BOLD Parameter Estimation using Sequential Monte Carlo Methods

Micah Chambers

Virginia Tech Bioimaging Systems Lab

February 24, 2010

VirginiaTech Invent the Future

Outline

The BOLD Response

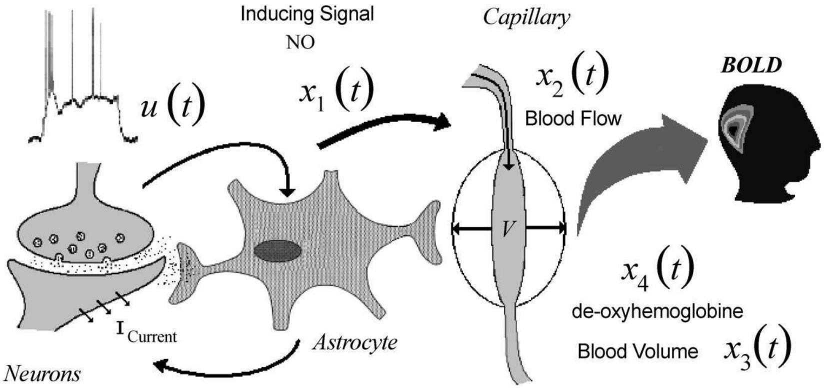
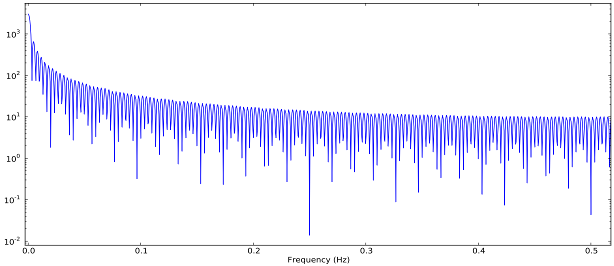


Figure: [Riera et al.(2004)Riera, Watanabe, Kazuki, Naoki, Aubert, Ozaki, and Kawashima]

BOLD Signal Properties

- Exact variables and parameters are unknown and are difficult to calculate.
- Significant Amount of Lag between activation and a measurable output - can be as much as 8 seconds.
- Slow Temporal Resolution
- Noise characterized by brownian motion, which clashes with low frequency elements.



Preprocessing

- Low Pass Filter (Gaussian Filter, not recommended)
- Drift Removal (not always performed)
  - High Pass Filter
  - Linear
  - Quadratic
  - Wavelet
  - Spline (Which I am using)

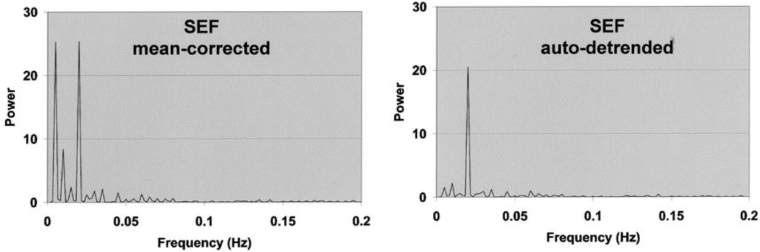


Figure: [Tanabe et al.(2002)Tanabe, Miller, Tregellas, Freedman, and Meyer]

Method

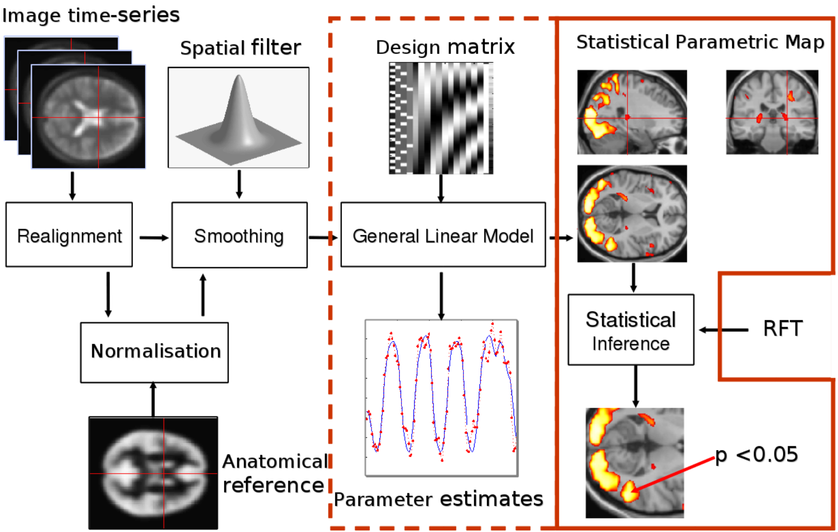
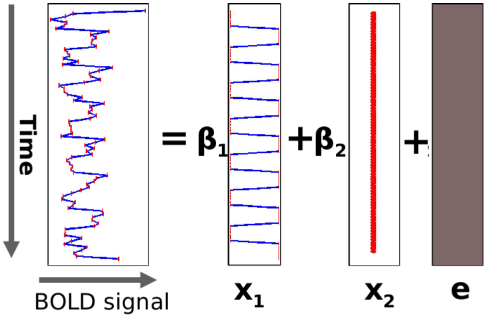


Figure: [Klaas(2009)]

Limitations

- Linear, for a signal which is known to be nonlinear
- Essentially the weighted sum of a set of "expected" responses.
- Parametric
    - Forced to make assumptions about underlying distributions
    - No time-scaling.



$$y = x_1\beta_1 + x_2\beta_2 + e$$

Figure: [Klaas(2009)]

Equations

- Normalized Cerebral Blood Flow:

$$\ddot{f}(t) = \epsilon u(t) - \dot{f}(t)/\tau_s - (f(t)/\tau_f - 1)$$

- Normalized Cerebral Blood Volume:

$$\dot{v}(t) = (1/\tau_0)(f(t) - v(t)^{1/\alpha})$$

- Normalized Deoxyhaemoglobin Content:

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

- Hemodynamic Response - BOLD Signal

$$y(t) = V_0(a_1(1 - Q(t)) - a_2(1 - V(t)))$$

Model Comparison

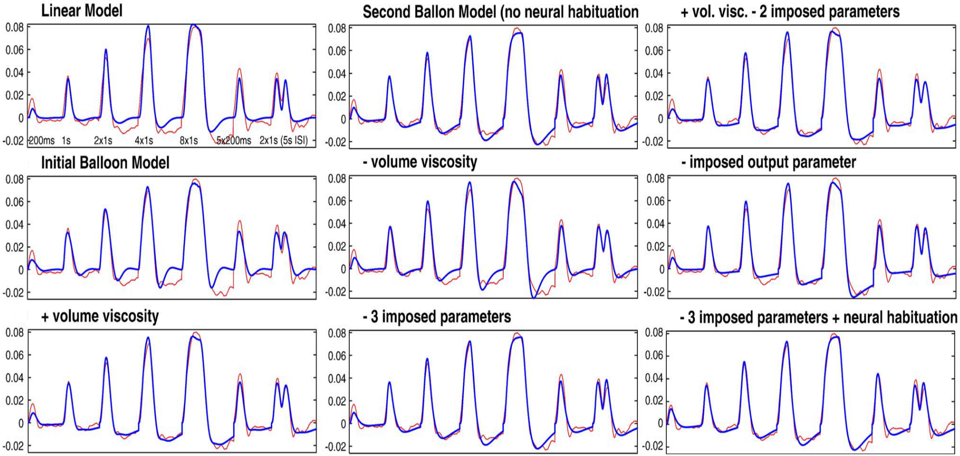


Figure: [Deneux and Faugeras(2006)]

Particle Filters

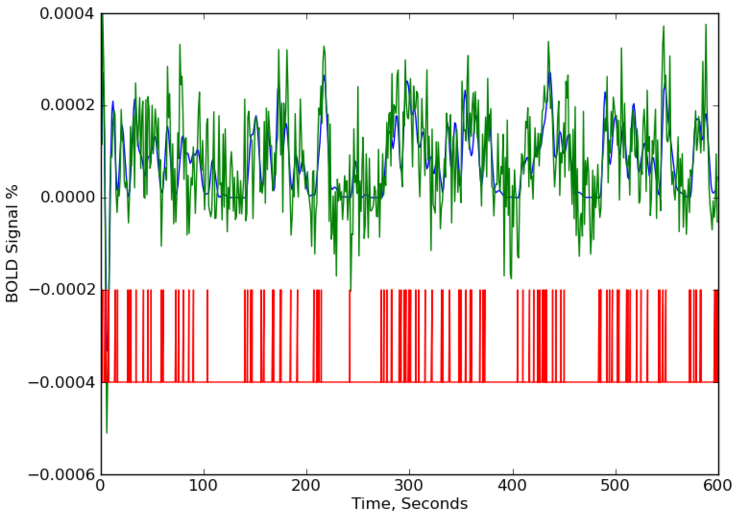
- Non-parametric, no assumptions are violated
- Model based, fit parameters to input, constrained by physical variables
- Fits a mixture PDF to the posterior of all parameters
- Non-trivial computation cost
- I use a Regularized Particle Filter
  1. Regularized Re-sampling prevents particles from de-generating into a small number of unique particles
  2. Allows distributions to move more freely

Particle Filter

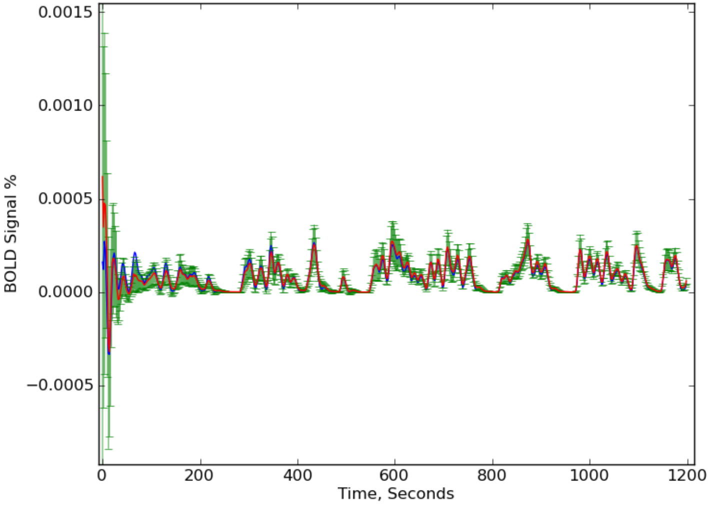
- $S_t = \{p_{0,t}, ..., p_{N,t}\}$, the set of particles
- $w_{i,t}$, weight of particle $p_{i,t}$
- y_t, measurement at time t, there is not a y_t for every t.
- $f(p_{i,t}, y_t)$, weighting function
- $s(p_{i,t})$, step function

Draw S_0 from prior distribution
for $t = 0 : t_{step} : t_{end}$ **do**
  for each $p_{i,t-1} \in S_{t-1}$ **do**
    $p_{i,t} = s(p_{i,t-1})$
    if There is a measurement at time t **then**
      for every $p_{i,t}$ **do**
        $w_{i,t} = w_{i,t-1} f(p_{i,t}, y_t)$
      end for
      Resample if weights are unevenly distributed
    end if
  end for
end for

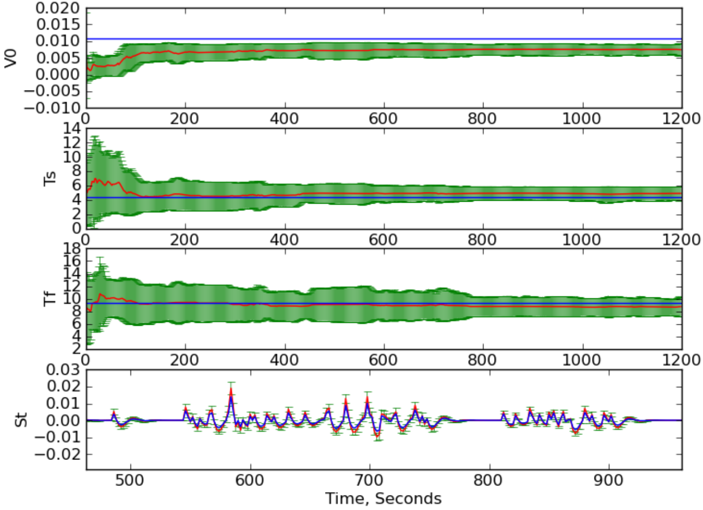
Single Timeseries Results



Single Timeseries Results, Measurement Convergence



Single Timeseries Results, State Convergence

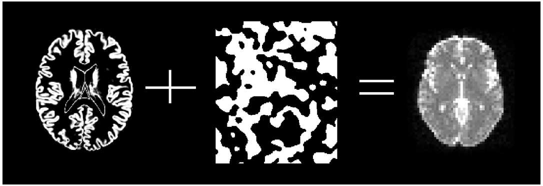


Factors Affecting Convergence

1. Weighting function
   - Needs to be continuous and defined for any input, should go to 0
   - Too wide a weighting function results in under-sensitivity, slow or no convergence
   - Too thin a weighting function reduces robustness to noise
2. How often re-sampling is done, re-sampling should be minimized
   - Stratified Resampling can result in truncated tails on posterior
   - Regularized Resampling can result in reduced robustness to noise
3. Number of particles
   - More particles give higher fidelity of posterior

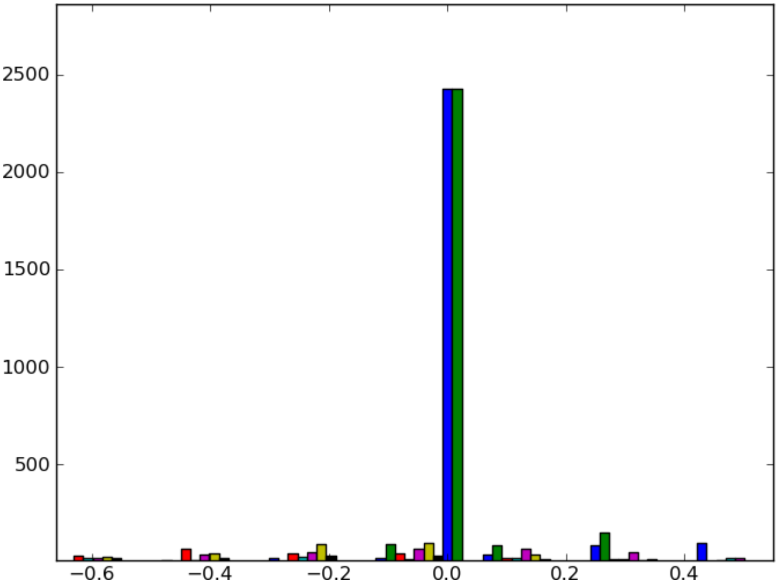
Parameter Map Generation/Simulation

1. Generate a parameter map, with a set of parameters for each voxel
2. Simulate every set of parameters, and use as input to possum
3. Perform preprocessing (de-trend and normalize)
4. Run particle filter on every grey matter voxel in image, generating a new parameter map



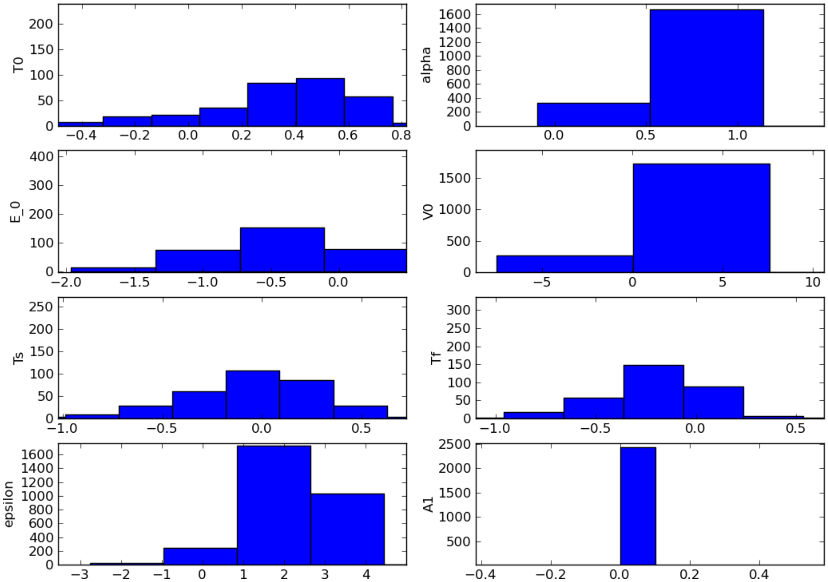
Figure

Simulation Results, τ_0



Virginia Tech
Invent the Future

Simulation Results, multiple



T. Deneux and O. Faugeras.
Using nonlinear models in fMRI data analysis: Model selection and activation detection.
NEUROIMAGE, 32(4):1669–1689, OCT 1 2006.
ISSN 1053-8119.
doi: {10.1016/j.neuroimage.2006.03.006}.

S. Klaas.
General Linear Model, 2009.
URL http://www.sciencedirect.com/science/article/B6WNP-47DKV9V-2/2/ccfad2db718319db51008a47c71ca54a.

J. Riera, J. Watanabe, T. Kazuki, M. Naoki, E. Aubert, T. Ozaki, and R. Kawashima.
A state-space model of the hemodynamic approach: nonlinear filtering of BOLD signals.
NEUROIMAGE, 21(2):547–567, FEB 2004.
ISSN 1053-8119.
doi: {10.1016/j.neuroimaging.2003.09.052}.

J. Tanabe, D. Miller, J. Tregellas, R. Freedman, and F. Meyer.
Comparison of detrending methods for optimal fMRI preprocessing.
NEUROIMAGE, 15(4):902–907, APR 2002.
ISSN 1053-8119.
doi: {10.1006/nimg.2002.1053}.