Full Brain Blood-Oxygen-Level-Dependent Signal Parameter Estimation Using Particle Filters

Micah Chambers

***Abstract*—Traditional methods of analyzing FMRI images use a linear combination of just a few static regressors. This work demonstrates an alternative approach using a physiologically inspired nonlinear model. By using a particle filter to optimize the model parameters, the computation time is kept below a minute per voxel without requiring a linearization of the noise in the state variables. The activation results show regions similar to those found in SPM; however, there are some notable regions not detected by SPM. Though the parameters selected by the particle filter based approach are more than sufficient to predict the BOLD response, more model constraints are needed to uniquely identify a single set of parameters. This ill-posed nature explains the large discrepancies found in other research that attempted to characterize the model parameters. For this reason the final distribution of parameters is more medically relevant than a single estimate. Because the output of the particle filter is a full posterior probability, the reliance on the mean to estimate parameters is unnecessary. This work presents not just a viable alternative to the traditional method of detecting activation, but an extensible technique of estimating the joint probability distribution function of the BOLD parameters.**

***Index Terms*—BOLD Response, FMRI, Nonlinear Systems, Particle Filter, Bayesian Statistics, System Identification**

I. Introduction

TRADITIONAL methods of analyzing Functional Magnetic Resonance Imaging (FMRI) time series perform regression using a linear combination of static explanatory variables. However, the static nature of the General Linear Model (GLM) limits its potential use. It is well known that different Hemodynamic Response Functions are necessary for different regions of the brain to prevent excessive false negatives [1]. Additionally quite a few studies have reported spatially varying nonlinearities [2], [3]. Besides not allowing HRF differences between patients, there is no reasonable way to incorporate other forms of physiological data. Combined FMRI CBF or CBV imaging methods are improving, as seen in Chen et al. [4]. These techniques could shed light on neural activation by providing extra measurements, yet a physiologically reasonable model is necessary to incorporate this extra data. Activation detection methods also don't have the ability to identify pathologies based on state variables or parameters. For example, decreased compliance of blood vessels could indicate, or even cause, a neurological condition that is not easily seen in other imaging modalities.

It is well known that the changes in deoxy-hemoglobin content is the primary driver of short term changes in MR signal for FMRI imaging techniques [5]–[8]. For this reason, the nonlinear state equations that change Deoxy-Hb (DHb) content have been heavily studied, and are well-characterized. Attempts at learning the parameters of the BOLD model have also been actively studied but significantly less successful. The most wide-spread method of calculating the parameters, from two papers by Friston et al., are part of the SPM toolkit; though this algorithm is rarely used as well [9], [10]. This method depends on a quadratic convolution (two term Volterra Kernel) based estimation of the BOLD output to obtain partials with respect to parameters. Problematically, quality of the Volterra estimate is not well known, so it is difficult to quantify error. This method also doesn't account for noise in the parameters or the state variables.

These shortcomings have led to several other separate attempts. In particular Riera et al. linearized the noise and thus performed regression between the model and steps of the BOLD output [11]. This aproach had the benefit of a Jacobian, but at the same time removes all DC signal. For this reason, the type of activation limits the algorithms' ability to learn the model. In Vakorin et al., a combination of Genetic Algorithms and Simulated Annealing were used to estimate not only the parameters, but the true stimuli driving the BOLD signal [12]. This addresses the inherent uncertainty of exactly where and when stimuli actually get applied. Unfortunately this algorithm was extremely slow; taking hours or days per single time course. In Johnston et al. alternating estimates of $P(X_t|\Theta, Y_t)$ and $P(\Theta|Y_t)$ were calculated, to maximize $P(X_t, \Theta|Y_t)$. $P(X_t|\Theta, Y_t)$ was calculated using a particle filter [13]. This however was quite slow, and the results were inconsistent with other similar works. Murray used a particle filter, similar to [13], but held the model parameters constant; thus all error in the output was treated as the result of noise in the underlying parameters [14]. Murray's results showed that differences in BOLD output cannot be well explained by error in the state variables. Hu et al. used an unscented kalman filter (UKF) to estimate parameters, and filtered parameter sets inconsistent with the output [15]. Furthermore, Birn et al. and Yacoub et al. showed that many of the characteristics of the BOLD output depend on the region of interest [3], [16]. For this reason, a single Hemodynamic Response Function is not likely to accurately estimate activation across the full brain.

With the exception of Hu et al., the works described thus far all worked to estimate a single set of parameters that could explain the output [15]. Yet Deneux et al showed that the very similar outputs can be achieved with very different parameters [17]. This explains the inconsistencies in parameters across the different studies. Thus, estimating parameters using a least squares framework is not likely to be fruitful; and though the posterior distributions will vary spatially, that difference

M. Chambers is with the Department of Electrical and Computer Engineering, Virginia Tech University, Blacksburg, VA, 24060 USA e-mail: (micahc@vt.edu

Manuscript received April 19, 2005; revised January 11, 2007.

will not be discernible from a single parameter estimate. The problem with using an Unscented Kalman Filter to approximate the distributions is that with a nonlinear, dissipative system, significant non-Gaussian effects will be present. In tests, significant non-Gaussian effects began to appear within a second of state integration. With FMRI repetition times being well above 1 second, a Gaussian approximation will result in significant error. The particle filter, which can be thought of an extension of UKF, uses non-parametric distributions rather than a Gaussian. For this reason the particle filter, described in subsection II-B, is used in this work.

II. Methods

A. Bold Model

In the past fifteen years, a steady stream of studies have built on the original Blood Oxygen Level Dependent (BOLD) signal derivation first described by Ogawa et al. [7]. The seminal work by Buxton et al. attempted to explain the time evolution of the BOLD signal using a windkessel model to describe the local changes in Deoxygenated Hemoglobin content [5]. Incremental improvements were made to this model until Friston et al. brought all the changes together into a single complete set of equations [9]. And while there have been numerous adaptations in the model, many of them summarized by Deneux et al., even the basic versions have less bias error than the empirically driven *Canonical Hemodynamic Model* [1], [17]. On the other hand BOLD signal models have numbers of parameters ranging from seven [11] to 50 [18] for a signal as short as 100 samples long. This number of parameters presents a significant risk of being underdetermined and having high computation cost. In this work, only the simplest physiologically inspired model will be used (with 7 parameters), and steps will be taken to make the most of computation time.

It is well known that the two types of hemoglobin act as a contrast agents in EPI imaging [5]–[7], however the connection between Deoxyhemoglobin/Oxygenated Hemoglobin and neural activity is non-trivial. Intuitively, increased metabolism will increase Deoxyhemoglobin, however blood vessels are quick to compensate by increasing local blood flow. Increased inflow, accomplished by loosening capillary beds, precedes increased outflow, driving increased blood storage. Since the local MR signal depends on the ratio of Deoxyhemoglobin to Oxygenated Hemoglobin, increased blood volume affects this ratio if metabolism doesn't exactly match the increased inflow of oxygenated blood. This was the impetus for the ground breaking balloon model [5] and windkessel model [19]. These works derive, from first principals, the changes in deoxyhemoglobin ratio and volume of capillaries given an inflow waveform. These were the first two attempts to quantitatively account for the shape of the BOLD signal as a consequence of the lag between the cerebral blood volume (CBV) and the inward cerebral blood flow (CBF).

Although Buxton et al. demonstrated that a well chosen flow waveform could explain most features of the BOLD signal, it stopped short of proposing a realistic waveform for the CBF and CMRO2 [5]. Friston et al. [9] gave a reasonable and simple expression for CBF input based on a flow inducing signal and in the same work proposed a simple method of estimating metabolic rate: as a linear function of the inward blood flow. By combining these equations with the balloon model from Buxton et al., it is possible to predict the BOLD signal directly from a stimulus time course:

$$\dot{s} = \epsilon u(t) - \frac{s}{\tau_s} - \frac{f-1}{\tau_f} \quad (1)$$

$$\dot{f} = s \quad (2)$$

$$\dot{v} = \frac{1}{\tau_0}(f - v^{\alpha}) \quad (3)$$

$$\dot{q} = \frac{1}{\tau_0}\left(\frac{f(1-(1-E_0)^f)}{E_0} - \frac{q}{v^{1-1/\alpha}}\right) \quad (4)$$

where s is a flow inducing signal, f is the input blood flow (CBF), v is normalized cerebral blood volume (CBV), and q is the normalized local deoxyhemoglobin. The parameters controlling blood flow are ϵ, which is a neuronal efficiency term, $u(t)$ which is the stimulus, and τ_f, τ_s which are time constants. The parameters for the evolution of blood volume are E_0 which the resting metabolic rate and α which is Grubb's parameter controlling the balloon model. τ_0 is a single time constant controlling the speed of v and q.

This completed balloon model was assembled and analyzed by Riera et al. [11]. Obata refined the readout equation of the BOLD signal based on the deoxyhemoglobin content (q) and local blood volume (v), resulting in the final BOLD equation [8].

$$\begin{aligned} y &= V_0((k_1+k_2)(1-q) - (k_2+k_3)(1-v)) \quad (5)\\ k_1 &= 4.3 \times \nu_0 \times E_0 \times TE = 2.8\\ K_2 &= \epsilon_0 \times r_0 \times E_0 \times TE = .57\\ k_3 &= \epsilon_0 - 1 = .43 \end{aligned}$$

Where $\nu_0 = 40.3s^{-1}$ is the frequency offset in Hz for fully deoxygenated blood (at 1.5T), $r_0 = 25s^{-1}$ is the slope relating change in relaxation rate with change in blood oxygenation, and $\epsilon_0 = 1.43$ is the ratio of signal MR from intravascular to extravascular regions at rest. Although, these constants change with experiment (TE, ν_0, r_0), patient, and brain region (E_0, r_0), often the estimated values by Obata et al. are taken as the constants $a_1 = k_1 + k_2 = 3.4$, and $a_2 = k_2 + k_3 = 1$ in studies using 1.5 Tesla scanners [8]. Additional compartments and parameters have been added since Riera et al., however Deneux et al. showed that only the viscoelastic effects of Buxton et al. were necessary to model the primary elements of the BOLD response [11], [17], [20]. Despite this, only the basic balloon model of Buxton et al. is used in this work, although future works may benefit from constraining the parameters as described by Deneux et al. [17], [20].

B. Particle Filter

Particle filters, a type of Sequential Monte Carlo (SMC) method, are a powerful method for estimating the posterior probability distribution of parameters given a time series of measurements. Unlike Markov Chain Monte Carlo (MCMC) estimation, particle filters are designed for time-varying random variables. Unlike the UFK, sample points are kept and

continuously propagated forward, rather than being used to calculate the first two moments. Thus, the posterior distribution is stored as an empircal distribution made up of sample points. Particle filters are preferred when the model is nonlinear, and non-Gaussian. This section based primarily on Arulampalam et al. [21] and Thrun et al. [22].

The idea of the particle filter is to build an empirical distribution out of a large number of parameter sets, called particles. Each particle contains all the parameters and states needed to propagate the model forward. The particle filter begins with a wide distribution (called the Prior Distribution) of possible particles and then, as measurements come in, weights particles based on the quality of their output estimates. Thus parameter sets that tend to give good estimations of the measurements get weighted higher than parameter sets that give poor estimates. Although the reliance on a prior distribution could be problematic, when the system being modeled has physical meaning, establishing reasonable ranges for parameters may be quite easy. Optimizing the prior distribution can be more difficult, unless the system has been extensively studied.

1) Assumptions: Suppose a set or stream of measurements at discrete times are given, $\{Y_k, k = 1, 2, 3, ...K\}$, where K may be infinite. Because k is a discrete time, let t_k define the continuous time of k. Suppose also that there is a hidden set of state variables, $X(t)$ that drives the value of $Y(t)$. Throughout this section with $X_k = X(t_k)$. The goal of the particle filter is to estimate the distribution of the true parameters Θ that dictates the movement of $X(t)$. The model also permits random motion in $X(t)$, so the particle filter also estimates the distribution of $X(t)$. The only difference between the members of parameter vector Θ and those of $X(t)$ is that the memebers of Θ have no known update equation. Members of both vectors are permitted to have some noise, although this may not be explicitly stated in the model. The generic, continuous, nonlinear system definition is shown in Equation 6.

$$\begin{aligned} \dot{X}(t) &= f(t, X(t), u(t), \theta, \nu_x) \\ Y(t) &= g(t, X(t), u(t), \theta, \nu_y) \end{aligned} \tag{6}$$

$X(t)$ is vector of state variables, Θ is a vector of system constants, $u(t)$ is an input, $Y(t)$ the observation, and ν_x and ν_y are random variates. Although any of these variables could be a vector, for the sake of simplicity only Θ and $X(t)$ will be considered as such.

I will also make a few simplifying assumptions for this work. First, the systems are assumed to be time invariant. This assumption is based on the idea that if you paused the system for Δt seconds, when unfrozen the system would continue as if nothing happened. Few biological systems are predictable enough for them to be summarized by a time varying function, least of all the brain. While heart beats are certainly periodic and have an effect on the BOLD signal, the period varies too much for the system to be considered varying with time. Next, its assumes that input cannot directly influence the output, which in the case of the BOLD signal is a good assumption. Finally, because the only difference between the members of $X(t)$ and Θ is an update function, from now on X will encapsulate Θ. The assumptions allow for a simplified version of the state space equations:

$$\dot{X}_k = f(X_{k-1}, u_k, \nu_x) \tag{7}$$

$$Y_k = g(X_k, \nu_y) \tag{8}$$

2) Prior Distribution: The goal of the particle filter is to evolve an empirical distribution $P(x_k|u_{0:k}, Y_{0:k})$, that asymptotically approaches the true probability distribution $P(X_k|u_{0:k})$. Note that capital X will be used as the actual realizations of the state variable, whereas x will denote estimates of X. Additionally, the notation $a : b$ indicates the set $[a, b]$, as in $u_{a:b}$, which would indicate all the inputs from time a to time b. Considering the noise present in X, $P(X_k|u_{0:k})$ is not a single true value but probability distribution.

To begin with, the particle filter must be given a prior distribution, from which the initial N_p particles are drawn. A particle contains a weight as well as an estimate of X_k, which as previously mentioned, contains every variable needed to run the model. Then the prior is generated from a given distribution, $\alpha(X)$, by:

$$\{(x_0^i, w^i) : x_0^i \sim \alpha(X), w^i = \frac{1}{N_p} : i \in \{1, 2, ..., N_p\}\} \tag{9}$$

Where N_p is the number of particles or points used to describe the prior using a Mixture PDF. Note that any exponents will be explicitly labeled as such, to avoid confusion with the particle numbering scheme.

Therefore, as $N_p \to \infty$:

$$\alpha(X) \approx P(x_0) = \sum_{i=0}^{N_p} w^i \delta(X - x_0^i) dx \tag{10}$$

Where $\delta(x - x_0)$ is 1 if and only if $x = x_0$ (the Kronecker delta function). $\alpha(X)$, the prior distribution, *must* cover the true posterior distribution in order for the particle filter to converge to the true posterior distribution.

3) Basic Particle Filter: For all the following areas, the probabilities implicitly depend on $u_{0:k}$, so they are left off for simplicity. Whenever a measurement becomes available it permits refinement of the posterior density. Incorporating new data is called sequential importance sampling. The weight of a particle in an empirical distribution is defined as

$$w_k^i \propto \frac{P(x_{0:k}^i|y_{0:k})}{q(x_{0:k}^i|y_{0:k})} \tag{11}$$

where q is called an *importance density*. The importance density is the density of the particles, thus by dividing by this value, the weight should not depend on the location of the estimation points, but rather only on $P(x_{0:k}^i|y_{0:k})$, the probability of that particle being correct given all the measurements up to time k. Note that if there is a far off peak in the posterior that q does not have support points in, there will be quantization errors, and that part of the density cannot be modeled. This is why q must fully cover $P(x_{0:k}^i|y_{0:k})$.

To update the weight of a particular particle at time k, it is necessary to calculate both $q(x^i_{0:k}|y_{0:k})$ and $P(x^i_{0:k}|y_{0:k})$, in Equation 14 and Equation 16 respectively.

Note that $q(x^i_{0:k}|y_{0:k})$ may be simplified by assuming that y_k doesn't contain any additional information about x_{k-1}.

$$q(x^i_{0:k}|y_{0:k}) = q(x^i_{0:k}|y_{0:k-1}) \tag{12}$$

The choice of the importance density, $q(x_k|x_{k-1}, y_{0:k})$ is a design decision; however it is common to use the state model:

$$q(x_k|x_{k-1}, y_{0:k}) = P(x_k|x_{k-1}) \tag{13}$$

This choice for importance density is helpful because $P(x_k|x_{k-1})$ is freely available: its simply the set of particles propagated forward in time using the state equations. Additionally it makes updating weights simple, as seen later (Equation 17). $q(x^i_{0:k}|y_{0:k})$ may then be simplified:

$$q(x_{0:k}|y_{0:k}) = P(x_k|x_{k-1})q(x_{0:k-1}|y_{0:k-1}) \tag{14}$$

Calculating $P(x_{0:k}|y_{0:k})$ uses the assumption that the distribution of y_k is fully constrained by x_k, and that x_k is similarly fully constrained by x_{k-1}, I make the good assumptions that:

$$\begin{aligned} P(y_k|x_{0:k}, y_{0:k-1}) &= P(y_k|x_k) \\ P(x_k|x_{0:k}, y_{0:k-1}) &= P(x_k|x_{k-1}) \end{aligned} \tag{15}$$

These are restatements of the state equations assumed by Equation 7 and Equation 8. Additionally, for the particle filter y_k and $y_{0:k-1}$ are constant across all particles, thus $P(y_k|y_{0:k-1})$ can be dropped when the equality is changed to a proportion. Using these properties and Bayes' Theorem, $P(x^i_{0:k}|y_{0:k})$ may be broken up as follows:

$$P(x_{0:k}|y_{0:k}) \propto P(y_k|x_k)P(x_k|x_{k-1})P(x_{0:k-1}|y_{0:k-1}) \tag{16}$$

Plugging Equation 13 and the result of Equation 16 into Equation 11 leads to:

$$\begin{aligned} w^i_k &\propto \frac{P(y_k|x^i_k)P(x^i_{0:k-1}|y_{0:k-1})}{q(x^i_{0:k-1}|y_{0:k-1})} \\ &\propto w^i_{k-1}P(y_k|x^i_k) \end{aligned} \tag{17}$$

Thus, by making the following simple assumptions, evolving a posterior density requires no knowledge of noise distribution.

1) $f(t, x(t), u(t)) = f(x(t), u(t))$ and $g(t, x(t), u(t)) = g(x(t))$
2) The prior distribution PDF, $q(x_i(0))$, covers $P(x_i(0))$
3) Markov Property: $P(x_k|x_{0:k-1}) = Pr(x_k|x_{k-1})$
4) $q(x_{0:k-1}|y_{0:k}) = q(x_{0:k-1}|y_{0:k-1})$

From the definition of w_i, the algorithm to calculate an approximation of $P(X(t_k)|Y_{0:k})$ or $P(X(t_k + \delta t)|Y_{0:k})$ is simple (see Algorithm subsection A-A).

4) Sequential Importance Resampling: As a consequence of the wide prior distribution (required for a proper discretization of a continuous distribution), a significant proportion of particles quickly develop insignificant weights. Ideally the computation time spent on tails would be proportional to the actual size of the tails. The case where a large number of the weights have become extremely small is called particle degeneracy. In Lui et al. an ideal measurement of degeneracy based on the particles' true weight is given [23]. An approximation of the effective number of particles is also given:

Parameter	Distribution	μ	σ
τ_0	Gamma	.98	.25
α	Gamma	.33	.045
E_0	Gamma	.34	.03
V_0	Gamma	.04	.03
τ_s	Gamma	1.54	.25
τ_f	Gamma	2.46	.25
ϵ	Gamma	.7	.6

TABLE I
PRIOR DISTRIBUTIONS USED IN THE PARTICLE FILTER.

$$N_{eff} \approx \frac{\sum_{i=0}^{N_p} w_i}{\sum_{i=0}^{N_p} w_i^2} \tag{18}$$

To alleviate particle degeneracy, a technique known as resampling may be applied. The idea of resampling is to draw from the approximate posterior, thus generating a replica of the posterior probability with a matching support. Each particle in the new distribution, $\hat{x}_j$, is drawn from the old distribution (see Algorithm A-B):

$$\hat{x}_j \sim \left(\sum_{i=0}^{N_p} w^i_k \delta(x - x^i_k) \right) \tag{19}$$

For infinite particles this new distribution will match the old. Unfortunately, in practice since the support is limited to the original particles, the number of *unique* particles can only decrease. This effect, called particle impoverishment, can result in excessive quantization errors in the final distribution. To prevent particle impoverishment, instead of sampling from the discrete distribution, a smoothing kernel is applied. Because the distribution is continuous, particle impoverishment is prevented. This regularization process is defined as:

$$x_i = x_i + h\sigma\epsilon \tag{20}$$

Where h is an optional bandwidth, σ is the standard deviation such that $\sigma\sigma^T = cov(x)$ and ϵ is drawn from the chosen kernel. In this case a Gaussian kernel was used, and the bandwidth was set to 1 (see subsection A-B). All the pseudocode may be found in section A, including the complete regularized particle filter used (Algorithm subsection A-D). Equation 17 still depends on knowing $P(y_k|x^i_k)$. Ideally this would exactly match the measurement error distribution of the system; however, even if this distribution is incorrect, the solution will still converge. For this work it was assumed that

$$P(y_k|x^i_k) = g(X_k, \nu_y) = g(X_k) + \nu_y,\ \nu_y \sim N(0, 0.005^2) \tag{21}$$

A standard deviation of 0.005 performed well in the general case, primarily because the BOLD signal tends to be in the range of $0.01 - 0.05$, (1%-5%). Additionally, 1400 Euler integration points were used per second, the system was assumed to start at rest. The system was initialized with 16,000 particles, but during the first resampling, the number of particles was dropped to 1,000. To prevent over-smoothing, resampling was only performed when $_{eff}$ (Equation 18) dropped below 50 for two consecutive measurements.

Region	τ_0	α	E_0	V_0	τ_s	τ_f	ϵ
1	1.454	0.321	0.369	0.036	0.994	2.774	1.348
2	1.151	0.353	0.380	0.026	1.98	2.333	1.645
3	1.951	0.317	0.348	0.027	1.657	3.719	0.757
4	1.203	0.310	0.326	0.036	2.168	2.272	0.086

TABLE II
ACTUAL PARAMETERS FOR EACH REGIONS IN THE POSSUM SIMULATED SLICE. REGIONS SHOWN IN 1A.

C. Preprocessing

The BOLD response has been extensively studied and despite minor discrepancies, the cause of the BOLD signal is well known. However, FMRI has an extremely low SNR. A particularly difficult form of noise present in FMRI is a low frequency drift, often characterized as a Wiener process [24]. Though not present in all regions, as many as ten to fifteen percent of voxels can be affected, thus it is prevalent enough to cause significant inference problems [25], [26]. Though the cause is still not clear it is clear that this drift signal is not solely due to a physiological effects, given its presence in cadavers and phantoms [27]. The problem mandates the use of a high pass filter [26]. In a comparison of high pass filters, Tanabe et al. showed that generally subtracting off a spline improved the SNR the most [25].

To prevent transients at the beginning of the FMRI run, the first 9 measurements (18.9 seconds) were dropped. Each of the remaining images were registered to the new first image, to account for movement. One knot was used for every 20 time points (at a TR of 2.1), with the first knot and last knot taking half as many points as the rest of the knots. The median level of each group of measurements was calculated, and the knot placed at the average time of the group. After subtracting this spline from the timeseries, the resulting timeseries was divided by the original sample mean. To compensate for the BOLD signal not generally being zero-mean, the Median-Absolute-Deviation was added to the final signal (Equation 22).

$$y_{\text{gain},0:K} = 1.4826 \underset{i=0:K}{\text{median}}(y_i - \text{median}(y_{0:K})) \tag{22}$$

For SPM, the first 9 measurements were again dropped, and the images co-registered across time. The co-registered data was spatially smoothed, and then filtered with a cut off of 0.0078125 Hz. (which default in SPM analysis). For the SPM analysis, the Canonical HRF was used and model time derivatives were included (but not in the contrast vector). All other settings used in SPM8 were left at the default for analyzing FMRI data.

D. Simulation

Single voxel simulations were performed by propagating the BOLD model forward through time with 2000 integration points per second, then adding I.I.D. Gaussian noise and a Wiener process with I.I.D. steps. For all single voxel simulations, the signal was generated using the following parameters: $\{\tau_0 = 1.45, \alpha = 0.3, E_0 = 0.47, V_0 = 0.044, \tau_s = 1.94, \tau_f = 1.99, \epsilon = 1.8\}$, with measurement noise (σ_y) of 0.001 and drift standard deviation (σ_x) of 0.0005. Two separate tests are discussed in the results. A single voxel test was performed with for 2400 seconds with random stimuli (Poisson distributed, $\lambda = 1$ event per second) lasting 0.1 seconds. The second single voxel tests discussed were the 310.8 seconds after preprocessing, and used stimulus matching subsection II-E.

Full volume simulations were performed using a modified version of the FSL tool POSSUM. The modifications allowed activation levels to be driven by spatially dependent BOLD parameter sets. Due to the modifications, the SNR configuration was set to 20, although actual SNR's ranged from .5 to 1.5 in active areas.

E. Data Acquisition

For the FMRI data discussed in **??**, tests were performed on a right handed volunteer using a GE SIGNA HDx 1.5 Tesla scanner with a single echo EPI sequence. Slice spacing was 5mm, and pixel sizes were 3.75mm. Repetition Time was 2.1s, Echo Time was 40 ms and the imaging frequency was set to 63.854MHz. The image resolution was $64 \times 64 \times 28$. The subject was presented with either a single or double flash and was asked to respond with a right handed or left handed finger tap, respectively. The FMRI began 18.9 seconds before the beginning of the experiment, to allow for transients in to image to settle out. The timing of the flashes are shown below with time 0 corresponding to the beginning of the 10th TR (so 9 images were dropped).

The timing of the single flashes were: 1.706, 11.944, 17.063, 18.769, 34.125, 39.244, 44.231, 47.644, 49.350, 61.294, 64.706, 66.413, 69.825, 71.531, 73.238, 76.650, 80.063, 90.169, 96.994, 110.644, 117.469, 120.881, 130.988, 132.694, 154.875, 158.288, 161.700, 165.113, 168.525, 176.925, 178.631, 183.750, 190.575, 204.225, 205.931, 211.050, 216.038, 222.863, 226.275, 236.513, 248.456, 255.281, 258.563, 263.681, 273.919, 277.331, 287.569, 292.688, 294.394, 299.381

Timing of the double flashes were: 0.131, 6.825, 20.475, 27.300, 35.831, 52.763, 54.469, 59.588, 86.756, 91.875, 107.231, 108.938, 112.350, 114.056, 115.763, 119.175, 126.000, 134.400, 136.106, 141.225, 144.638, 156.581, 159.994, 166.819, 171.806, 175.219, 185.456, 188.869, 202.519, 212.756, 217.744, 221.156, 227.981, 229.688, 233.100, 243.338, 245.044, 246.750, 250.163, 261.975, 270.506, 272.213, 280.744, 282.450, 289.275, 296.100, 301.088, 304.500

III. RESULTS

A. Simulation

The final square root Mean-Squared-Error (between the estimated and noiseless BOLD signal) for the 2400s run was 0.00381, signal peaks were 0.035 and the final correlation matrix is shown in Table III.

The final parameter estimates of the 310.8s runs are shown in Table IV; the square-root MSE for these parameters ranged from 0.00147 to 0.0027, with a mean of 0.00217. Peak signals reached levels of 0.025.

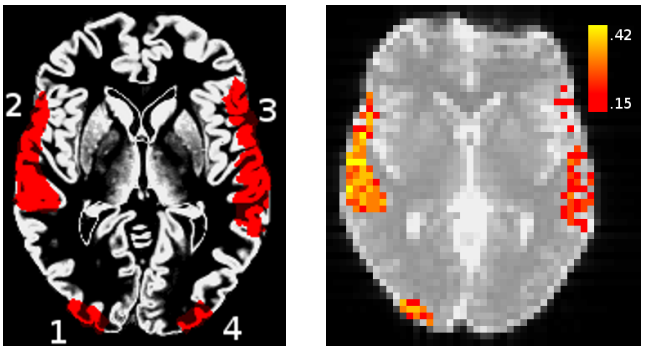
In the POSSUM simulation, areas passing a threshold of mutual information of 0.15 are shown in 1b whereas the active region numbers are shown in 1a. Note that region 4 had a very small ϵ and so had an extremely low SNR ($< .1$). The histograms of parameter estimates for sections 2 and 3 are shown in Figure 2 and Figure 3, respectively. Region 1 is left out because of the low number of points and because its SNR was between region 2 and region 3 (0.8), so it was a less interesting result.

	τ_0	α	E_0	V_0	τ_s	τ_f
α	0.889884					
E_0	-0.7661395	-0.5230723				
V_0	0.6244049	0.4239271	-0.7964774			
τ_s	0.6204843	0.295425	-0.7481253	0.3440421		
τ_f	-0.0004259	-0.3966881	-0.4314174	0.1962954	0.6990775	
ϵ	0.6158116	0.6558179	-0.641348	0.2846632	0.4458142	-0.097079

TABLE III
CORRELATION OF PARAMETER ESTIMATES FOR A 2400 SECOND SIMULATION.

τ_0	α	E_0	V_0	τ_s	τ_f	ϵ
1.45	0.3	0.47	0.044	1.94	1.99	1.8
1.2221	0.3449	0.3346	0.0714	1.6045	2.2753	1.5945
1.3749	0.3318	0.3630	0.0733	1.6408	2.1030	1.5763
1.1660	0.3221	0.3406	0.0822	1.6477	2.3535	1.2452
1.2318	0.3271	0.3403	0.0796	1.6270	2.1852	1.3033
1.1832	0.3179	0.3472	0.0821	1.5496	2.2912	1.2782
1.1424	0.334	0.3473	0.0737	1.6221	2.2908	1.4025
1.3004	0.3596	0.3564	0.0768	1.5641	2.1323	1.6034
1.2401	0.3460	0.3398	0.0891	1.6499	2.2366	1.2900
1.1709	0.3274	0.3464	0.0826	1.5373	2.2826	1.3783
1.1897	0.3434	0.3355	0.0798	1.5358	2.3075	1.4277
1.184	0.3405	0.3502	0.0892	1.6103	2.2793	1.1645
1.2187	0.3359	0.3456	0.0800	1.599	2.2488	1.3876

TABLE IV
ESTIMATED PARAMETERS ON 11 SEPARATE 310.8 SECOND RUNS. TRUE VALUE IN FIRST ROW, AND AVERAGE IN LAST.



(a) Region labels for simulated slice.

(b) Mutual Information Map. Scale is bits. Higher (yellow) is better.

Fig. 1. Comparison of activation with greymatter, parameter regions.

B. Real Data

The result of real experiments concentrates on regions of activation, because of the significant possiblity that the parameters are strictly defined. The quality of fit used for the particle filter results is mutual information which in preliminary testing showed fewer false positives than normalized Mean Squared Error.

IV. DISCUSSION

There are two possible causes for the large variance in parameter estimates. First, it could be that the particle filter is incapable of learning the model. However, given the high quality of BOLD estimates, this is almost certainly not the case. The other possible explanation is that the given stimulus is not sufficiently varied to bring out all the properties of the system. While it is possible that this is the case, it is unlikely that any stimulus sequence could differentiate all the parameters. Its also extremely interesting that the histograms of parameters' mean approach the width of the prior, even when the particle filter converges properly to far thinner distributions. Its possible the prior distribution is far wider than it needs to be. Finally, the fact that **??**, which used randomly spaced impulses, still had a high correlation further indicates that the parameters are ill-defined.

Thus there is little doubt that the parameters of the BOLD model are under constrained. While unsurprising given the sensitivity analysis by Deneux et al., it is a notable conclusion [17]. Other methods that calculate a point estimate of the parameters, such as least squares or Kalman filters (which uses the first two moments) are limited in their capacity to estimate such parameters. While estimates of the BOLD signal may still be correct, the underlying parameters and state variables cannot be described without using a joint probability distribution function. In this sense, particle filters represent an important step forward in BOLD parameter estimation. Representing uncertainty with a mean and variance is insufficient; so using a particle filter or Bayesian estimate of the posterior is not simply an enhancement, but a necessary precaution.

Because of the ill-defined nature of the parameters, the results of Friston et al. may have been wider than true parameter distribution [10]. Future works that attempt to calculate parameters of the BOLD model should be run multiple times to ensure consistency, which most likely cannot be attained. This result is particularly troublesome given the number of papers that use those parameters for a prior distribution.

As the results in **??** show, the heat maps, especially those of mutual information, closely resembled the results of SPM. While the activation tests were more sensitive, there were some additional false positives, though the problem is difficult to quantify. Regions with M.I. above 0.15 consistently fit the FMRI data well, and simulations showed that the particle filter performed extremely well in the face of significant noise. In sum, the estimated BOLD output remained consistent despite large swings in parameters.

The Particle Filter algorithm was originally designed for online parameter estimation. For this reason, there is no guarantee of optimality or even convergence for finite measurements. However, for the BOLD nonlinear ODE this is less of a concern than it might first appear to be. For this particular problem there can be no guarantee of a global minimum, and although other techniques guarantee a local minimum, tests show that the particle filter did converge relatively quickly **??**.

One difficulty with the use of a particle filter when given a finite number of measurements is finding a good weight function $P(y_k|x_k)$. This is more important for a finite number of measurements because $P(y_k|x_k)$ needs to converge in finite time. In spite of this potential problem, **??** managed to converge in less than 500 seconds. Given sufficient measurements it is better to let the algorithm take longer to converge, because the convergence will be less prone to particle deprivation. The particle filter takes longer to run than Volterra approximation method from **??**; however, it is free from the uncertainty of whether a quadratic approximation is sufficient for the BOLD model.

The particle filter also has advantages over other estimation procedures discussed in **??**. The most important advantage is that it provides an estimate of the posterior probability, rather than a single estimate. While researchers often want a simple estimate of parameters, such an estimate is impossible with this particular model.

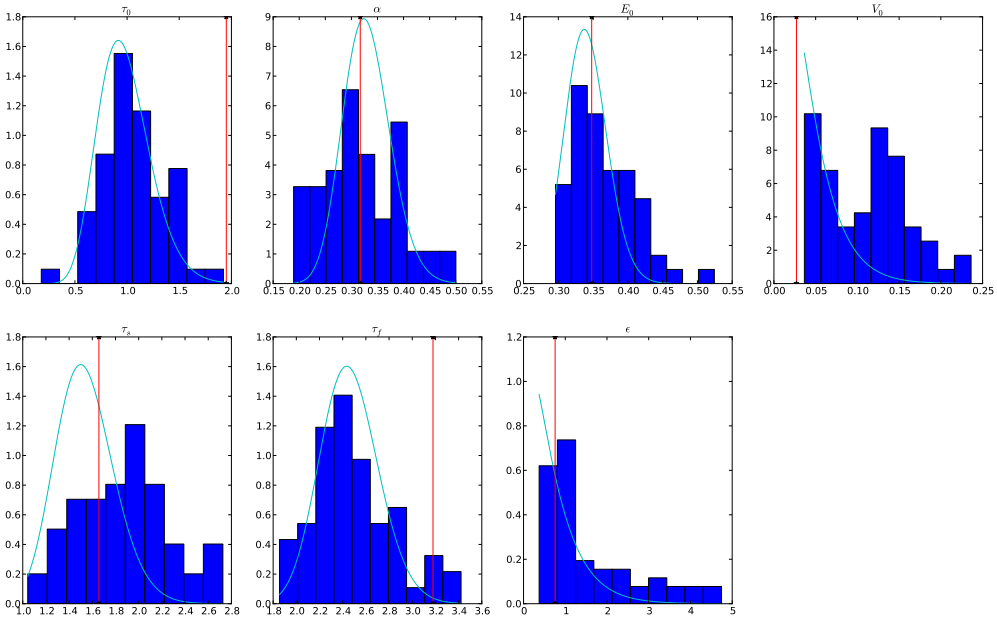


Fig. 2. Histogram of estimated parameters. Average SNR in the region was 0.97 and Mutual Information between simulation and BOLD estimate was greater than 0.15.

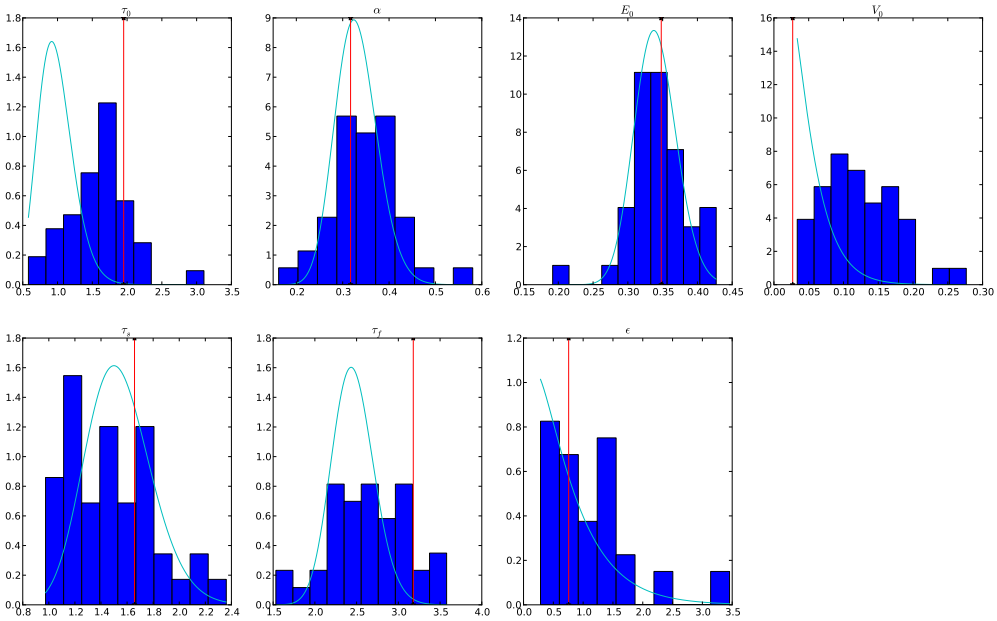


Fig. 3. Histogram of estimated parameters for region 3. Average SNR in the region was 0.39 and Mutual Information between simulation and BOLD estimate was greater than 0.15.

The fact that the final distribution does not need to conform to a parametric distribution is also advantageous, given the nonlinearities in the system. While the particle filter took a day to calculate for full brain calculations, its speed was sufficient on a quad core machine to perform real time calculations of small regions (approximate run time .27 seconds per voxel-measurement). Today it would be possible to perform real time analysis of 10 voxels on an average quad core. The algorithm also scales well and does not require burdensome amounts of memory (approximately 11 megabytes).

There are a number of advantages to the particle filter approach presented here. In the past, FMRI data has been analyzed strictly for determining correlation between a stimulus and response. With this new method the correlation is simply a means to determining the joint posterior distribution of the parameters. While only regions that correlate with the input will be calculable, this method exploits that correlation to constrain the prior distribution of the parameters. The resulting distribution, while difficult to visualize because of the high-dimensionality, could nevertheless be correlated with neural pathologies. In spite of the fact that the parameters are under-determined, the final distribution is still a reduction in the uncertainty

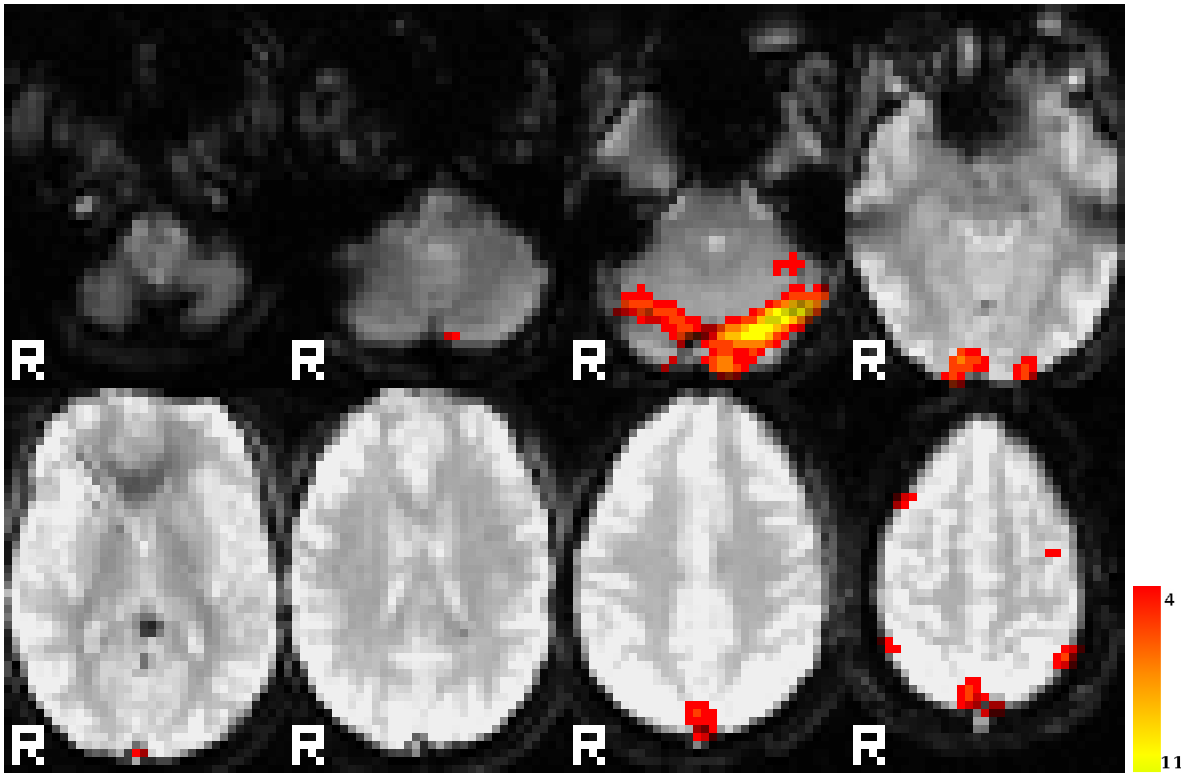


Fig. 4. SPM results. Units of activation are in Student's t-scores; higher indicates higher assurance that the signal cannot have occurred through noise alone.

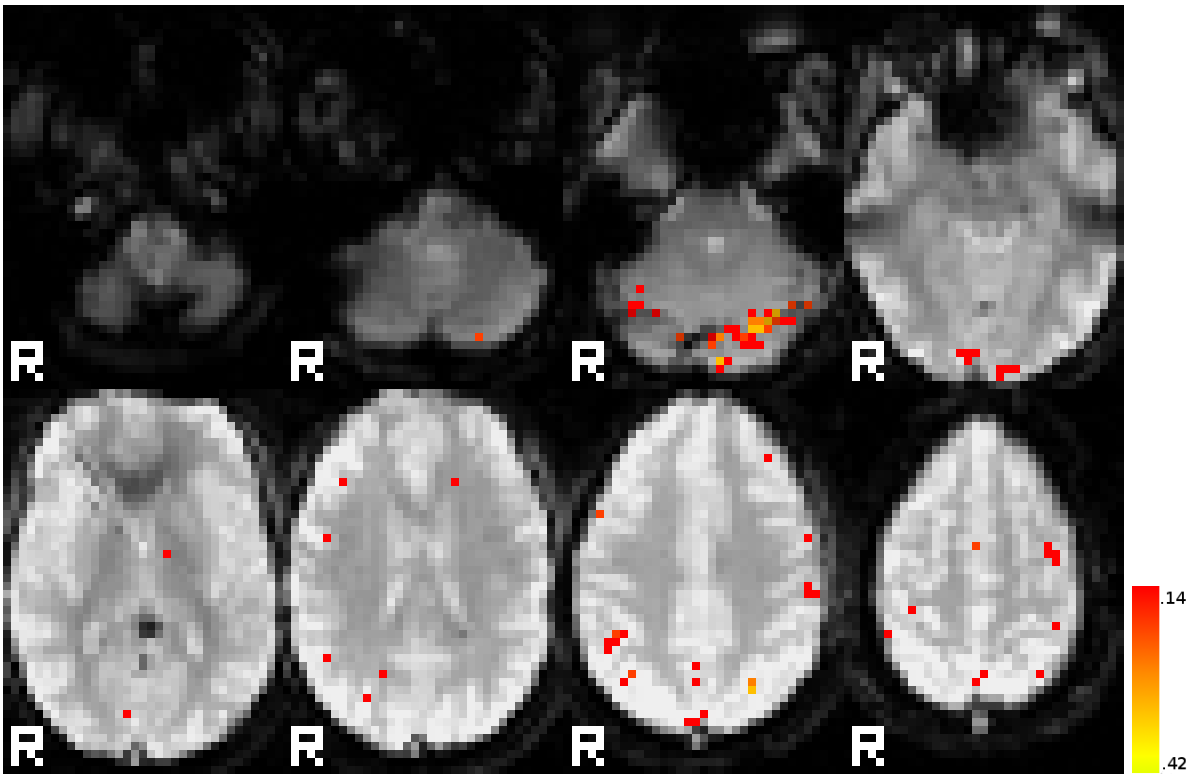


Fig. 5. Particle filter results. Units of activation are in mutual information; higher indicates more assured activation.

of the parameters. The availability of a full posterior distribution opens up many avenues for further inquiry, for instance to compare normal vs. symptomatic populations. This would be especially useful in patients whose symptoms are caused by functional difference in the brain, rather than structural differences. This would effectively be a way to differentiate the way the brains operate.

Because of limitations present in every imaging modality, its becoming increasingly clear that combining data from multiple sources will be necessary to push Neurology forward. In order to do so however, the output of each source needs to fully represent what information that source can provide. Combining the sources using Bayesian statistics is promising yet often difficult because full probability distributions are hard to come by. However, in this case, the particle filter provides a full posterior which is extremely versatile. Therefore, future works will easily be able to plug in data from multiple sources if they all output Bayesian posteriors. For instance, if two different modalities have calculated the probability distribution of neural efficiency, those two beliefs may be combined into one conditional belief for a new conditional probability of neural efficiency.

An advantageous aspect of using a physiological model such as this, is that it permits estimates of otherwise hidden parameters. In particular the BOLD model gives an estimate of the value of the flow inducing signal, s. Having this value available opens up new avenues for determining inter-regional dependencies. The values of s could serve as a proxy for activation in that region and thus could be used to drive other inputs regions' $u(t)$. Thus, the particle filter could be re-run with the time-series of a particular voxel's s value as an additional stimulus. In this way, it could be possible to determine chains of events. This is just one possible benefit being able to determine the time course of the hidden state variables present in the BOLD equations; the potential benefits of being able to determine this information is limitless. Of course the quality of these estimates will continue to improve with the model priors.

APPENDIX A
PARTICLE FILTER ALGORITHM

A. Sequential Importance Sampling

Initialize Particles:
for i : each of N_p particles **do**
  $x_0^i \sim \alpha(X)$
  $w_0^i = \frac{1}{N_p}$
end for
for k : each measurement **do**
  for i : each particle **do**
    $x_k^i = x_{k-1}^i + \int_{t-1}^{t} f(x(\tau), u(\tau))d\tau$
    $w_k^i = w_{k-1}^i P(y_k|x_k)$
  end for
end for

$$P(x(t_k + \Delta t)) \approx \sum_{i=0}^{N_p} w_k^i \delta \left(x - x_k^i - \int_{t_k}^{t_k+\Delta t} f(x(\tau), u(\tau))d\tau \right)$$

B. Resampling Algorithm

Calculate total weight, $W_t = \sum_{i=0}^{N_p} w^i$
for all $0 < i < N_p$ **do**
  Draw V from uniform range $[0, W]$
  $C = W_t$
  for all $0 < j < N_p$ and $C < V$ **do**
    $C = C - w^j$
  end for
  Add $[x^j, \frac{1}{N_p}]$ to the new distribution
end for

C. Regularized Resampling Algorithm

Calculate Covariance, C, of empirical distribution, $\hat{x}$
Find D such that $DD^T = C$
Resample $\hat{x}$ using algorithm from A-B
for $0 < i < N_p$ **do**
  Draw ϵ from the standard normal, same dimensionality as X
  $x^i = x^i + hD\epsilon$
end for

D. Regularized Particle Filter

Initialize Particles:
for i : each of N_p particles **do**
  $x_0^i \sim \alpha(X)$
  $w_0^i = \frac{1}{N_p}$
end for
for k : each measurement **do**
  for i : each particle **do**
    $x_k^i = x_{k-1}^i + \int_{t-1}^{t} f(x(\tau), u(\tau))d\tau$
    $w_k^i = w_{k-1}^i P(y_k|x_k)$
  end for
  Calculate N_{eff} with Equation 18
  if $N_{eff} < N_R$ (recommend $N_R = min(50, .1N_p)$) **then**
    Resample using algorithm from subsection A-B
  end if
end for

At $t + \Delta t$, $t \in T$, $P(x(t + \Delta t)) \approx \sum_{i=1}^{N_p} w_i(t) \delta \left(x - (x_i(t) + \int_{t}^{t+\Delta t} f(x(\tau), u(\tau))d\tau) \right)$

APPENDIX B

Appendix two text goes here.

REFERENCES

[1] D. A. Handwerker, J. M. Ollinger, and M. D'Esposito, "Variation of BOLD hemodynamic responses across subjects and brain regions and their effects on statistical analyses." *NeuroImage*, vol. 21, no. 4, pp. 1639–51, Apr. 2004. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/15050587

[2] T. D. Wager, A. Vazquez, L. Hernandez, and D. C. Noll, "Accounting for nonlinear BOLD effects in fMRI: parameter estimates and a model for prediction in rapid event-related studies." *NeuroImage*, vol. 25, no. 1, pp. 206–18, Mar. 2005. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/15734356

[3] R. M. Birn, Z. S. Saad, and P. A. Bandettini, "Spatial heterogeneity of the nonlinear dynamics in the FMRI BOLD response." *NeuroImage*, vol. 14, no. 4, pp. 817–26, Oct. 2001. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/11554800

[4] J. J. Chen and G. B. Pike, "Origins of the BOLD post-stimulus undershoot." *NeuroImage*, vol. 46, no. 3, pp. 559–68, 2009. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/19303450

[5] R. B. Buxton, E. C. Wong, and L. R. Frank, "Dynamics of blood flow and oxygenation changes during brain activation: the balloon model," *Magnetic Resonance in Medicine*, vol. 39, pp. 855–864, 1998.

[6] R. M. Weisskoff, C. S. Zuo, J. L. Boxerman, and B. R. Rosen, "Microscopic susceptibility variation and transverse relaxation : theory and experiment," *Magnetic Resonance in Medicine*, vol. 31, no. 6, pp. 601–610, 1994. [Online]. Available: http://cat.inist.fr/?aModele=afficheN&cpsidt=4091595

[7] S. Ogawa, R. S. Menon, D. W. Tank, S. Kim, H. Merkle, J. M. Ellermann, and K. Ugurbilt, "Functional brain mapping by blood oxygenation level-dependent contrast magnetic resonance imaging A comparison of signal characteristics with a biophysical model," *Biophysical Journal*, vol. 64, pp. 803–812, 1993.

[8] T. Obata, "Discrepancies between BOLD and flow dynamics in primary and supplementary motor areas: application of the balloon model to the interpretation of BOLD transients," *NeuroImage*, vol. 21, no. 1, pp. 144–153, 2004. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1053811903005494

[9] K. J. Friston, A. Mechelli, R. Turner, and C. J. Price, "Nonlinear Responses in fMRI: the Balloon Model, Volterra kernels, and other Hemodynamics," *NeuroImage*, vol. 12, pp. 466–477, 2000.

[10] K. J. Friston, "Bayesian estimation of dynamical systems: an application to fMRI," *NeuroImage*, vol. 16, pp. 513–530, 2002.

[11] J. J. Riera, J. Watanabe, I. Kazuki, M. Naoki, E. Aubert, T. Ozaki, and R. Kawashima, "A state-space model of the hemodynamic approach: nonlinear filtering of BOLD signals," *NeuroImage*, vol. 21, pp. 547–567, 2004.

[12] V. A. Vakorin, O. O. Krakovska, R. Borowsky, and G. E. Sarty, "Inferring neural activity from BOLD signals through nonlinear optimization." *NeuroImage*, vol. 38, no. 2, pp. 248–60, Nov. 2007. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/17825582

[13] L. A. Johnston, E. Duff, I. Mareels, and G. F. Egan, "Nonlinear estimation of the BOLD signal." *NeuroImage*, vol. 40, no. 2, pp. 504–14, 2007. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/18203623

[14] L. Murray and A. Storkey, "Continuous Time Particle Filtering for fMRI," *Advances in Neural Information Processing Systems*, vol. 20, pp. 1049–1068, 2008.

[15] Z. Hu, X. Zhao, H. Liu, and P. Shi, "Nonlinear Analysis of the BOLD Signal," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, pp. 1–14, 2009. [Online]. Available: http://www.hindawi.com/journals/asp/2009/215409.html

[16] E. Yacoub, K. Ugurbil, and N. Harel, "The spatial dependence of the poststimulus undershoot as revealed by high-resolution BOLD- and CBV-weighted fMRI," *Journal of Cerebral Blood Flow and Metabolism*, vol. 26, pp. 634–644, 2006.

[17] T. Deneux and O. Faugeras, "Using nonlinear models in fMRI data analysis: model selection and activation detection," *NeuroImage*, vol. 32, no. 4, pp. 1669–1689, 2006.

[18] Y. Behzadi and T. T. Liu, "An arteriolar compliance model of the cerebral blood flow response to neural stimulus," *NeuroImage*, vol. 25, pp. 1100–1111, 2005.

[19] J. B. Mandeville, J. J. A. Marota, C. Ayata, G. Zaharchuk, M. A. Moskowitz, B. R. Rosen, and R. M. Weisskoff, "Evidence of a cerebrovascular postarteriole Windkessel with delayed compliance," *Journal of Cerebral Blood Flow and Metabolism*, vol. 19, no. 6, pp. 679–689, 1999.

[20] R. B. Buxton, K. Uludag, D. J. Dubowitz, and T. Lui, "Modeling the hemodynamic response to brain activation." *NeuroImage*, vol. 23 Suppl 1, pp. S220–33, 2004. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/15501093

[21] 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, no. 2, pp. 174–188, 2002. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=978374

[22] S. Thrun, W. Burgard, and D. Fox, "Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)," *cognition*, vol. 96, 2008.

[23] A. K. Liu, J. W. Belliveau, and A. M. Dale, "Spatiotemporal imaging of human brain activity using functional MRI constrained magnetoencephalography data: Monte Carlo simulations." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 95, no. 15, pp. 8945–50, Jul. 1998. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/9671784

[24] J. Riera, J. Bosch, O. Yamashita, R. Kawashima, N. Sadato, T. Okada, and T. Ozaki, "fMRI activation maps based on the NN-ARx model," *NeuroImage*, vol. 23, pp. 680–697, 2004.

[25] J. Tanabe, D. Miller, J. Tregellas, R. Freedman, and F. G. Meyer, "Comparison of detrending methods for optimal fMRI preprocessing," *NeuroImage*, vol. 15, no. 4, pp. 902–907, 2002.

[26] A. T. Smith, K. D. Singh, and J. H. Balsters, "A comment on the severity of the effects of non-white noise in fMRI time-series." *NeuroImage*, vol. 36, no. 2, pp. 282–8, 2007. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/17098446

[27] A. M. Smith, B. K. Lewis, U. E. Ruttimann, F. Q. Ye, T. M. Sinnwell, Y. Yang, J. H. Duyn, and J. A. Frank, "Investigation of Low Frequency Drift in fMRI Signal," *NeuroImage*, vol. 533, no. 5, pp. 526–533, 1999.