Efficient Bayesian Tracking of Multiple Sources of Neural Activity: Algorithms and Real-Time FPGA Implementation

Lifeng Miao, Jun Jason Zhang
Chaitali Chakrabarti and Antonia Papandreou-Suppappola

***Abstract*—We propose new Bayesian algorithms to automatically track current dipole sources of neural activity in real time. We integrate multiple particle filters to track the dynamic parameters of a known number of dipole sources, resulting in reducing the computational intensity incurred due to the large number of sensors required to observe magnetoencephalography (MEG) or electroencephalography (EEG) measurements. When we also need to estimate the time-varying number of dipole sources, we develop an algorithm based on applying probability hypothesis density filtering (PHDF) for multiple object tracking. The PHDF is implemented using particle filters (PF-PHDF), and it is applied in a closed-loop with MEG/EEG measurements to first estimate the number of sources and then their corresponding amplitude, location and orientation. The PF-PHDF tracking algorithm uses an online, window-based multiple channel decomposition processing approach that reduces the overall processing time and computational complexity. We demonstrate the improved performances of the proposed algorithms by simulating neural activity tracking systems with both synthetic and real data. We map the proposed algorithms onto Xilinx Virtex-5 field-programmable gate array (FPGA) platforms and demonstrate real-time tracking performance. For example, our results showed that the PF-PHDF algorithm can process 100 data samples from three dipoles in only 5.1 ms, when 3 dipole sources are present.**

***Index Terms*—Probability hypothesis density filter, particle filter, multiple particle filters, neural activity, dipole sources, parallel architecture, FPGA implementation.**

I. Introduction

The field of neuroscience provides insight into the mechanism of the brain and nervous system, and it can lead to new treatments of illnesses such as stroke, epilepsy, and Parkinson's disease. New research in this field follows from advances in brain scanning technology, including magnetoencephalography (MEG) and electroencephalography (EEG) [1]–[3]. The human brain consists of a large number of neurons that have a resting state characterized by a cross-membrane voltage difference. When an electromagnetic signal is transferred from one neuron to another, a chemical postsynaptic potential is created that can be modeled as a localized current dipole [4]. When thousands of neighboring neurons are simultaneously in this postsynaptic excitation state, localized current is generated that creates an electromagnetic field outside the skull. The magnetic field can be recorded as an MEG signal using a superconducting quantum interference device; the corresponding electric potential can be recorded as an EEG signal using multiple electrodes placed at different locations on the scalp.

Localizing and tracking localized current dipoles using EEG/MEG measurements can provide very useful information during brain surgery of patients with medically uncontrolled partial seizures [5]. For instance, it can distinguish different kinds of seizures based on the location and orientation of the seizure foci. For Parkinsons disease, neural dipole tracking can identify and locate the exact source of the electrical nerve signals and as a result, improve the performance of deep brain stimulation based treatment [6].

Both EEG and MEG yield high temporal resolution in the order of tens of milliseconds. Achieving high spatial resolution requires accurate solution to the EEG/MEG inverse problem; the inverse problem is the estimation of the localized current dipole model from EEG/MEG measurements. Several methods have been applied to solve the inverse problem, including the recursively applied and projected multiple signal classification (RAP MUSIC) approach [7], [8], spatial filters or beamformers [9], and Bayesian methods [5], [10]–[19]. As Bayesian methods have been extensively used for tracking multiple objects, they are good candidates for the online tracking of current dipole sources [5], [17]. In [11], [12], a Kalman filter was used to solve the EEG inverse problem whereas a particle filter (PF) was applied to estimate the locations of dipole sources in [12]–[14]. In an effort to improve estimation performance, a Rao-Blackwellized PF and beamforming PF were applied in [15], [16]. Note that the number of dipole source parameters to be estimated increases considerably when tracking multiple neural activities. However, this can result in a significant increase in the PF computational complexity as the number of particles required must be increased to maintain certain level of tracking performance.

In this paper, we propose a sequential Monte Carlo neural tracking algorithm that integrates multiple PFs [20] to reduce the computational complexity of a single PF. The algorithm uses separate, parallel and interactive sub-PFs to track each dipole, thus dividing the high-dimensional, multiple-dipole source model into multiple, low-dimensional, single-dipole source model systems [21]. Each sub-PF can then use a largely-reduced number of particles, so that the overall integrated system has a much higher implementation efficiency. We implement the resulting algorithm on a Xilinx Virtex-5 field-programmable gate array (FPGA) platform to demonstrate its applicability to real-time neural activity tracking.

The approaches mentioned thus far typically assume that neural activity can be represented by a fixed and known number of current dipole sources. However, this is not a realistic

Lifeng Miao, Chaitali Chakrabarti, and Antonia Papandreou-Suppappola: School of Electrical, Computer and Energy Engineering at Arizona State University in Tempe, AZ; Jun Jason Zhang: Department of Electrical and Computer Engineering at the University of Denver in Denver, CO.

This work was supported by the National Science Foundation under Grant No. 0830799.

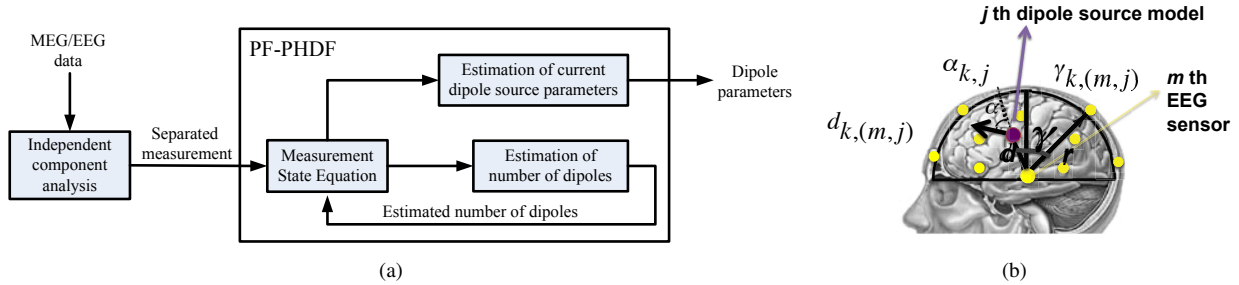


Fig. 1. (a) Block diagram of proposed closed-loop neural dipole source tracking system. (b) Equivalent current dipole model for MEG/EEG localization for the jth dipole source and the mth MEG/EEG sensor. Here, $d = d_{k,m,j}$, $\gamma = \gamma_{k,m,j}$, and $\alpha = \alpha_{k,j}$, as defined in Equation (2).

assumption: neural activity varies with time, so current dipole source models and their parameters should also vary with time. In [5], [18], [19], an approach was used to dynamically estimate the number of dipoles and their parameters at each time step by modeling them as random finite sets (RFS) or random numbers of random objects [22]. In particular, the authors first estimated the number of sources by maximizing the marginal distribution of the current dipole set posterior density and then tracked their locations as the maxima of the RFS first moment or probability hypothesis density. This system was implemented using a PF, and each particle was sampled from a point process, resulting in particles with varying dimensions. The multiple PF tracker algorithm was thus impractical as it used a very large number of particles (in the order to 100,000 particles for five dipoles [18]). In order to avoid the large computational complexity of the PF in their implementation, the authors in [18] overestimated the number of sources and then iteratively used k-means clustering to track the source parameters.

We also propose a new algorithm for estimating both the unknown number of neural dipole sources and their parameters for real EEG/MEG data, with much fewer particles, using the probability hypothesis density filter (PHDF). The PHDF was used in [22] to recursively estimate the number of objects and their parameters in order to overcome the PF bottleneck. However, the use of the PHDF for solving the EEG/MEG inverse problem is more challenging. Specifically, the PHDF requires that each measurement is generated exclusively from a single object. However, each EEG/MEG sensor measurement consists of contributions from all dipole sources. Also, as the PHDF propagation equations are complex, their real time implementation is computationally intensive. Extending our initial work in [23], we propose a computationally inexpensive and feasible algorithm in order to overcome these limitations. In particular, we first reduce the dimension of the EEG/MEG measurements using a threshold-based eigenvalue distilling algorithm; this is needed to facilitate separating the measurements into independent components corresponding to the different dipole sources [24]. We then use the PF implementation of the PHDF (PF-PHDF) [25] to estimate the time-varying number of dipoles at each time step before estimating their unknown parameters. The proposed approach is described in Figure 1(a). It simplifies the dual estimation problem to a known-number of dipole sources estimation problem; it decreases the number of required particles and thus the computational complexity of the overall algorithm. Furthermore, for efficient neural tracking, we use windowing along with parallelization to speed up processing. The algorithm is implemented on a Xilinx Virtex-5 FPGA platform. For a 4-processor architecture, the processing time for one iteration of PHDF was shown to be 48.52 μs, and the processing time for a windowed data segment of 100 samples, obtained using a 1 kHz sampling rate, took only 5.1 ms. We can demonstrate that our EEG/MEG tracking system has enough computing power to perform real-time processing for up to 10 kHz sampling rate.

The rest of the paper is organized as follows. In Section II, we present the neural activity dipole model for the EEG/MEG inverse problem. We develop an algorithm based on the multiple PF to track the parameters of a known number of dipole sources in Section III. In Section IV, we first review the PHDF and its PF implementation. We then present the new algorithm for jointly estimating the unknown number of dipole sources and their parameters using the PF-PHDF. Hardware implementations of both algorithms are provided in Section V. We present numerical simulation results, together with algorithmic and hardware performance results, in Section VI.

II. Neural Activity Dipole Source Model

Analytic solutions for the EEG/MEG source localization problem, based on current dipole source models, can be obtained when the head is assumed to consist of nested concentric spheres of constant conductivity [3], [8], [12]–[14]. Following this model, the primary current $I_k(\mathbf{r})$ at time k can be represented in terms of N_d current dipoles as [26]

$$I_k(\mathbf{r}) = \sum_{j=1}^{N_d} \mathbf{m}_{k,j}\ \delta(\mathbf{r} - \mathbf{r}_{k,j})\,,$$

where $\mathbf{r}_{k,j} = [r_{k,j}^{(x)}\ \ r_{k,j}^{(y)}\ \ r_{k,j}^{(z)}]^{\mathrm{T}}$ and $\mathbf{m}_{k,j} = [m_{k,j}^{(x)}\ \ m_{k,j}^{(y)}\ \ m_{k,j}^{(z)}]^{\mathrm{T}}$ are the three-dimensional (3-D) location and moment vectors, respectively, in Cartesian coordinates, of the jth current dipole, $j = 1, \ldots, N_d$, at time k, $k = 1, \ldots, K$. The moment of the jth dipole is given by $\mathbf{m}_{k,j} = \mathbf{q}_{k,j}\, s_{k,j}$, where $\mathbf{q}_{k,j} = [q_{k,j}^{(x)}\ \ q_{k,j}^{(y)}\ \ q_{k,j}^{(z)}]^{\mathrm{T}}$ and $s_{k,j}$ are the orientation vector and amplitude of the dipole. The EEG/MEG signals acquired

by M sensors can be represented as [3]

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{n}_k = \mathbf{A}_k\,\mathbf{s}_k + \mathbf{n}_k, \tag{1}$$

where $\mathbf{z}_k = [z_{k,1} \quad z_{k,2} \quad \dots \quad z_{k,M}]^{\mathrm{T}}$ and $\mathbf{n}_k = [n_{k,1} \;\; n_{k,2} \;\; \dots \;\; n_{k,M}]^{\mathrm{T}}$ are the $M \times 1$ signal and measurement noise vectors, respectively, from the M sensors, $\mathbf{x}_k = [\mathbf{x}_{k,1}^{\mathrm{T}} \;\; \mathbf{x}_{k,2}^{\mathrm{T}} \;\; \dots \;\; \mathbf{x}_{k,N_d}^{\mathrm{T}}]^{\mathrm{T}}$ is the $N_d \times 1$ dipole source parameters vector, and $\mathbf{x}_{k,j} = [\mathbf{r}_{k,j}^{\mathrm{T}} \;\; \mathbf{q}_{k,j}^{\mathrm{T}} \;\; s_{k,j}]^{\mathrm{T}}$ is the 7-D parameter vector consisting of the 3-D location $\mathbf{r}_{k,j}$, 3-D orientation $\mathbf{q}_{k,j}$ and 1-D amplitude $s_{k,j}$ of the jth dipole source. We assume that the dipole sources are mutually independent of each other and statistically independent of the noise $\mathbf{n}_k$, and that the noise components $n_{k,m}$, $m = 1, \dots, M$, are mutually independent as in [3]. Also in (1), $\mathbf{s}_k = [s_{k,1} \;\; s_{k,2} \;\; \dots \;\; s_{k,N_d}]^{\mathrm{T}}$ is the amplitude vector corresponding to the N_d dipoles and $\mathbf{A}_k$ is the $M \times N_d$ lead-field matrix that depends on the jth dipole location $\mathbf{r}_{k,j}$ and orientation $\mathbf{q}_{k,j}$. Note that EEG/MEG systems have different lead-field matrices. Specifically, the (m,j)th element $a_{k,m,j}$, $j = 1, \dots, N_d$, $m = 1, \dots, M$, of the lead-field matrix $\mathbf{A}_k$ at time step k of an EEG model is given by [2]

$$a_{k,m,j} = \frac{1}{4\pi\sigma}\cos(\theta_{k,j})\Big[\frac{2}{d_{k,m,j}^3}\Big(|\mathbf{r}_{k,j}|\,\cos(\gamma_{k,m,j}) - r\Big) + \big(d_{k,m,j}|\mathbf{r}_{k,j}|\big)^{-1} - \big(r|\mathbf{r}_{k,j}|\big)^{-1}\Big] + \frac{1}{4\pi\sigma}\sin(\theta_{k,j})\cos(\beta_{k,j})\sin(\gamma_{k,m,j})\left[\frac{2\,r}{d_{k,m,j}^3} + \frac{d_{k,m,j} + r}{r\,d_{k,m,j}\,(r - |\mathbf{r}_{k,j}| + d_{k,m,j})}\right]. \tag{2}$$

The corresponding dipole source model is given in Figure 1(b). Here r is the radius of the head model, $d_{k,m,j}$ is the distance between the jth dipole source and the mth sensor, $\gamma_{k,m,j}$ is the angle between the vector pointing to the mth sensor and the vector pointing to the jth dipole location, $\theta_{k,j}$ is the angle between the jth dipole orientation and the vector pointing to the jth dipole location, $\beta_{k,j}$ is the angle between the plane formed by the jth dipole and the origin, and σ is the head tissue conductivity constant. Also, $|\mathbf{r}_{k,j}| = \big[(r_{k,j}^{(x)})^2 + (r_{k,j}^{(y)})^2 + (r_{k,j}^{(z)})^2\big]^{1/2}$ in (2).

For a MEG system, the (m,j)th lead-field matrix element for the mth sensor, with position $\mathbf{r}_m$ and orientation $\mathbf{q}_m$, is given by [2]

$$a_{k,m,j} = \Big[\frac{\mu_0}{4\pi g^2(\mathbf{r}_{k,j}, \mathbf{r}_m)}\,\mathbf{r}_{k,j} \times \big(g(\mathbf{r}_{k,j}, \mathbf{r}_m)\,\mathbf{q}_m - \mathbf{f}^{\mathrm{T}}(\mathbf{r}_{k,j}, \mathbf{r}_m)\mathbf{q}_m\,\mathbf{r}_m\big)\Big]^{\mathrm{T}}\mathbf{q}_{k,j},$$

where $(\mathbf{a} \times \mathbf{b})$ denotes the cross product between vectors $\mathbf{a}$ and $\mathbf{b}$. The scalar $g(\mathbf{r}_{k,j}, \mathbf{r}_m)$ and the vector $\mathbf{f}(\mathbf{r}_{k,j}, \mathbf{r}_m)$ are obtained as

$$g(\mathbf{r}_{k,j}, \mathbf{r}_m) = d_{k,m,j}\,\big(d_{k,m,j}\,|\mathbf{r}_m| + |\mathbf{r}_m|^2 - \mathbf{r}_{k,j}^{\mathrm{T}}\mathbf{r}_m\big)$$

$$\mathbf{f}(\mathbf{r}_{k,j}, \mathbf{r}_m) = \left(\frac{d_{k,m,j}^2}{|\mathbf{r}_m|} + \eta_{k,m,j} + 2\,d_{k,m,j} + 2\,|\mathbf{r}_m|\right)\mathbf{r}_m - \Big(d_{k,m,j} + 2|\mathbf{r}_m| + \eta_{k,m,j}\Big)\,\mathbf{r}_{k,j}$$

where $\eta_{k,m,j} = (\mathbf{r}_{k,j} - \mathbf{r}_m)^{\mathrm{T}}\mathbf{r}_m / d_{k,m,j}$, μ_0 is the permitivity of free space, and $d_{k,m,j}$ is defined in (2).

The EEG/MEG inverse problem is to dynamically estimate the dynamic parameters, $\mathbf{s}_{k,j}$, $\mathbf{r}_{k,j}$ and $\mathbf{q}_{k,j}$, of the jth dipole source at time step k from the EEG/MEG signal $\mathbf{z}_k$ defined in (1).

III. Tracking a Known Number of Dipole Sources

The particle filter (PF) technique [27] has been used to track dipole source parameters [12]–[14]. The state equation for tracking the jth dipole source is given by

$$\mathbf{x}_{k,j} = \mathbf{x}_{k-1,j} + \boldsymbol{\nu}_{k-1}\,, \tag{3}$$

where $\boldsymbol{\nu}_k$ is a modeling error random process that is independent of the measurement noise $\mathbf{n}_k$ in the measurement equation in (1). Using the state-space model in (3) and (1), the PF provides an iterative approach to sequentially compute the posterior probability density function of the state $\mathbf{x}_{k,j}$ at every time step k, conditioned on the EEG/MEG measurements $\mathbf{z}_{1:k}$. Since there are only 7 unknown parameters for the jth dipole, the PF can achieve good tracking performance with a fairly small number of particles. For N_d dipole sources, the state-space model can be written as in (3) and (1) with the state vector $\mathbf{x}_k$ having $7N_d$ dimensions. As the dimension of the dipole source state increases when multiple dipole sources are tracked, the number of particles required for accurate tracking increases substantially. As a result, it becomes computationally impractical to directly use a PF for multiple dipole neural tracking.

To efficiently track multiple dipole sources, we use multiple particle filtering (MPF) [20], [21]. Specifically, we divide the high-dimensional multiple-dipole state space model into multiple, low-dimensional single-dipole state space models that can be solved using PFs with smaller number of particles. We assume that the number of dipoles is constant and known to be N_d. Then, for the MPF formulation, we divide the $7N_d \times 1$ vector $\mathbf{x}_k$ into N_d single-dipole 7×1 vectors $\mathbf{x}_{k,j}$, $j = 1, \dots, N_d$. As before, $\mathbf{x}_{k,j}$ includes the location, orientation and amplitude of the jth dipole source. Then, we use N_d sub-PFs to track the N_d different states. The particle generation and resampling step of each sub-PF are implemented using the sampling importance resampling (SIR) PF [28]. However, the weight update step for the jth sub-PF is modified to [20]

$$w_{k,j}^{(\ell)} \propto w_{k-1,j}^{(\ell)}\; p\Big(\mathbf{z}_k | \mathbf{x}_{k,j}^{(\ell)}, \tilde{\mathbf{x}}_{k,-j}\Big), \quad \ell = 1, \dots, L_j$$

where $\mathbf{x}_{k,j}^{(\ell)}$ and $w_{k,j}^{(\ell)}$ are the ℓth particle and ℓth weight of the jth sub-PF, L_j is the number of particles in the jth sub-PF, $\tilde{\mathbf{x}}_{k,-j} = [\tilde{\mathbf{x}}_{k,1}^{\mathrm{T}} \;\; \dots \;\; \tilde{\mathbf{x}}_{k,j-1}^{\mathrm{T}} \;\; \tilde{\mathbf{x}}_{k,j+1}^{\mathrm{T}} \;\; \dots \;\; \tilde{\mathbf{x}}_{k,N_d}^{\mathrm{T}}]^{\mathrm{T}}$ are the predicted dipole parameters, excluding $\tilde{\mathbf{x}}_{k,j}$, and $\hat{\mathbf{x}}_{k,-j} = [\hat{\mathbf{x}}_{k,1}^{\mathrm{T}} \;\; \dots \;\; \hat{\mathbf{x}}_{k,j-1}^{\mathrm{T}} \;\; \hat{\mathbf{x}}_{k,j+1}^{\mathrm{T}} \;\; \dots \;\; \hat{\mathbf{x}}_{k,N_d}^{\mathrm{T}}]^{\mathrm{T}}$ are the estimated dipole parameters, excluding $\hat{\mathbf{x}}_{k,j}$. Thus, the predicted values of the jth dipole parameters are obtained as $\tilde{\mathbf{x}}_{k,j} = \sum_{\ell=1}^{L_j} w_{k-1,j}^{(\ell)}\,\mathbf{x}_{k,j}^{(\ell)}$. Note that the overall number of required particles can be significantly reduced since we

use the predicted value $\tilde{\mathbf{x}}_{k,-j}$ instead of a set of random particles $\mathbf{x}_{k,-j}^{(\ell)}$ to calculate the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k^{(\ell)}) = p(\mathbf{z}_k|\mathbf{x}_{k,j}^{(\ell)}, \tilde{\mathbf{x}}_{k,-j})$.

When applying the MPF to track neural activity, we make the critical assumption that the number of dipole sources is fixed and known at any given time step. However, in real scenarios, the number of dipoles is unknown and changing over time. As a result, the number of dipoles has to be estimated together with their location, orientation, and amplitude at each time step.

IV. Tracking an Unknown Number of Dipole Sources

A. Problem Formulation

In order to realistically model neural activity, the number of dipole sources N_k is unknown and changing with time, together with the source parameters. Under this scenario, the multiple dipole source state and measurements are modeled as random finite sets (RFS) [22], [29]. Thus, the problem under consideration is to estimate N_k as well as the parameters of the N_k dipole sources at each time step k.

Using the RFS formulation, the multiple dipole source state RFS is given by $\mathbf{X}_k = \{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,N_k}\}$, where $\mathbf{x}_{k,j} = [\mathbf{r}_{k,j}^{\mathrm{T}} \ \mathbf{q}_{k,j}^{\mathrm{T}} \ s_{k,j}]^{\mathrm{T}}$ is the state vector of the jth dipole source in (3). Specifically, the multiple dipole source state RFS $\mathbf{X}_k$ is a finite-set-valued random vector whose elements, $\mathbf{x}_{k,j}$, $j = 1, \ldots, N_k$, are jointly characterized by a *discrete* probability distribution. Similarly, the multiple sensor measurement RFS is given by $\mathbf{Z}_k = \{z_{k,1}, \ldots, z_{k,M_k}\}$, where $z_{k,m}$ is the mth measurement in (1) and M_k is the number of measurements at time step k; the measurements could also include false alarm measurements due to the presence of clutter. Since both the number of sources N_k and the number of measurements M_k can vary randomly in time, it is difficult to estimate both N_k and $\mathbf{X}_k$ at each time step k, given all measurements, up to time k, $\mathbf{Z}_{1:k} = \{\mathbf{Z}_1, \ldots, \mathbf{Z}_k\}$. It is also not known *a priori* which source has generated a given measurement. As dipole sources generate neural activity independently of one another, the PHDF has been shown to provide an efficient approach towards solving this problem [22], [23].

B. Probability Hypothesis Density Filtering

Assuming a state RFS $\mathbf{X}_k$, with $\mathbf{x}_k \in \mathbf{X}_k$, and a measurement RFS $\mathbf{Z}_k$, then integrating the probability hypothesis density or complexity function $\zeta(\mathbf{x}_k|\mathbf{Z}_k)$ on a given region $\mathcal{R}$, provides the expected number of sources present in region $\mathcal{R}$. Also, the locations of the peaks of the density provide estimates of the parameters of the sources in the region $\mathcal{R}$ [22]. To formulate the PHDF for the dipole source estimation problem, we use the method in [22] to first describe an RFS model for the time evolution of the multiple dipole source state and an RFS model for the sensor measurements. Specifically, given the dipole source state $\mathbf{X}_{k-1}$ at time step $k-1$, the dipole source state $\mathbf{X}_k$ is formed by combining: (a) dipole sources still present from the previous time step, $\mathbf{X}_{k|k-1}^{\text{prev}}$; (b) dipole sources that are new at the present time step, $\mathbf{X}_k^{\text{new}}$; and (c) dipole sources spawning from sources from the previous time step, $\mathbf{X}_{k|k-1}^{\text{spn}}$. For the RFS measurement model, we consider the likelihood $p(\mathbf{Z}_k|\mathbf{x}_k)$ for each possible dipole source state $\mathbf{x}_k \in \mathbf{X}_k$.

The PHDF assumes that the predicted multiple dipole source posterior density $p(\mathbf{x}_k|\mathbf{Z}_{1:k-1})$ can be completely characterized by its first moment that can be represented by the multiple dipole source RFS probability hypothesis density or intensity function $\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1})$ [22]. Thus, given the posterior intensity $\zeta(\mathbf{x}_{k-1}|\mathbf{Z}_{1:k-1})$ at time step $(k-1)$, the predicted intensity $\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1})$ can be obtained as [22]

$$\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1}) = \zeta(\mathbf{x}_k^{\text{new}}|\mathbf{Z}_{1:k}) + \int \Big[\Pr_{k|k-1}(\tilde{\mathbf{x}}_{k-1})\, p(\mathbf{x}_k|\tilde{\mathbf{x}}_{k-1}) + \zeta(\mathbf{x}_k^{\text{spn}}|\mathbf{Z}_{1:k-1})\Big]\, \zeta\left(\tilde{\mathbf{x}}_{k-1}|\mathbf{Z}_{1:k-1}\right) d\tilde{\mathbf{x}}_{k-1} \tag{4}$$

where $\mathbf{x}_k^{\text{new}} \in \mathbf{X}_k^{\text{new}}$, $\mathbf{x}_k^{\text{spn}} \in \mathbf{X}_{k|k-1}^{\text{spn}}$, and $\Pr_{k|k-1}(\mathbf{x}_{k-1})$ is the probability that a dipole source that was present at time step $(k-1)$ will still be present at time step k. The posterior intensity is given by

$$\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k}) = \left(1 - \Pr_k^{\text{det}}(\mathbf{x}_k)\right) \zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1}) + \sum_{\mathbf{z}_k \in \mathbf{Z}_{1:k}} \frac{\Pr_k^{\text{det}}(\mathbf{x}_k)\ p(\mathbf{Z}_k|\mathbf{x}_k)\ \zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1})}{\zeta(\mathbf{Z}_k^{\text{clt}}) + \int \Pr_k^{\text{det}}(\tilde{\mathbf{x}}_k)\, p(\mathbf{Z}_k|\tilde{\mathbf{x}}_k)\, \zeta(\tilde{\mathbf{x}}_k|\mathbf{Z}_{1:k-1})\, d\tilde{\mathbf{x}}_k} \tag{5}$$

where $\Pr_k^{\text{det}}(\mathbf{x}_k)$ is the probability of detecting a dipole source at time step k. As the received multiple measurement RFS $\mathbf{Z}_k$ can also include clutter, $\mathbf{Z}_k^{\text{clt}}$, due to possible false alarms, $\zeta(\mathbf{Z}_k^{\text{clt}})$ is used to denote the clutter intensity; it is assumed that the clutter RFS is independent of the dipole source measurement RFS and that the dipole source measurement RFS elements are mutually independent.

C. PHDF Implementation Using Particle Filtering

The PHDF prediction and update equations in (5) and (6) involve multiple integrals that do not have computationally tractable closed form expressions, even for the simple linear Gaussian dynamic case. One possible implementation of the PHDF is using particle filtering [25]. Specifically, assuming that the intensity function $\zeta(\mathbf{x}_{k-1}|\mathbf{Z}_{1:k-1})$ at time step $(k-1)$ can be approximated by a set of N_{k-1} particles $\mathbf{x}_{k-1}^{(\ell)}$ and their corresponding weights $w_{k-1}^{(\ell)}$, $\ell = 1, \ldots, N_{k-1}$,

$$\zeta(\mathbf{x}_{k-1}|\mathbf{Z}_{1:k-1}) = \sum_{\ell=1}^{N_{k-1}} w_{k-1}^{(\ell)} \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^{(\ell)}). \tag{6}$$

where $\delta(\cdot)$ is the Dirac delta function, then substituting (6) into (5), we obtain

$$\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1}) = \zeta(\mathbf{x}_k^{\text{new}}|\mathbf{Z}_{1:k}) + \sum_{\ell=1}^{N_{k-1}} w_{k-1}^{(\ell)} \Big[\Pr_{k|k-1}(\mathbf{x}_{k-1}^{(\ell)})\ p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(\ell)}) + \zeta(\mathbf{x}_k^{\text{spn}}|\mathbf{Z}_{1:k-1})\Big]. \tag{7}$$

A particle approximation of $\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1})$ can be obtained by applying importance sampling to each term in (7). Specifically, the samples $\mathbf{x}_k^{(\ell)}$, $\ell = 1, \ldots, N_{k-1}$, are drawn from the importance density function $q_k(\mathbf{x}_k^{(\ell)}|\mathbf{x}_{k-1}^{(\ell)}, \mathbf{Z}_k) = p(\mathbf{x}_k^{(\ell)}|\mathbf{x}_{k-1}^{(\ell)})$, and the samples $\mathbf{x}_k^{(\ell)}$, $\ell = N_{k-1} + 1, \ldots, (N_{k-1} + N_q)$ are drawn from the

importance intensity function $\xi_k(\mathbf{x}_k^{(\ell)}|\mathbf{Z}_k) = \zeta(\mathbf{x}_k^{\text{new}}|\mathbf{Z}_{1:k})$, where N_q is the additional number of particles for the new dipole sources at time step k. Then, the prior intensity function $\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1})$ can be approximated by particles $\mathbf{x}_k^{(\ell)}$ and their weights $w_{k|k-1}^{(\ell)}$, $\ell = 1, 2, \ldots, (N_{k-1} + N_q)$ as

$$\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k-1}) = \sum_{\ell=1}^{N_{k-1}+N_q} w_{k|k-1}^{(\ell)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(\ell)}) \tag{8}$$

where

$$w_{k|k-1}^{(\ell)} = \begin{cases} w_{k-1}^{(\ell)} \left(\Pr_{k|k-1}(\mathbf{x}_{k-1}^{(\ell)}) + \dfrac{\zeta(\mathbf{x}_k^{\text{spn}}|\mathbf{Z}_{1:k-1})}{p(\mathbf{x}_k^{(\ell)}|\mathbf{x}_{k-1}^{(\ell)})} \right), \\ \qquad \ell = 1, \ldots, N_{k-1} \\ 1/N_q \,, \ell = N_{k-1}+1, \ldots, (N_{k-1}+N_q) \end{cases} \tag{9}$$

Substituting (8) into (6), we obtain the particle approximation of the posterior intensity function as

$$\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k}) = \sum_{\ell=1}^{N_{k-1}+N_q} w_k^{(\ell)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(\ell)}), \tag{10}$$

where

$$w_k^{(\ell)} = w_{k|k-1}^{(\ell)} \left(1 - \Pr_k^{\text{det}}(\mathbf{x}_k^{(\ell)}) + \sum_{\mathbf{Z}_k \in \mathbf{Z}_{1:k}} \frac{\Pr_k^{\text{det}}(\mathbf{x}_k^{(\ell)})\, p(\mathbf{Z}_k|\mathbf{x}_k^{(\ell)})}{\zeta(\mathbf{Z}_k^{\text{clt}}) + C_k(\mathbf{Z}_k)}\right)$$

and

$$C_k(\mathbf{Z}_k) = \sum_{\ell=1}^{N_{k-1}+N_q} w_{k|k-1}^{(\ell)} \Pr_k^{\text{det}}(\mathbf{x}_k^{(\ell)})\, p(\mathbf{Z}_k|\mathbf{x}_k^{(\ell)}) \,.$$

Based on (5) and (6), a particle approximation of the posterior intensity $\zeta(\mathbf{x}_k|\mathbf{Z}_{1:k})$ can be obtained at time step k from its particle approximation at the previous time step $k-1$. Using this particle approximation, both the number of sources in a given region and the sources' parameters can be estimated. The PF-PHDF method can solve the EEG/MEG inverse problem even when the number of dipoles is unknown. It assumes that the mth measurement $z_{k,m}$ is generated from a single-target or false alarm. However, as the EEG/MEG measurements $\mathbf{z}_k$ in (1) are due to contributions from multiple dipoles, we next consider an approach that first decomposes the EEG/MEG data before estimating the unknown number of dipoles using the PF-PHDF.

D. Proposed Algorithm for Tracking an Unknown Number of Dipole Sources

The PF-PHDF is robust and computationally inexpensive compared to existing multiple target tracking techniques, and it has been successfully used in radar and sonar tracking [30], [31]. However, there are significant challenges in applying the PF-PHDF to solve the EEG/MEG inverse problem. Whereas each measurement in radar is generated from a single target, EEG/MEG sensor measurements are due to contributions from all dipole sources; thus, the observed data must be decomposed before it can be used by the PF-PHDF. Also challenging is the large dimensionality of the data due to the large number of sensors used to collect it.

To address these challenges, the proposed dipole estimation PF-PHDF algorithm: 1) reduces the size of the measurement matrix and simplifies computation using efficient pre-whitening of the EEG/MEG measurement; 2) reduces the number of pre-whitened components using independent component analysis (ICA); and 3) applies the PF-PHDF to estimate the number of dipoles and their parameters. These three stages are described next in more detail.

1) Efficient Pre-whitening of Measurements: The PF-PHDF algorithm requires each measurement vector to be generated by a single object. However, for the EEG/MEG dipole model, the EEG/MEG measurement $\mathbf{z}_k = \mathbf{A}_k \mathbf{s}_k$ in (1) is generated by multiple neural sources. Thus, we first need to decompose the EEG/MEG measurements into individual components, with each component corresponding to an individual neural source. The decomposition can be simplified by first performing a preliminary pre-whitening of the EEG/MEG data $\mathbf{z}_k = [z_{k,1} \;\; z_{k,2} \;\; \ldots \;\; z_{k,M}]^{\text{T}}$. Specifically, the data is linearly transformed to $\mathbf{v}_k = \mathbf{U}\mathbf{z}_k = \mathbf{U}\mathbf{A}_k\mathbf{s}_k$ whose elements are mutually uncorrelated with unit variance, $E[\mathbf{v}_k\mathbf{v}_k^{\text{T}}] = \mathbf{I}$ [24], where $\mathbf{I}$ is the identity matrix and $\mathbf{U}$ is a linear transformation matrix. An example of such a linear transformation can be obtained using eigenvalue decomposition $\mathbf{U} = \Lambda^{-1/2}\mathbf{\Psi}^{\text{T}}$ of the measurement covariance matrix $\mathbf{\Sigma}_{\text{cov}} = E[\mathbf{z}_k\mathbf{z}_k^{\text{T}}]$, where Λ is a diagonal matrix whose elements are the eigenvalues of $\mathbf{\Sigma}_{\text{cov}}$ and $\mathbf{\Psi}$ is a matrix whose columns consist of the corresponding eigenvectors. In particular, assuming M EEG/MEG measurement sensors, the eigenvalues of the $M \times M$ covariance matrix $\mathbf{\Sigma}_{\text{cov}}$ need to be obtained. Although there are many algorithms to calculate eigenvectors from a covariance matrix, when the size of $M > 5$, most of them can hardly be mapped into an efficient VLSI architecture. In most EEG/MEG systems, the number of sensors is between $M = 30$ and $M = 150$, and so the challenge is to find an efficient approach to calculate the eigenvalues of a large matrix. In our previous work [23], we used channel decomposition to solve this problem. The channels were divided into several groups and ICA was applied to each group. However, this approach resulted in estimation performance loss, and, despite the fact that the computational intensity was reduced, this step was still the bottleneck of our implementation.

We consider a different approach, where we assume that only a small set of patches of the human brain are activated at a time [3], [10], [17], [32]. Under this assumption, the number of dipoles N_d is much smaller than the number of sensors M, $N_d << M$. Thus instead of calculating all the eigenvalues and eigenvectors, we only need to find the leading eigenvalues and eigenvectors corresponding to the N_d active dipoles. We propose an efficient leading eigenvector and eigenvalue calculation approach based on the eigenvector distilling algorithm [33]. The steps of the proposed method are described in Algorithm 1. In order to obtain the ith largest eigenvector $\mathbf{\Psi}_i$, we first initialize $\mathbf{\Psi}_i^{(0)}$ randomly with norm 1 and then update it using r iterations to obtain $\mathbf{\Psi}_i$. Its corresponding eigenvalue λ_i is obtained and compared to the threshold V_{thr}. If $\lambda_i > V_{\text{thr}}$, this eigenvector is retained and the procedure continues to calculate the next one. Note that the leading eigenvectors are calculated one by one, in reducing

Algorithm 1 Threshold-based leading-only eigenvalue-eigenvector distilling algorithm [33]

Choose eigenvalue threshold V_{thr} and number of iterations Ω; compute covariance matrix $\mathbf{\Sigma}_{\text{cov}}$
Calculate the leading eigenvector matrix $\mathbf{\Psi}$ and eigenvalues λ
Set $i=1$ and $\mathbf{\Psi}_i^{(0)}=0$
Initialize eigenvector $\mathbf{\Psi}_i^{(0)}$ randomly with norm 1
for $j=1:\Omega$ **do**
$\mathbf{\Psi}_i^{(j)}=\dfrac{1}{||\mathbf{\Phi}_i^{(j)}||}\,\mathbf{\Phi}_i^{(j)}$ where $\Phi_i^{(j)}=\mathbf{\Sigma}_{\text{cov}}\mathbf{\Psi}_i^{(j-1)} - \sum_{l=0}^{i-1}\left(\left[\mathbf{\Sigma}_{\text{cov}}\mathbf{\Psi}_i^{(j-1)}\right]^{\mathrm{T}}\mathbf{\Psi}_l\right)\mathbf{\Sigma}_{\text{cov}}\mathbf{\Psi}_i^{(j-1)}$
end for
Set $\mathbf{\Psi}_i=\mathbf{\Psi}_i^{(\Omega)}$ and $\lambda_i=\dfrac{1}{\mathbf{\Psi}_i^{\mathrm{T}}\mathbf{\Psi}_i}\left(\mathbf{\Sigma}_{\text{cov}}\mathbf{\Psi}_i\right)^{\mathrm{T}}\mathbf{\Psi}_i$
if $\lambda_i > V_{\text{thr}}$ **then**
Store λ_i and $\mathbf{\Psi}_i$ and set $i=i+1$
else
Stop
end if

order of dominance. As the number of dipoles is unknown, the number of eigenvalues and eigenvectors to be distilled is unknown as well. So we use a threshold to pick the eigenvalues to be computed. We will analyze the influence of this threshold on algorithm performance in Section VI.

2) Component Analysis of Pre-Whitened Measurements: After pre-processing, the new data vector $\mathbf{v}_k$ has reduced dimensionality and reduced noise power. Since distinct neural sources are mutually independent [3], the independent component analysis (ICA) technique can be used to decompose the pre-processed EEG/MEG data. Here, we assume that in a short time window the location and orientation of dipole sources are fixed. As a result, the lead-field matrix $\mathbf{A}$ doesn't change with time in this period. The window length is a critical parameter which will be analyzed in Section IV-E. For the EEG/MEG data in one window, we use FastICA, a free MATLAB software package that implements a fast fixed-point ICA algorithm [24], to obtain a de-mixing matrix, $\mathbf{W}=\mathbf{UA}$. Using this matrix, we can obtain J estimated independent sources as $\hat{\mathbf{s}}_k=\mathbf{W}^{\mathrm{T}}\mathbf{v}_k=[\hat{s}_{k,1}\ \ldots\ \hat{s}_{k,J}]^{\mathrm{T}}$. The estimated mixing matrix $\hat{\mathbf{A}}=[\hat{\mathbf{a}}_1\ \ldots\ \hat{\mathbf{a}}_J]$, where $\hat{\mathbf{a}}_j$ is the jth column of $\hat{\mathbf{A}}$, can be obtained as $\hat{\mathbf{A}}=\mathbf{U}^{-1}\mathbf{W}$. Using $\hat{\mathbf{a}}_j$, the EEG/MEG signal that contributed from the jth individual source is given by

$$\hat{\mathbf{z}}_{k,j}=\hat{\mathbf{a}}_j\hat{s}_{k,j}\,,\quad j=1,\ldots,J\,. \tag{11}$$

After this stage, the decomposed unmixed EEG/MEG measurement vectors are given by $\hat{\mathbf{z}}_k=[\hat{z}_{k,1}\ \ldots\ \hat{z}_{k,J}]$. Thus, each new measurement $\hat{z}_{k,j}$ is assumed to have originated either from a single dipole source or an artifact (non-brain) activity (false alarm).

3) Multi-dipole Estimation Using PF-PHDF: The decomposed pre-whitened EEG/MEG measurement RFS $\hat{\mathbf{Z}}_k=\{\hat{z}_{k,1},\ \hat{z}_{k,2},\ \ldots\ \hat{z}_{k,J}\}$ can now be directly used as the input to the PF-PHDF algorithm to estimate the number of dipole sources and their unknown parameters. The corresponding state-space RFS model for the EEG/MEG source estimation problem is given by

$$\mathbf{X}_k = \mathbf{X}_{k-1}+\boldsymbol{\nu}_{k-1} \tag{12}$$
$$\hat{\mathbf{Z}}_k = h(\mathbf{X}_k)+\mathbf{n}_k\,. \tag{13}$$

The specific PF-PHDF algorithm steps are described next; Figure 2 demonstrates the PF-PHDF framework for the EEG/MEG dipole source estimation problem.

Step 1-Initialization: At time step $k=0$, the particles $\tilde{\mathbf{x}}_0^{(\ell)}$, $\ell=1,\ldots,N_0$, are drawn from the initial intensity function $\zeta(\mathbf{x}_0)$, where $N_0=N\,N_{d_0}$, N_{d_0} is the initial number of dipoles and N is the number of particles used for each dipole, and the corresponding weights are obtained as $w_0^{(\ell)}=N_{d_0}/N_0$, $\ell=1,\ldots,N_0$.

Step 2-Prediction: At time step k, particles $\tilde{\mathbf{x}}_k^{(\ell)}$, $\ell=1,\ldots,N_{k-1},\ldots,N_{k-1}+N_q$ are sampled and, assuming no dipole source spawning, the corresponding weights are evaluated using (9)

$$\tilde{w}_{k|k-1}^{(\ell)}=\begin{cases}\tilde{w}_{k-1}^{(\ell)}\Pr_{k|k-1}(\tilde{\mathbf{x}}_{k-1}^{(\ell)})\,,\ell=1,\ldots,N_{k-1}\\ 1/N_q\,,\ell=N_{k-1}+1,\ldots,N_{k-1}+N_q\end{cases} \tag{14}$$

Step 3-Updating: Using (11), the weights are updated to

$$\tilde{w}_k^{(\ell)}=\tilde{w}_{k|k-1}^{(\ell)}\left(1-\Pr_k^{\text{det}}(\tilde{\mathbf{x}}_k^{(\ell)})+\sum_{j=1}^{J}\frac{\Pr_k^{\text{det}}(\tilde{\mathbf{x}}_k^{(\ell)})\,p(\hat{\mathbf{z}}_{k,j}|\tilde{\mathbf{x}}_k^{(\ell)})}{\zeta(\mathbf{Z}_k^{\text{clt}})+C_k(\hat{\mathbf{z}}_{k,j})}\right)$$

where

$$C_k(\hat{\mathbf{z}}_{k,j})=\sum_{\ell=1}^{N_{k-1}+N_q}\tilde{w}_{k|k-1}^{(\ell)}\Pr_k^{\text{det}}(\tilde{\mathbf{x}}_k^{(\ell)})\,p(\hat{\mathbf{z}}_{k,j}|\tilde{\mathbf{x}}_k^{(\ell)})\,.$$

Step 4-Resampling: The number of dipole sources is estimated as $\hat{N}_{d_k}=\sum_{\ell=1}^{N_{k-1}+N_q}\tilde{w}_k^{(\ell)}$; the particles $\tilde{x}_k^{(\ell)}$ are resampled and their corresponding weights $\tilde{w}_k^{(\ell)}$, $\ell=1,\ldots,N_{k-1}+N_q$ are computed to obtain, respectively, $\tilde{\mathbf{x}}_k^{(\ell)}$ and $\tilde{w}_k^{(\ell)}=\hat{N}_{d_k}/N_k$, $\ell=1,\ldots,N_k$. Here, $N_k=N\,\text{round}(\hat{N}_{d_k})$, where $\text{round}(N)$ denotes the nearest integer to N.

Step 5-Estimating dipole state: The resampled particles are clustered and the state parameters are estimated. The clustering is performed in 3-D using the k-means clustering algorithm.

Using the PF-PHDF, the number of particles changes over time and is proportional to the number of dipoles, i.e., at time k, $N_k\propto\hat{N}_{d_k}$. Unlike standard PF, there is a summation among sub-measurements $\hat{\mathbf{z}}_{k,j}$, $j=1,\ldots,J$, corresponding to individual sources, when updating the weights in Step 3. After Step 3, the posterior intensity $\zeta(\mathbf{x}_k|\hat{\mathbf{Z}}_k)$ at time k is approximated using particles $\tilde{\mathbf{x}}_k^{(\ell)}$ and weights $\tilde{w}_k^{(\ell)}$, $\ell=1,\ldots,N_{k-1}+N_q$, that contain all available dipole source information. For example, the number of dipoles can be obtained by integrating the posterior intensity, which is equal to the summation of the weights; and the dipole source parameters can be estimated from the peaks of the intensity. In the resampling step, the new weights are not normalized to 1, but sum to $\hat{N}_{d_k}$, the estimated number of dipole sources.

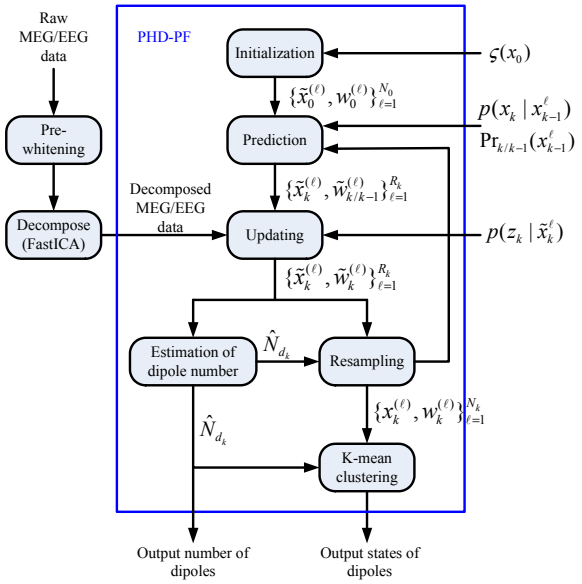


Fig. 2. Framework of the proposed neural activity tracking system based on PF-PHDF.

E. Data Windowing for On-Line Processing

Most methods process EEG/MEG data off-line, after all of the data has been collected [34]. Our proposed algorithm can be used to process EEG/MEG data on-line by windowing the data and then applying the algorithm on each windowed data segment in a pipelined fashion. Figure 3shows how we pre-whiten and run FastICA on the data from Window 1, while obtaining data in Window 2. The length of the window, L_w, is a critical parameter because the processing time and tracking accuracy both depend on it. If L_w is small, the computations take less time. Unfortunately, the data in a short window do not provide an accurate statistical representation of neural activity and result in a large estimation error (see Table V). We also require the window length to be larger than the execution time of pre-whitening and FastICA, as shown in Figure 3.

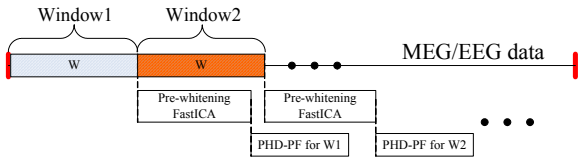


Fig. 3. Pipelined window processing of MEG/EEG data.

F. Computational Complexity Analysis

Even with the computational complexity reduction due to the use of eigenvalue distilling, the proposed estimation system can still be computationally intensive. Table I lists the number of operations when the system uses $M = 32$ sensors, $N_d = 3$ dipole sources, and 1,000 particles per-dipole. The pre-whitening step includes the computation of the measurement covariance matrix, which is computationally intensive. However, the PF-PHDF computational complexity is the largest, creating a system bottleneck. Table I also lists the computation time when running Matlab implementation of the different algorithms on a multi-core PC (Intel Core Quad CPU Q6600 @ 2.4GHz). We see that the processing time for PF-PHDF is more than 25 seconds which is too large to support the real-time processing of an EEG/MEG system. In the next section, we describe a hardware implementation of the PF-PHDF which makes use of pipelining and parallel processing to reduce the computation time.

V. Hardware Implementation

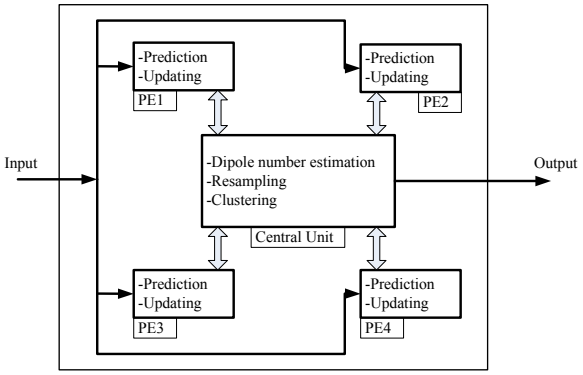


Fig. 4. Parallel PF-PHDF architecture with four processing elements: PE1, PE2, PE3, and PE4.

In this section, we describe an efficient multi-processor architecture for the PF-PHDF. Pre-whitening and FastICA can be implemented using the architecture in [33], [35] and are not described here.

The high level block diagram of the proposed PF-PHDF architecture is shown in Figure 4. It consists of P processing elements (PE) and one central unit (CU) connected by a global bus. Local processing steps, such as initialization (Step 1), prediction (Step 2) and part of updating (Step 3) are conducted in each PE. Global processing steps, such as computing normalization factors C_k, estimating the number of dipoles (Step 3), resampling (Step 4) and clustering (Step 5), are executed in the CU. Each PE communicates with the CU, but there is no communication among PEs.

The operation flow for the PF-PHDF is shown in Figure 5. Each PE processes N_k/P particles, where N_k is the number of particles at time step k. First, the particles $\mathbf{x}_{k-1}^{(\ell)}$ are processed in the prediction unit to generate the new particles $\tilde{\mathbf{x}}_k^{(\ell)}$ by sampling the transition density $\Pr_{k|k-1}(\mathbf{x}_{k-1}^{(i)\ell})$. The predicted weights $\tilde{w}_{k|k-1}^{(\ell)}$ are calculated in the prediction unit. Next, the likelihoods $p(\hat{\mathbf{z}}_{k,j}|\tilde{\mathbf{x}}_k^{(\ell)})$ are calculated for each individual measurement $\hat{\mathbf{z}}_{k,j}$, for each particle $\tilde{\mathbf{x}}_k^{(\ell)}$. Since the calculations of $p(\hat{\mathbf{z}}_{k,j}|\tilde{\mathbf{x}}_k^{(\ell)})$, $j = 1, \ldots, J$ are independent, they are implemented in parallel. After obtaining the likelihoods for all particles, $p(\hat{\mathbf{z}}_{k,j}|\tilde{\mathbf{x}}_k^{(\ell)})$, $\ell = 1, \ldots, N_k/P$, the sum of the likelihoods $\sum_\ell p(\hat{\mathbf{z}}_{k,j}|\tilde{\mathbf{x}}_k^{(\ell)})$ is sent to the CU by each PE. Then, the central unit calculates the normalization factor, $C_k(\hat{\mathbf{z}}_{k,j}) = \sum_\rho \sum_\ell p(\hat{\mathbf{z}}_{k,j}|\tilde{\mathbf{x}}_k^{(\ell)})^\rho$ and sends it back to each PE; each PE then computes the final weights $\tilde{w}_k^{(\ell)}$ based on Equation (14) in the final weight unit.

TABLE I
NUMBER OF OPERATIONS AND PC PROCESSING TIME OF THE PROPOSED DIPOLE SOURCE ESTIMATION SYSTEM IN FIGURE 2

	Number of Operations					
Block	Additions	Multiplications	Divisions	Square roots	Exponentials	PC Processing time
Pre-whitening	107,680	107,680	11	10	0	0.32 s
FastICA	3,070	8,070	10	10	0	0.24 s
PF-PHDF	560,000	320,000	16,000	9,600	3,200	25.63 s

TABLE II
HARDWARE OPERATORS FOR EACH BLOCK IN FIGURE 5.

Unit	Block	+	×	/	$\sqrt{}$	exp
PEs	Sampling	6	0	0	0	0
	Likelihood	88	93	1	2	1
	Group mean	52	0	1	0	0
	Final weight	6	1	1	0	0
CU	Global	6	0	0	0	0
	Normal	9	0	0	0	0
	Resampling	2	3	1	0	0
	Clustering	6	3	1	1	0

Since the resampling step is operated in the CU, the weights of all the particles have to be transferred from the PEs to the CU, which results in a large communication overhead. In order to reduce this overhead, we employ the grouping method in [21] and add the group-and-mean unit in each PE. The main idea is as follows. The particles $\tilde{\mathbf{x}}_k^{(\ell)}$ and their corresponding weights $\tilde{w}_k^{(\ell)}$ are divided into G groups based on the range of the particles; the averages of each group, $x_{k,\text{mean}}^{(g)}$, $g = 1, \ldots, G$, are used as the new particles, and only the average weights $w_{k,\text{mean}}^{(g)}$, $g = 1, \ldots, G$, are transmitted to the CU to be used as input to the resampling step. These particles are stored in the mean particle memory (MPMEM) for future use. Before the resampling step, we estimate the number of dipole sources by summing the final weights, $\hat{N}_{d_k} = \sum_i \hat{w}_k^{(\ell)}$. During the resampling step, the replication index ρ is calculated at the CU based on the average weights. After the resampling step, the group averaged particles are read from MPMEM and sent to the prediction unit for the computation at the next iteration. These particles are also sent to the clustering unit at the CU, and are used to estimate the dipole parameters.

This procedure significantly reduces the communication between the PEs and the CU. Figure 5 depicts eight data transactions (DT): DT 1 corresponds to the local extremum, $\mathbf{x}_{\min}$ and $\mathbf{x}_{\max}$, transmitted from the PEs to the CU; DT 2 is the summation of the likelihoods, $\sum_\ell p(\hat{\mathbf{z}}_{k,j}|\tilde{\mathbf{x}}_k^{(\ell)})$, from the PEs to the CU; DT 3 corresponds to the normalization factors, $C_k(\hat{\mathbf{z}}_{k,j})$, from the CU to the PEs; DT 4 corresponds to the updated weights, $\tilde{w}_k^{(\ell)}$, from the PEs to the CU; DT 5 corresponds to the global extremum, $\mathbf{x}_{\text{Min}}$ and $\mathbf{x}_{\text{Max}}$, from the CU to the PEs; DT 6 corresponds to the weights with the average values, $w_{k,\text{mean}}^{(g)}$, from the PEs to the CU; DT 7 corresponds to the replication index, ρ, from the CU to the PEs; and DT 8 corresponds to the updated particles, $\mathbf{x}_k^{(\ell)}$, from the PEs to the CU. The hardware resource for each block in Figure 5 is shown in Table II. Note that the likelihood computation unit is the most demanding part in terms of resources.

TABLE III
COMPARISON OF RMSE AND PC PROCESSING TIME FOR TWO-DIPOLE MODEL

Approach	SPF	Multiple PF	
Number of particles	20,000	10,000	20,000
Dipole 1 Location RMSE	7.5 mm	7.0 mm	6.2 mm
Dipole 2 Location RMSE	7.3 mm	6.9 mm	6.0 mm
PC Processing time	583 s	375 s	759 s

VI. SIMULATION AND FPGA IMPLEMENTATION RESULTS

A. Simulation Results for Tracking Known Number of Dipoles

1) Synthetic Data Results: The known number of dipoles tracking results are first demonstrated with the MPF approach using synthetic data. The data was created by inserting current dipoles into the sphere head model and calculating the resulting magnetic field using Equation (1) with Gaussian noise. For the N_d=2 multi-dipole case, the tracking results are shown in Figures 6(a)-6(d). The MPF used two sub-PFs with 5,000 particles each, and the conventional SIR PF (SPF) used 20,000 particles. The average root mean-squared error (RMSE) comparison for 100 Monte Carlo simulations is summarized in Table III. Although both trackers provide reasonable estimates of the MEG source locations, the MPF tracker needs significantly fewer number of particles. When the MPF tracker uses the same number of particles as the SPF, then the MPF results in improved RMSE performance. In addition, we compared the timing performance of Matlab implementation of the proposed MPF and SPF on a multi-core PC (Intel Core Quad CPU Q6600 @ 2.4GHz). For comparable performance, the processing time of SPF for 100 samples was 583 seconds compared to 375 seconds required for MPF. Thus, for two dipole sources tracking problem, MPF takes significantly shorter time than SPF.

2) Real Data Results: The performance of the new MPF approach is also compared with the SPF using real data from a language study experiment [36]. Figure 7 compares the SPF and MPF tracking performance with that of a dipole fitting beamformer technique that uses a spatially adaptive filter to estimate the amount of activity at any given location in the brain [36], [37]. We used 10,000 particles for the SPF and 5,000 particles for each of the sub-PFs of the MPF. Note that for the real data, we do not have the accurate location of the source dipoles, so we cannot show RMSE results.

B. Numerical Simulation Results for Tracking Unknown Number of Dipoles

1) Simulation Set Up: In order to demonstrate the tracking performance of the proposed PF-PHDF, first we use synthetic data from a previous study in [18] with three dipoles localized at $V1$ (1.11, 5.34, 4.98), $V5_R$ (4.36, 3.68, 4.44)

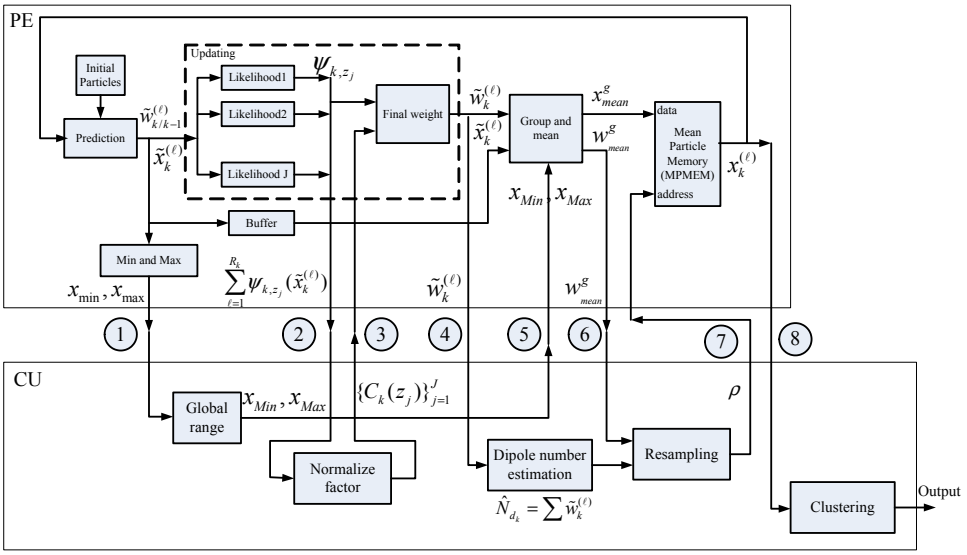
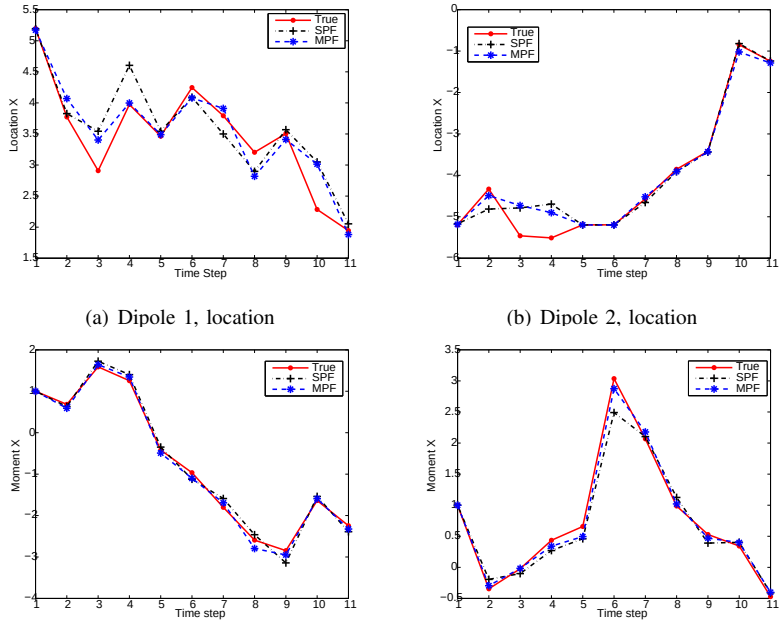


Fig. 5. Operation flow and data transactions between PE p and the central unit (CU) at each time step.



(a) Dipole 1, location (b) Dipole 2, location

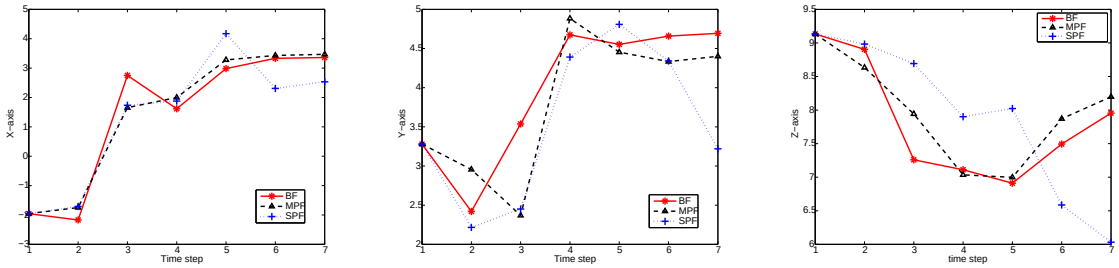
(c) Dipole 1, moment (d) Dipole 2, moment

Fig. 6. Comparison between the true (red) and estimated, using SIR PF (black) and MPF (blue), x Cartesian coordinate of 2 dipoles (a) Dipole 1 location; (b) Dipole 2 location; (c) Dipole 1 moment; and (d) Dipole 2 moment.

and $V5_L$ (3.37, 4.85, 4.81) in an example where the brain volume hemisphere has a radius of 10 cm. The measurement noise is Gaussian with 0 mean and variance $\sigma = 10^{-5}$. For this simulation, we used uniformly distributed particles with $N = 1,000$ particles for each dipole. For a three dipole system, the maximum number of particles is $3N + N_q = 3,200$, which is much less than the 100,000 particles used in [18]. With $\mathbf{x}_k = [\mathbf{r}_k^{\mathrm{T}} \ \mathbf{q}_k^{\mathrm{T}} \ s_k]^{\mathrm{T}}$, the dipole state transition model in Equation (12) is a random walk model with Gaussian transition kernel $p(\mathbf{r}_k|\mathbf{r}_{k-1}) = \mathcal{N}(\mathbf{r}_{k-1}, \sigma_r)$ and $p(\mathbf{q}_k|\mathbf{q}_{k-1}) = \mathcal{N}(\mathbf{q}_{k-1}, \sigma_q)$, with $\sigma_r = 1$ cm and $\sigma_q = 2$ nA. Each existing dipole has a probability of survival $\Pr_{k|k-1}(\mathbf{x}_{k-1}) = 0.8$ and a probability of detection $\Pr_k^{det}(\mathbf{x}_k) = 0.95$.

2) Eigenvalue Threshold Selection: In Section IV-D, we discussed that eigenvalue decomposition can be used to reduce the dimension of EEG data by choosing several leading eigenvalues instead of all the eigenvalues. Eigenvalue selection method is crucial since it determines the number of independent components and the reconstruction error of ICA. A threshold based method is used to select the leading eigenvalues. Figure 8 shows the amplitudes of all the eigenvalues of the covariance matrix.

In order to find the optimal threshold for leading eigenvalue selection, we use the root mean square error (RMSE) of the



(a) x-coordinate (b) y-coordinate (c) z-coordinate

Fig. 7. Tracking results comparison for beamformer method (BF), SIR PF (SPF), and MPF for the dipole location using real MEG data.

TABLE IV
RMSE OF RECONSTRUCTED EEG DATA FOR DIFFERENT THRESHOLD VALUES.

Threshold	500	150	130	100	50
Dipole1	2.68 μV	2.91 μV	4.97 μV	7.45 μV	NaN
Dipole2	2.67 μV	2.87 μV	5.08 μV	8.35 μV	NaN
Dipole3	2.53 μV	2.79 μV	4.16 μV	7.95 μV	NaN

reconstructed EEG data as the cost function. The RMSE is defined as the difference of estimated individual EEG data $\hat{\mathbf{z}}_{k,j}$ obtained using Equation (11) and the true EEG data $\mathbf{z}_{k,j}$

$$\text{RMSE}_j^{\text{rec}} = \left[\frac{1}{K}\sum_{k=1}^{K}\frac{1}{M}(\mathbf{z}_{k,j}-\hat{\mathbf{z}}_{k,j})^{\text{T}}(\mathbf{z}_{k,j}-\hat{\mathbf{z}}_{k,j})\right]^{1/2}$$

where K is the number of time steps and M is the number of sensors. Table IV shows the RMSE of the reconstructed EEG data for different threshold values. We can see that as the threshold value decreases, the RMSE increases and that there is a significant RMSE degradation when the threshold is smaller than 150. In addition, with thresholds smaller than 100, we can hardly distinguish dipole signals with noise which causes the FastICA algorithm to fail. Based on these results, we choose a threshold value of 500 for the rest of the simulation.

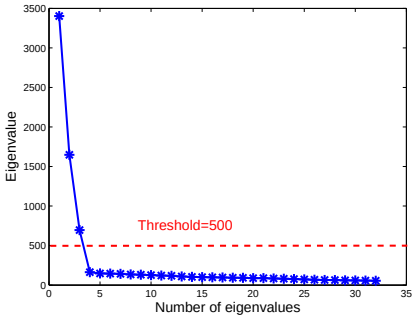


Fig. 8. Eigenvalues of EEG covariance matrix for simulated data.

3) Window Length Selection: The choice of window length L_w is crucial as it greatly impacts the estimation results. Table V shows the reconstruction RMSE with respect to different L_w. We can see from the table that if the window length is too long (much longer than the duration of dipole), the independent component analysis cannot capture the changing information of the dipole. As a result, the reconstruction error is large. If the window length is too small, the samples in the window cannot statistically represent the whole data, which also leads to larger reconstruction error. Based on these results, we choose a window length $L_w = 100$ samples with sampling rate 1 kHz.

TABLE V
RMSE OF RECONSTRUCTED EEG DATA FOR DIFFERENT WINDOW LENGTHS.

Window length	50	100	250	500
Dipole 1	2.78 μV	2.16 μV	2.33 μV	4.22 μV
Dipole 2	2.68 μV	2.21 μV	2.42 μV	4.51 μV
Dipole 3	2.60 μV	2.13 μV	2.44 μV	4.13 μV

4) Estimation results: The eigenvalue selection threshold was chosen as 500 and the window length as $L_w = 100$. We used 1,000 particles for each existing dipole and 200 particles for the new dipoles. The estimation results for the amplitudes of three dipoles are shown in Figure 9. The RMSE for the dipole current amplitude is 2.07 nA. Figure 10 shows the estimation results for the dipoles 3-D location; the position RMSE is 6.93 mm. The orientation estimation results are demonstrated in Figure 11. We compared the performance of the proposed tracking algorithm with those in [13], [15], [18], as shown in Table VI. Compared to [13] which has the best tracking performance, the proposed PF-PHDF based system has a tracking error of 1.5 mm. However, the PF-PHDF only uses 3,200 particles (compared to 100,000) and also does not require knowledge of the number of dipoles. Compared to [18] which also tracks an unknown number of dipoles, the tracking error is only 0.7 mm but the number of particles used is only 3,200 compared to 100,000. This is because we need preprocessing to separate the EEG/MEG signals into independent components such that each component corresponded to one dipole source. Thus all the particles were for only one dipole source and the number of particles was reduced to a few thousands. In summary, the proposed system obtains comparable estimation results with significantly reduced computational complexity for tracking unknown number of dipoles.

C. Real EEG Data

We applied the proposed PF-PHDF algorithm to a real EEG data set from a visual experiment described in [38]. This experiment tracks brain activity when the subject reacts to green squares appearing on the screen. We first used the EEG

TABLE VI
COMPARISON OF NEURAL ACTIVITY TRACKING FOR SYNTHETIC DATA.

Approach	Number of particles	Number of dipoles	Knowledge of di
PF [13]	100,000	4	Know
RB-PF [15]	50,000	2	Know
D-PF [18]	100,000	3	Unknov
PF-PHDF	3,200	3	Unknov

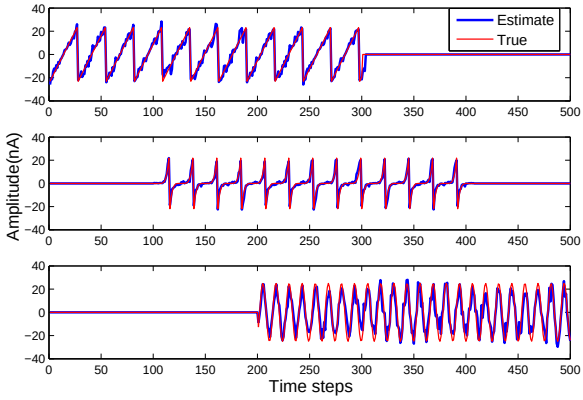


Fig. 9. Amplitude tracking result of three dipoles for synthetic EEG data.

data from one trial as training data and found the threshold for eigenvalue selection. The result in Figure 12 is similar to the one in Figure 8. As there is a significant drop when the eigenvalues are smaller than 500, we choose the threshold as 500; this is the same value used for the synthetic data simulation. The sampling rate of this experiment is 128 Hz. Since the frequency of the stimuli is once every 3 s and the subject pushes the button about 1 s after the stimuli, we assume that the states and number of dipoles will change every 1 s. As a result, we choose the window length to be $L_w = 128$ samples.

We show the estimation results for one trial spanning 3 s. The EEG data for 5 (out of 32) channels is shown in Figure 13. Figure 14 shows the estimation result of the 3-D location for the dipoles. Since the true location of the dipoles is unknown, we use the estimation result of a standard dipole fitting procedure [39] as the ground truth. The RMSE of the proposed PF-PHDF for the 3-D dipole location calculated with respect to [39] is 2.1 mm. From Figure 14, we can see that during the first second, there is only one dipole which means that in this period most of the brain is in the inactive state. During the next two seconds, the number of dipoles increased to three. At 0.78 s, the subject received a stimulus from the screen and made a response by pushing the button at 1.13 s. As a result, certain neurons located in the posterior of the brain were activated which led to the increase in number of dipoles.

D. Hardware Implementation Evaluation

The proposed PF-PHDF hardware architecture is implemented using Verilog HDL and synthesized on the Xilinx Virtex-5 device. The design is verified using Modelsim.

Resource Utilization: Table VII summarizes the architecture resource utilization for the 4 PE parallel architecture for the PF-PHDF. Since the total number of particles is 3,200, each

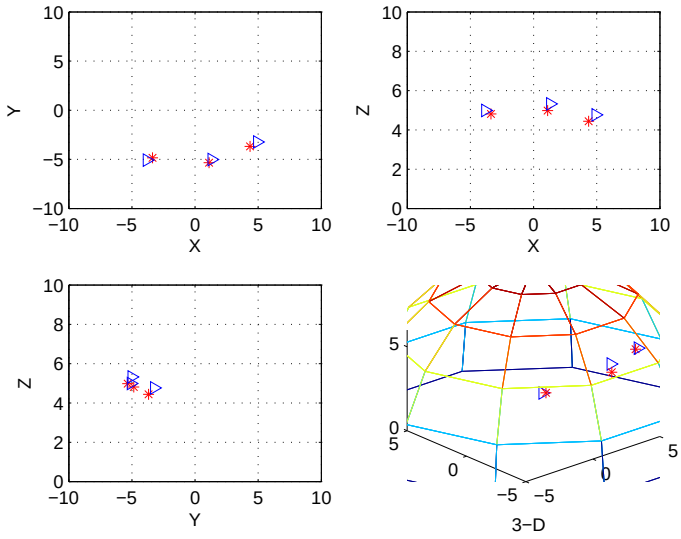


Fig. 10. Estimated 3-D locations of dipoles for simulated EEG data.

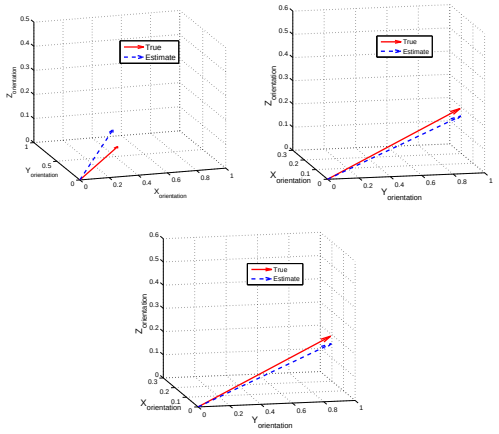


Fig. 11. Orientation estimation results of three dipoles for simulated EEG data.

PE processes 800 particles. For the likelihood calculation in the updating step, the exponential functions are implemented using CORDIC units, and the rest of the units are implemented using DSP cores.

TABLE VII
RESOURCE UTILIZATION FOR PF-PHDF ON XILINX XC5VSX240K

Unit	Occupied slices	Slice Reg.	Slice LUTs	Block Ram	DSP48Es
PF-PHDF	14,291 (39%)	43,637 (30%)	42,383 (29%)	134 (26%)	283 (27%)

Execution Time: Figure 15 shows the timing performance for one iteration of the proposed PF-PHDF; the actual number of cycles is given in Table VIII. One iteration takes N_{total}=$(N_s + N_g + N_{gm} + N_c)$=4,852 cycles. We choose a system clock rate of 100 MHz and so the processing time for one iteration is only $T_{\text{total}} = N_{\text{total}}\, T_{\text{clk}} = 48.52\mu$s. Based on the pre-whitening and FastICA implementation in [33], [35], the execution time for preprocessing is about 265 μs. Thus, for a window with

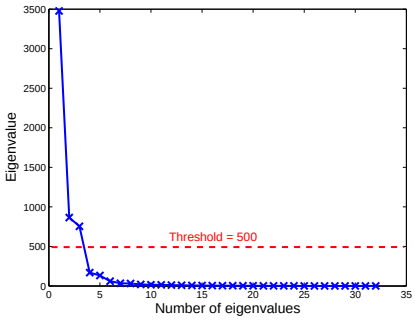


Fig. 12. Eigenvalues of EEG covariance matrix for real data.

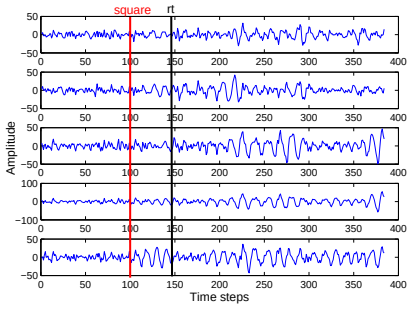


Fig. 13. One typical cycle of the EEG data (5 channels and 3 seconds).

100 samples spanning 1 s, the total processing time is $265 + (48.52 \times 100) = 5,117$ μs or 5.1 ms.

TABLE VIII
EXECUTION CYCLES FOR EACH BLOCK

Unit	N_s	N_l	N_g	N_n	N_w	N_{gm}	N_k	N_r	N_c
Cycles	804	827	5	4	818	832	4	53	3211

E. Scalability

For the proposed tracking system, the number of particles used for each dipole N is a critical parameter as it impacts the estimation accuracy and processing time. Figure 16 shows the tracking performance in terms of RMSE for dipole location and the processing time for PF-PHDF with respect to N. Here we choose the number of PE $P = 4$. From Figure 16, we can see that as N increases, the RMSE decreases and the processing time increases. However, when N is greater than 1000, there is no significant improvement in the RMSE but the processing time increases rapidly. A good tradeoff between RMSE and processing time is obtained at $N = 1000$.

For the real data case corresponding to the visual experiment of a person tracking green squares, the maximum number of dipoles was small (less than 5). From the FPGA timing results, we project that if the maximum number of dipoles is 3, the proposed system can perform real-time processing at sampling rates of up to 10 kHz for window length $L_w = 100$ samples. However, for epilepsy patients, the number of dipoles during seizures can be greater than 10 [32]. Figure 17 shows the timing performance of the proposed system with respect to the maximum number of dipoles for 1,000 particles per dipole and $P = 4$. From Figure 17, we can see that as the maximum

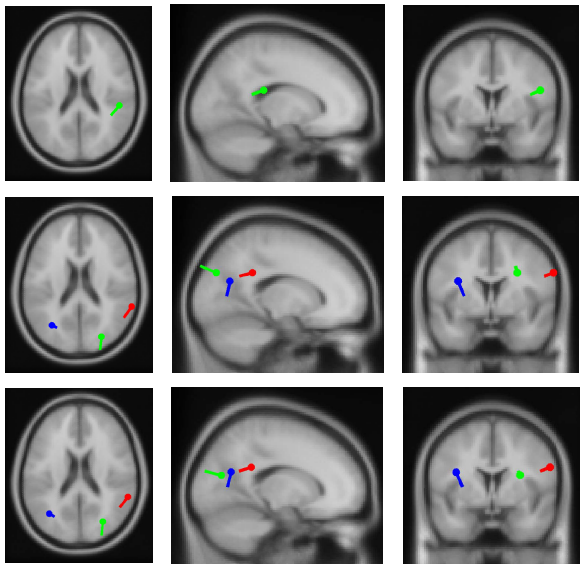


Fig. 14. 3-D dipole location estimation result for the EEG data. (a) dipole location for the 1st second (top three); (b) dipole location for the 2nd second (middle three); (c) dipole location for the 3rd second (bottom three).

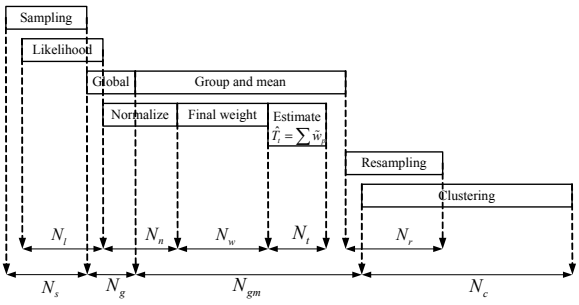


Fig. 15. Execution time breakdown for one PF-PHDF iteration.

number of dipoles increases, the processing time for a window grows, as expected. Even when the number of dipoles is as large as 15, our system can still support real-time tracking with sampling rate of about 1.5 kHz.

VII. CONCLUSION

In this paper, we described the application of the PF-PHDF for tracking an unknown number of neural dipoles, and we demonstrated its performance using numerical simulations for both synthetic and real EEG data. The proposed method achieves good performance in terms of RMSE using significantly fewer number of particles compared to existing approaches. We also presented a window based processing method and a threshold based eigenvalue distilling algorithm to enable real-time processing. The proposed method was implemented on the Xilinx Virtex-5 FPGA platform. The processing time for a window with 100 samples using 3,200 particles for a system with 3 dipoles was shown to be only 5.1 ms. Thus, this implementation is capable of real-time processing of systems with larger number of dipoles and/or larger number of samples per second.

REFERENCES

[1] M. Hämäläinen, R. Hari, R. J. Ilmoniemi, J. Knuuttila, and O. V. Lounasmaa, "Magnetoencephalography–theory, instrumentation and ap-

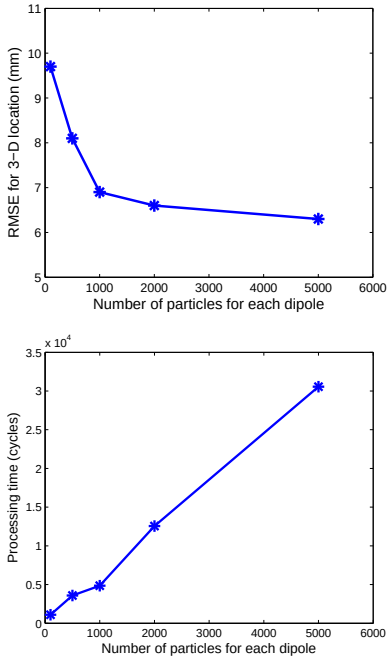


Fig. 16. RSME for location and processing time for PF-PHDF with respect to N.

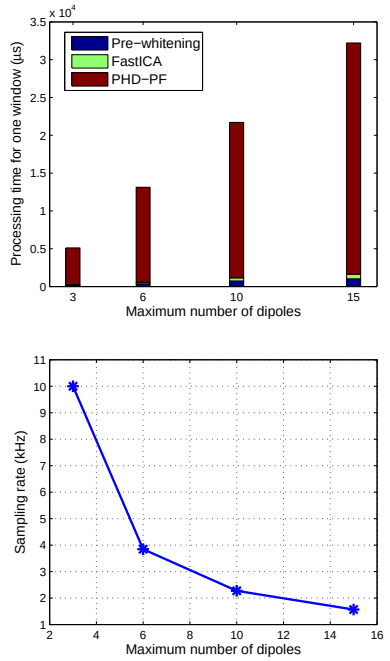


Fig. 17. Scalability of the proposed system with respect to maximum number of dipoles: Effect on (a) processing time and (b) maximum sampling rate for real-time processing.

plications to nonivasive studies of the working human brain," *Reviews of Modern Physics*, vol. 65, pp. 413–497, 1993.

[2] J. C. Mosher, R. M. Leahy, and P. S. Lewis, "EEG and MEG: Forward solutions for inverse methods," *IEEE Transactions in Biomedical Engineering*, vol. 46, pp. 245–259, 1999.

[3] S. Baillet, J. C. Mosher, and R. M. Leahy, "Electromagnetic brain mapping," *IEEE Signal Processing Magazine*, vol. 18, pp. 14–30, 2001.

[4] E. Somersalo, "The inverse problem of magnetoencephalography: Source localization and the shape of a ball," *SIAM News*, vol. 40, no. 2, March 2007.

[5] A. Pascarella and A. Sorrentino, "Statistical approaches to the inverse problem," in *Magnetoencephalography*, E. W. Pang, Ed. InTech, November 2011.

[6] A. Benabid, "Deep brain stimulation for parkinsons disease," *Current Opinion in Neurobiology*, vol. 12, pp. 767–791, 2003.

[7] J. C. Mosher, P. S. Lewis, and R. M. Leahy, "Multiple dipole modeling and localization from spatio-temporal MEG data," *IEEE Transactions on Biomedical Engineering*, vol. 39, pp. 541–557, June 1992.

[8] J. C. Mosher and R. M. Leahy, "Source localization using recursively applied and projected (RAP) MUSIC," *IEEE Transactions on Signal Processing*, vol. 47, pp. 332–340, 1999.

[9] B. D. Van Veen and K. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Magazine*, vol. 5, pp. 4–24, 1988.

[10] E. Somersalo, A. Voutilainen, and J. P. Kaipio, "Non-stationary magnetoencephalography by Bayesian filtering of dipole models," *Inverse Problems*, vol. 19, pp. 1047–1063, 2003.

[11] A. Galka, O. Yamashita, T. Ozaki, R. Biscay, and P. Valde, "A solution to the dynamical inverse problem of EEG generation using spatiotemporal Kalman filtering," *Inverse Problems*, pp. 435–453, 2004.

[12] J. Antelis and J.Minguez, "Dynamic solution to the EEG source localization problem using Kalman filters and particle filters," in *International Conference of the IEEE Engineering in Medicine and Biology Society*, 2009, pp. 77–80.

[13] A. Sorrentino, L. Parkkonen, and M. Piana, "Particle filters: A new method for reconstructing multiple current dipoles from MEG data," in *International Congress Series*, vol. 1300, 2007, pp. 173–176.

[14] H. R. Mohseni, E. L. Wilding, and S. Sanei, "Sequential Monte Carlo techniques for EEG dipole placing and tracking," in *Sensor Array and Multichannel Signal Processing Workshop*, July 2008, pp. 95–98.

[15] C. Campi, A. Pascarella, A. Sorrentino, and M. Piana, "A Rao-Blackwellized particle filter for magnetoencephalography," *Inverse Problems and Imaging*, vol. 24: 025023, 2008.

[16] H. R. Mohseni, F. Ghaderi, E. L. Wilding, and S. Sanei, "A beamforming particle filter for EEG dipole source localization," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 337–340.

[17] A. Pascarella, A. Sorrentino, C. Campi, and M. Piana, "Particle filtering, beamforming and multiple signal classification for the analysis of MEG time series: A comparison of algorithms," *Inverse Problems and Imaging*, vol. 4, pp. 169–190, February 2010.

[18] A. Sorrentino, L. Parkkonen, A. Pascarella, C. Campi, and M. Piana, "Dynamical MEG source modeling with multi-target Bayesian filtering," *Human Brain Mapping*, vol. 30, pp. 1911–1921, June 2009.

[19] C. Campi, A. Pascarella, A. Sorrentino, and M. Piana, "Highly Automated Dipole EStimation (HADES)," *Computational Intelligence and Neuroscience*, 2011, article ID 982185.

[20] P. M. Djuric, T. Lu, and M. F. Bugallo, "Multiple particle filering," in *IEEE International Confernece on Acoustics, Speech, and Signal Processing*, June 2007, pp. 1181–1184.

[21] L. Miao, J. J. Zhang, C. Chakrabarti, and A. Papandreou-Suppappola, "Multiple sensor sequential tracking of neural activity with FPGA implementation," in *Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA*, November 2010, pp. 369–373.

[22] R. Mahler, "Multi-target Bayes filtering via first-order multi-target moments," *IEEE Transactions on Aerospace and Electronic System*, vol. 39, pp. 1152–1178, 2003.

[23] L. Miao, J. J. Zhang, C. Chakrabarti, A. Papandreou-Suppappola, and N. Kovvali, "Real-time closed-loop tracking of an unknown number of neural sources using probability hypothesis density particle filtering," in *IEEE Workshop on Signal Processing Systems*, October 2011, pp. 367–372.

[24] A. Hyvärinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," *Neural Computation*, vol. 9, pp. 1483–1492, 1997.

[25] B. N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo implementation of the PHD filter for multi-target tracking," in *International Conference of Information Fusion*, vol. 2, 2003, pp. 792–799.

[26] K. Uutela, M. Hämäläinen, and R. Salmelin, "Global optimization in the localization of neuromagnetic sources," *IEEE Transactions on Biomedical Engineering*, vol. 45, pp. 716–723, 1998.

[27] N. J. Gordon, D. J. Salmon, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," in *IEE Proceedings F: Radar and Signal Processing*, vol. 140, 1992, pp. 107–113.

[28] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, pp. 174–188, 2002.

[29] D. J. Daley and D. Vere-Jones, *An introduction to the theory of point processes, Volume II: General theory and Structure*, 2nd ed. Springer-Verlag, 1997.
[30] M. Tobias and A. D. Lanterman, "Probability hypothesis density-based multi-target tracking with bistatic range and doppler observations," *IEEE Radar, Sonar and Navigation*, vol. 152, pp. 195–205, 2005.
[31] D. E. Clark and J. Bell, "Bayesian multiple target tracking in forward scan sonar images using the PHD filter." *IEEE Radar, Sonar and Navigation*, vol. 152, pp. 327–334, 2005.
[32] K. Shindo, A. Ikeda, T. Musha, K. Terada, H. Fukuyama, W. Taki, J. Kimura, and H. Shibasaki, "Clinical usefulness of the dipole tracing method for localizing interictal spikes in partial epilepsy," *Epilepsia*, vol. 39, pp. 371–379, 1998.
[33] T. Chen, W. Liu, and L. Chen, "VLSI architecture of leading eigenvector generation for on-chip principal component analysis spike sorting system," in *International Conference of the IEEE Engineering in Medicine and Biology Society*, 2008, pp. 3192–3195.
[34] J. Cao, N. Murata, S. Amari, A. Cichocki, and T. Takeda, "Independent component analysis for unaveraged single-trial MEG data decomposition and single-dipole source localization," *Neurocomputing*, vol. 49, no. 1-4, pp. 255–277, 2002.
[35] S. Fujio, H. Shiomi, and Y. Okamura, "Acceleration of FPGA-based ICA processor for real-time processing," in *International Symposium on Antennas and Propagation*, July 2010, pp. 1–4.
[36] "FieldTrip Matlab software toolbox for MEG and EEG analysis," http://fieldtrip.fcdonders.nl.
[37] J. Gross, J. Kujala, M. Hamalainen, L. Timmermann, A. Schnitzler, and R. Salmelin, "Dynamic imaging of coherent sources: Studying neural interactions in the human brain," *National Academy of Sciences*, vol. 98, pp. 694–699, January 2001.
[38] S. Makeig, M. Westerfield, T. P. Jung, S. Enghoff, J. Townsend, E. Courchesne, and T. J. Sejnowski, "Dynamic brain sources of visual evoked responses," *Science*, vol. 295, pp. 690–694, 2002.
[39] "EEG lab: Open source Matlab toolbox for EEG analysis," http://sccn.ucsd.edu/eeglab/.