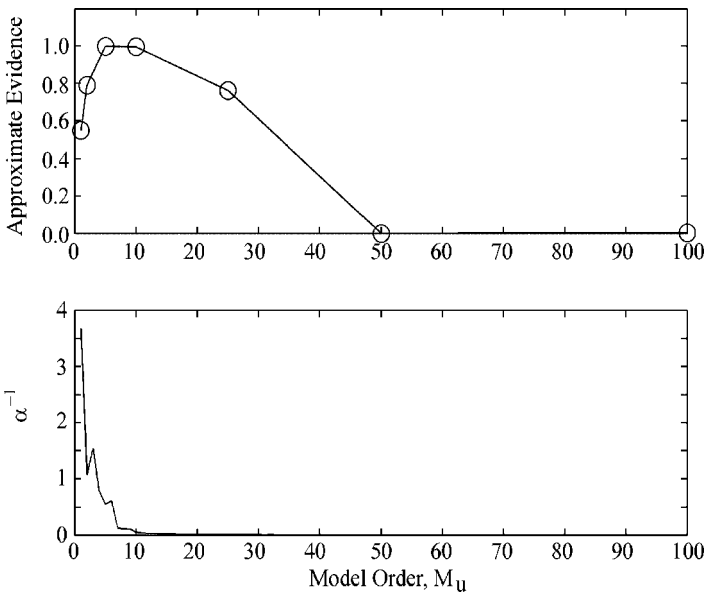

Figure 12. Model-order selection as a function of M_x, where M_u is assumed to be 25.

most of the line noise is concentrated in three channels). The proposed algorithm uses 10 trials and assumes that there are $M_x = 5$ sources and $M_u = 25$ interferences. Since the 60 Hz oscillations occur during both pre-stimulus and post-stimulus periods, our algorithm treats it as an interference

DOI: 10.1002/sim

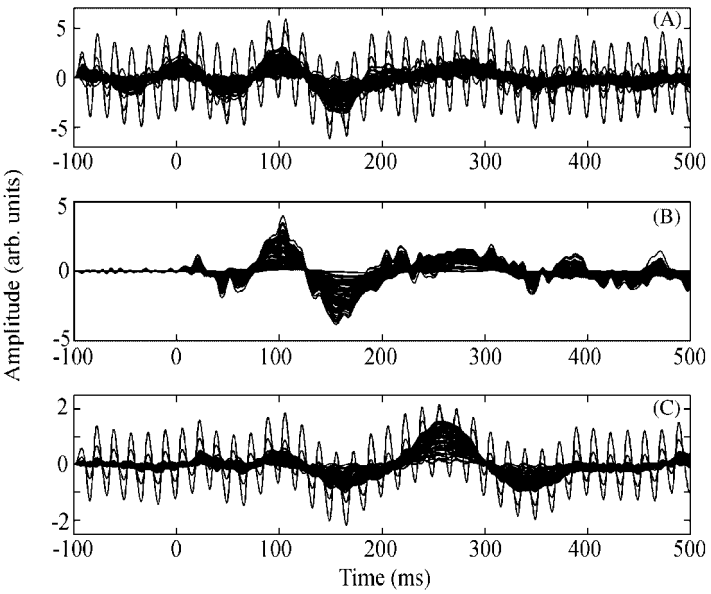

Figure 13. Interference suppression of real EEG data. (A) Trial Mean (10 trials); (B) PFA (10 trials); and (C) Trial mean (120 trials).

signal and is therefore able to remove it successfully. Simple temporal filtering, which can also be used to remove the line noise, will necessarily repress other activity in and near the 60 Hz frequency, whereas this does not occur with our algorithm. The trial mean, on the other hand, is unable to remove the line noise since the stimulus onset is approximately synchronous with the line noise. The P1, N1, and P2 responses are clearly visible in the output after interference suppression (the convention of inverting the polarity, commonly used in EEG analyses, is not used here).

7. DISCUSSION

The robustness of the proposed algorithm to the choice of the maximal model orders, M_x, and M_u, can be explained using a process known as automatic relevance determination (ARD). The hyperparameters represent the inverse power of the associated factor/interference signal. When the model order is chosen larger than necessary, the hyperparameters associated with the redundant signals approach infinity [8–10]. As a hyperparameter approaches infinity, the observations can be explained without the associated factor/interference signal. The cost of overestimating the model order is that the computational complexity increases as either M_u or M_x is increased. The tendency of the hyperparameters to approach infinity can be used to estimate the two model orders. The most straightforward way to estimate M_u, for example, is to count the number of diagonal elements of β that have an inverse value less than a given threshold. In the previous section, we showed results using the evidence, which does not require an arbitrary selection of a threshold.

DOI: 10.1002/sim

The proposed algorithm, as given above, has a potential problem of identifiability between the estimation of A and B, especially if the amount of data in the pre-stimulus period is small and both A and B are primarily estimated from the post-stimulus period (data may have equal likelihood to arise from a source factor and from an interference factor). To avoid this problem, we perform a two-step procedure for PFA. In the first step, we estimate the interference factor mixing matrix and the sensor noise precision from data in the pre-stimulus period. In this case, the update equation for λ uses only the pre-stimulus data and is

$$\lambda^{-1} = \frac{1}{N} \operatorname{diag}(R_{yy} - \bar{B} R_{yu}{}^{\mathrm{T}}) \tag{27}$$

The update rules for B and β are the same as listed in equation (20), with a modified $\Psi_{BB} = (R_{uu} + \beta)^{-1}$. Subsequently, for post-stimulus data, we freeze the above parameters and estimate A using a modified update rule,

$$\bar{A} = (R_{yx} - B R_{yx}) \Psi_{AA} \tag{28}$$

where $\Psi_{AA} = (R_{xx} + \alpha)^{-1}$. All other update rules are identical to those listed above.

Furthermore, the proposed model currently assumes that the interferences are statistically stationary between the pre- and post-stimulus periods. However, we can relax this assumption and model non-stationary changes in the power of interference if there are no changes in the location of the interferences. We assume that, in the post-stimulus period, the probability distribution of the interference factors is $p(u_n) = \prod_{m=1}^{M_u} p(u_{m,n}) = \mathcal{N}(u_n|0, v)$, where v is a diagonal precision matrix that is equal to the inverse of the power fluctuations of the interference in the post-stimulus period. In this case, we can learn v from the post-stimulus period using the update rule $v^{-1} = \operatorname{diag}((1/N) R_{uu})$, where R_{uu} is calculated only for the post-stimulus period.

The algorithm currently assumes that the prior distributions for evoked and interference factors are i.i.d. and invariant to the time-index permutation. However, this does not appear to impact performance because in all the simulations presented in the paper both the background sources were assumed to be sinusoidal (with bimodal distributions) or damped sinusoids (with super-Gaussian distributions), rather than Gaussians as assumed in the model. Moreover, since the performance of the algorithm is also good on real bioelectromagnetic data, where the interference factors are indeed oscillatory, the algorithm has some degree of robustness with respect to assumptions about the prior distribution of interference and evoked factors. Since estimation is data dependent, if the data suggest that factors have temporal continuity, then the estimated factors will have some smoothness. Nevertheless, an algorithm that is able to exploit temporal correlation in factors could potentially be more powerful. We are currently pursuing such an extension, using several different models that incorporate temporal statistics of the evoked and interference factors, whose parameters are inferred from data. Algorithms derived from such models perform interference suppression using not just spatial but also spatio-temporal filtering. On a separate note, since bioelectromagnetic data are often non-Gaussian, we are currently extending the model to incorporate non-Gaussian factor models.

APPENDIX A: THE VB-EM ALGORITHM

This section outlines the derivation of the VB-EM algorithm that infers the PFA model from data.

DOI: 10.1002/sim

A.1. Model

The full joint distribution of the PFA model is given by

$$p(y, x, u, A, B) = p(y \mid x, u, A, B)p(x)p(u)p(A)p(B) \tag{A1}$$

together with equations (5), (7), (8).

A.2. Variational Bayesian inference

The Bayesian approach, as discussed above, treats hidden variables and parameters on equal footing: both are unobserved quantities for which posterior distributions must be computed. A direct application of Bayes rule to the PFA model would compute the joint posterior over the hidden variables x, u and parameters A, B

$$p(x, u, A, B \mid y) = \frac{1}{p(y)} p(y, x, u, A, B) \tag{A2}$$

where the normalization constant $p(y)$, termed the marginal likelihood, is obtained by integrating over all other variables

$$p(y) = \int \mathrm{d}x\, \mathrm{d}u\, \mathrm{d}A\, \mathrm{d}B\; p(y, x, u, A, B) \tag{A3}$$

However, this exact posterior is computationally intractable, because the integral above cannot be obtained in closed form.

The VB approach approximates this posterior using a variational technique. The idea is to require the approximate posterior to have a particular factorized form, and then optimize it by minimizing the Kullback–Leibler (KL) distance from the factorized form to the exact posterior $\int q \log(p/q)$ [11].

Here, we choose a form which factorizes the hidden variables from the parameters given the data

$$p(x, u, A, B \mid y) \approx q(x, u, A, B \mid y) = q(x, u \mid y)q(A, B \mid y) \tag{A4}$$

It is worth emphasizing that (1) beyond the factorization assumption, we make no further approximation when computing q, and (2) the factorized form still allows correlations among x, u, as well as among the matrix elements of A, B, conditioned on the data.

Rather than minimize the KL distance directly, it is convenient to start from an objective function defined by

$$\mathscr{F}[q] = \int \mathrm{d}x\, \mathrm{d}u\, \mathrm{d}A\, \mathrm{d}B q(x, u, A, B \mid y)[\log p(y, x, u, A, B) - \log q(x, u, A, B \mid y)] \tag{A5}$$

It can be shown that

$$\mathscr{F}[q] = \log p(y) - KL[q(x, u, A, B \mid y) \,||\, p(x, u, A, B \mid y)] \tag{A6}$$

and, since the marginal likelihood $p(y)$ is independent of q, maximizing $\mathscr{F}$ w.r.t. q is equivalent to minimizing the KL distance. Furthermore, $\mathscr{F}$ is upper bounded by $\log p(y)$ because the KL distance is always non-negative. Hence, any algorithm that successively maximizes $\mathscr{F}$, such as VB-EM, is guaranteed to converge.

DOI: 10.1002/sim

A.3. Derivation of VB-EM

VB-EM is derived by alternately maximizing $\mathscr{F}$ w.r.t. the two components of the posterior q. In the E-step one maximizes w.r.t. the posterior over hidden variables $q(x, u \mid y)$, keeping the second posterior fixed. In the M-step one maximizes w.r.t. the posterior over parameters $q(A, B \mid y)$, keeping the first posterior fixed. When performing maximization, normalization of q must be enforced by adding two Lagrange multiplier terms to $\mathscr{F}$ in (A5).

Maximization is performed by setting the gradients to zero:

$$\begin{aligned}\frac{\partial \mathscr{F}}{\partial q(x, u \mid y)} &= \langle \log p(y, x, u, A, B)\rangle_2 - \log q(x, u \mid y) + C_1 = 0 \\ \frac{\partial \mathscr{F}}{\partial q(A, B \mid y)} &= \langle \log p(y, x, u, A, B)\rangle_1 - \log q(A, B \mid y) + C_2 = 0\end{aligned} \tag{A7}$$

where C_1, C_2 are constants depending only on the data y. $\langle\cdot\rangle_1$ denotes averaging only w.r.t. $q(x, u \mid y)$, and $\langle\cdot\rangle_2$ denotes averaging only w.r.t. $q(A, B \mid y)$. Hence, the posteriors are given by

$$\begin{aligned}q(x, u \mid y) &= \frac{1}{Z_1} \exp[\langle \log p(y, x, u, A, B)\rangle_2] \\ q(A, B \mid y) &= \frac{1}{Z_2} \exp[\langle \log p(y, x, u, A, B)\rangle_1]\end{aligned} \tag{A8}$$

where Z_1, Z_2 are normalization constants.

A.4. E-step

It follows from (A8) that the posterior over u, x factorizes over time, and has different pre- and post-stimulus forms,

$$q(u, x \mid y) = \prod_{n=1}^{N_0} q(u_n \mid y_n) \cdot \prod_{n=N_0+1}^{N} q(u_n, x_n \mid y_n) \tag{A9}$$

It also follows that in the pre-stimulus period $q(u_n \mid y_n)$ is Gaussian in u_n, and in the post-stimulus period $q(u_n, x_n \mid y_n)$ is Gaussian in u_n, x_n. To see this, consider $\log q(x, u \mid y)$ in (A8) and observe that it is a sum over n, where the nth element depends only on x_n, u_n and the dependence is quadratic.

For the pre-stimulus period we obtain

$$q(u_n \mid y_n) = \mathscr{N}(u_n \mid \bar{u}_n, \Phi^{-1}) \tag{A10}$$

with mean $\bar{u}_n$ and covariance matrix Φ given by (10). (One first obtains $\Phi = (\langle B^{\mathrm{T}} \lambda B\rangle + I)^{-1}$, and then performs the average using (A18).) For the post-stimulus period, the posterior is also Gaussian

$$q(x_n, u_n \mid y_n) = q(x'_n \mid y_n) = \mathscr{N}(x'_n \mid \bar{x}'_n, \Gamma^{-1}) \tag{A11}$$

with mean $\bar{x}'_n$ and covariance matrix Γ^{-1} given by (12) (as for Φ above, one first obtains $\Gamma = (\langle A'^{\mathrm{T}} \lambda A'\rangle + I)^{-1}$, then applies (A18)).

DOI: 10.1002/sim

It is useful to make explicit the correlations among the factors implied by their posteriors (A10), (A11). For the pre-stimulus period, we obtain

$$\langle u_n u_n^{\mathrm{T}} \rangle = \bar{u}_n \bar{u}_n^{\mathrm{T}} + \Phi \tag{A12}$$

For the post-stimulus period, we obtain $\langle x'_n \rangle = \bar{x}'_n$ and $\langle x'_n x_n'^{\mathrm{T}} \rangle = \bar{x}'_n \bar{x}_n'^{\mathrm{T}} + \Gamma$. In terms of x_n, u_n

$$\begin{aligned} \langle x_n \rangle &= \bar{x}_n \\ \langle u_n \rangle &= \bar{u}_n \\ \langle x_n x_n^{\mathrm{T}} \rangle &= \bar{x}_n \bar{x}_n^{\mathrm{T}} + \Gamma_{xx} \\ \langle u_n u_n^{\mathrm{T}} \rangle &= \bar{u}_n \bar{u}_n^{\mathrm{T}} + \Gamma_{uu} \\ \langle x_n u_n^{\mathrm{T}} \rangle &= \bar{x}_n \bar{u}_n^{\mathrm{T}} + \Gamma_{xu} \end{aligned} \tag{A13}$$

where we have used (13), (14).

A.5. M-step

It follows from (A8) that the parameter posterior factorizes over the rows of the mixing matrices, and correlates their columns. Let w^i denote a column vector containing the ith row of the combined mixing matrix $A' = (A, B)$

$$A' = \begin{pmatrix} \cdots & w^1 & \cdots \\ \cdots & w^2 & \cdots \\ \cdots & \cdots & \cdots \\ \cdots & w^{M_y} & \cdots \end{pmatrix} \tag{A14}$$

so $w^i_j = A'_{ij}$. Then, the posterior over each row is Gaussian

$$q(A, B \mid y) = q(A' \mid y) = \prod_{i=1}^{M_y} \mathcal{N}(w^i \mid \bar{w}^i, \lambda_i \Psi^{-1}) \tag{A15}$$

with mean $\bar{w}^i_j = \bar{A}_{ij}$ computed by (15). The precision matrix $\lambda_i \Psi^{-1}$ is computed using (16). To see this, consider $\log q(A, B \mid y)$ in (A8) and observe that it is a sum over i, where the ith element depends only on the ith rows of A, B and the dependence is quadratic.

It is now evident that $p(A, B)$ of equation (9) is indeed a conjugate prior. Rewriting it in the form

$$p(A, B) = p(A') = \prod_{i=1}^{M_y} \mathcal{N}(w^i \mid 0, \lambda_i \alpha') \tag{A16}$$

where α' is a diagonal matrix with the hyperparameter matrices α, β on its diagonal, shows that its functional form is identical to that of the posterior (A15), with Ψ^{-1} replacing α'.

DOI: 10.1002/sim

It is useful to make explicit the correlations among the elements of the mixing matrices implied by their posterior (A15). They are $\langle A'_{ij} A'_{kl} \rangle = \bar{A}'_{ij} \bar{A}'_{kl} + \delta_{ik} \Psi_{jl} / \lambda_i$, or, in terms of A, B,

$$\begin{aligned} \langle A_{ij} A_{kl} \rangle &= \bar{A}_{ij} \bar{A}_{kl} + \delta_{ik} \frac{1}{\lambda_i} (\Psi_{AA})_{jl} \\ \langle B_{ij} B_{kl} \rangle &= \bar{B}_{ij} \bar{B}_{kl} + \delta_{ik} \frac{1}{\lambda_i} (\Psi_{BB})_{jl} \\ \langle A_{ij} B_{kl} \rangle &= \bar{A}_{ij} \bar{B}_{kl} + \delta_{ik} \frac{1}{\lambda_i} (\Psi_{AB})_{jl} \end{aligned} \tag{A17}$$

where we used (17). It follows that

$$\begin{aligned} \langle A^{\mathrm{T}} \lambda A \rangle &= \bar{A}^{\mathrm{T}} \lambda \bar{A} + M_y \Psi_{AA} \\ \langle B^{\mathrm{T}} \lambda B \rangle &= \bar{B}^{\mathrm{T}} \lambda \bar{B} + M_y \Psi_{BB} \\ \langle A'^{\mathrm{T}} \lambda A' \rangle &= \bar{A}'^{\mathrm{T}} \lambda \bar{A}' + M_y \Psi \end{aligned} \tag{A18}$$

which are needed for (10), (12).

To obtain the update rules for the hyperparameters (18), observe that the part of the objective function $\mathscr{F}$ (A5) that depends on α, β is

$$\langle \log p(A) + \log p(B) \rangle \tag{A19}$$

where the averaging is w.r.t. the posterior q. Next, compute the derivative of this expression w.r.t. α, β and set it to zero. The solution of the resulting equation is (18). It is easier to first compute the derivative and then apply the average. Similarly, to obtain the update rule for the noise precision (19), observe that the part of $\mathscr{F}$ that depends on λ is

$$\langle \log p(y \mid x, u, A, B) + \log p(A) + \log p(B) \rangle \tag{A20}$$

and set its derivative w.r.t. λ to zero.

ACKNOWLEDGEMENT

This work was supported by NIH grants R01DC004855 and R01DC006435 to S. S. N.

REFERENCES

1. Ossadtchi A, Baillet S, Mosher JC, Thyerlei D, Sutherling W, Leahy RM. Automated interictal spike detection and source localization in magnetoencephalography using independent components analysis and spatio-temporal clustering. *Clinical Neurophysiology* 2004; **115**(3):508–522.
2. Ungan P, Basar E. Comparison of Wiener filtering and selective averaging of evoked potentials. *Electroencephalography and Clinical Neurophysiology* 1976; **40**(5):516–520.
3. Nagarajan S, Attius HT, Sekihara K, Hild II KE. Partitioned factor analysis for interference suppression and source extraction. *International Workshop on Independent Component Analysis and Signal Separation* (*ICA '06*), Charleston, SC. Lecture Notes in Computer Science, vol. 3889. Springer: Berlin, 2006; 189–197.
4. Attias H. A variational Bayesian framework for graphical models. *Advances in Neural Information Processing Systems* (*NIPS '99*). MIT Press: Cambridge, MA, 2000; 209–215.
5. Jackson JE. *A User's Guide to Principal Components*. Wiley: Hoboken, NJ, 2003.

DOI: 10.1002/sim

6. Ziehe A, Muller KR. TDSEP—an efficient algorithm for blind separation using time structure. *International Conference on Artificial Neural Networks* (*ICANN '98*), Skovde, Sweden, September 1998; 675–680.
7. Hyvarinen A. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks* 1999; **10**(3):626–634.
8. MacKay DJC. Bayesian non-linear modeling for the energy prediction competition. *ASHRAE Transactions* 1994; **100**(2):1053–1062.
9. Sahani M, Linden J. Evidence optimization techniques for estimating stimulus–response functions. *Advances in Neural Information Processing Systems* (*NIPS '02*), vol. 15. MIT Press: Cambridge, MA, December 2002; 301–308.
10. MacKay DJC. Bayesian interpolation. *Neural Computation* 1992; **4**(3):415–447.
11. Cover TM, Thomas JA. *Elements of Information Theory*. Wiley: New York, 1991.

DOI: 10.1002/sim