Statistica Sinica **18**(2008): Supplement, S??∼S??

CONTINUOUS-TIME FILTERS FOR STATE ESTIMATION FROM POINT PROCESS MODELS OF NEURAL DATA

Uri T. Eden and Emery N. Brown

Boston University and Massachusetts Institute of Technology

Supplementary Material

This online supplement to 'Continuous-Time Filters for State Estimation from Point Process Models of Neural Data,' by Eden, U.T. and Brown, E.N., provides extended information about the theory and methods discussed in the manuscript. The first section provides a step-by-step derivation of the continuous time approximate filter detailed in section VII of the paper. The second section provides details about the simulation methods and parameters used in the decoding example presented in section VIII of the paper.

I. Step-by-step derivation of the continuous time approximate filter

In Eden, Frank, Barbieri, Solo and Brown (2004), we constructed a discrete time point process filter based on a Gaussian approximation to the posterior distribution of the state given the observed spiking activity to the current time, called the Stochastic State point Process Filter (SSPPF). When the state equation is expressed in discrete time as in equation 3.2 and the observations are point processes with conditional intensities $\lambda^i(t_k)$ as defined in equation 3.3, then the recursive update equations for estimators of the variance and mean of the posterior distribution are respectively,

$$(W_{k|k})^{-1} = (F_k W_{k-1|k-1} F_k^T + Q_k)^{-1} + \sum_{j=1}^{C} \Bigg[\left(\frac{\partial \log \lambda^j}{\partial x_k} \right)^T [\lambda^j \Delta t_k] \left(\frac{\partial \log \lambda^j}{\partial x_k} \right) \\ -(\Delta N_k^j - \lambda^j \Delta t_k) \frac{\partial^2 \log \lambda^j}{\partial x_k \partial x_k^T} \Bigg]_{x_k = F_k x_{k-1|k-1}}$$

and,

$$x_{k|k} = F_k x_{k-1|k-1} + W_{k|k} \sum_{j=1}^{C} \left[\left(\frac{\partial \log \lambda^j}{\partial x_k} \right)^T (\Delta N_k^j - \lambda^j \Delta t_k) \right]_{x_k = F_k x_{k-1|k-1}} .$$

When expressed equivalently in terms of the parameters of the continuous time state equation (3.1), the filter equations become,

$$\begin{aligned}(W_{k|k})^{-1} = \Bigg(& \exp(A\Delta t_k)W_{k-1|k-1}\exp(A^T\Delta t_k) \\ & + \int_0^{\Delta t_k} \exp(A(\Delta t_k-\tau))BB^T\exp(A^T(\Delta t_k-\tau))\Bigg)^{-1} \\ & + \sum_{j=1}^{C}\Bigg[\left(\frac{\partial\log\lambda^j}{\partial x_k}\right)^T[\lambda^j\Delta t_k]\left(\frac{\partial\log\lambda^j}{\partial x_k}\right) \\ & \qquad -(\Delta N_k^j-\lambda^j\Delta t_k)\frac{\partial^2\log\lambda^j}{\partial x_k\partial x_k^T}\Bigg]_{x_k=\exp(A\Delta t_k)x_{k-1|k-1}}, \end{aligned} \tag{S.1}$$

and,

$$\begin{aligned}x_{k|k} = & \exp(A\Delta t_k)x_{k-1|k-1} \\ & + W_{k|k}\sum_{j=1}^{C}\Bigg[\left(\frac{\partial\log\lambda^j}{\partial x_k}\right)^T(\Delta N_k^j-\lambda^j\Delta t_k)\Bigg]_{x_k=\exp(A\Delta t_k)x_{k-1|k-1}}. \end{aligned} \tag{S.2}$$

For the derivation of a continuous time analogue to the SSPPF, we examine, without loss of generality, the observation interval $[0,\Delta t]$, and the values of the estimated mean and variance process at the beginning and end of this interval in the limit as $\Delta t \to 0$. We define $\hat{x}$ and $\hat{W}$ as our continuous time expressions for the posterior mean and variance respectively, in this limit, and write $\hat{x}^- = \lim_{\Delta t\to 0} W_{0|0}$ and $\hat{x}^+ = \lim_{\Delta t\to 0} W_{1|1}$ for the mean estimator and $\hat{W}^- = \lim_{\Delta t\to 0} W_{0|0}$ and $\hat{W}^+ = \lim_{\Delta t\to 0} W_{1|1}$ for the variance estimator.

We can expand each of the exponential terms in a Taylor series about the point $\Delta t = 0$, and rewrite the posterior variance equation $(W_{1|1})^{-1} = P^{-1} + C$, where

$$P = (I + A\Delta t)W_{0|0}(I + A^T\Delta t) + BB^T\Delta t + O(\Delta t^2), \tag{S.3}$$

and

$$C = \sum_{j=1}^{C}\Bigg[\left(\frac{\partial\log\lambda^j}{\partial x}\right)^T[\lambda^j\Delta t_k]\left(\frac{\partial\log\lambda^j}{\partial x}\right) - (\Delta N^j - \lambda^j\Delta t)\frac{\partial^2\log\lambda^j}{\partial x\partial x^T}\Bigg]_{x=\exp(A\Delta t)x_{0|0}} \tag{S.4}$$

Once again, we deal with the case of spike times and non-spike times separately. When a spike occurs, we are interested in the change in the estimator from immediately before to immediately after the spike. Therefore we can drop any terms of order $O(\Delta t)$. Clearly, if $\partial^2 \log \lambda^j / \partial x \partial x^T = 0$, the estimated variance process will not jump when there is a spike at t. When this term is nonzero, we first use the matrix inversion lemma to write,

$$W_{1|1} = P - P(P + C^{-1})^{-1} P, \tag{S.5}$$

and use the fact that in the limit $\lim_{\Delta t \to 0} P = \hat{W}^-$ and

$$\lim_{\Delta t \to 0} C = - \sum_{\substack{j \in \{\text{spiking} \\ \text{neuronsat } t\}}} \left[\frac{\partial^2 \log \lambda^j}{\partial x \partial x^T} \right]_{\hat{x}^-}$$

to compute the jump in the variance estimator,

$$W^+ - W^- = -W^- \left(W^- - \left(\sum_{\substack{j \in \{\text{spiking} \\ \text{neuronsat } t\}}} \left[\frac{\partial^2 \log \lambda^j}{\partial x \partial x^T} \right]_{\hat{x}^-} \right)^{-1} \right)^{-1} W^-. \tag{S.6}$$

We similarly compute the jump in the mean estimator following a spike by taking the limit of equation S.2,

$$x^+ - x^- = \hat{W}^+ \sum_{\substack{j \in \{\text{spiking} \\ \text{neuronsat } t\}}} \left[\left(\frac{\partial \log \lambda^j}{\partial x} \right)^T \right]_{\hat{x}^-}. \tag{S.7}$$

Now for non-spike intervals, we use the matrix inversion lemma again on the inverse in equation S.5,

$$w_{1|1} = P - PCP + PC(P^{-1} + C)^{-1} CP. \tag{S.8}$$

Examining the limit of the observed component of the variance equation, we see that in the non-spike interval,

$$\lim_{\Delta t \to 0} C = \sum_{j=1}^{C} \lambda^j \Delta t \left[\left(\frac{\partial \log \lambda^j}{\partial x} \right)^T \left(\frac{\partial \log \lambda^j}{\partial x} \right) + \frac{\partial^2 \log \lambda^j}{\partial x \partial x^T} \right]_{\hat{x}^-} = O(\Delta t). \tag{S.9}$$

Since P is positive definite and $C = O(\Delta t)$, the final term of equation S.8 vanishes in the limit. Subtracting $W_{0|0}$ from both sides of equation S.8, dividing by Δt, and taking the limit, we obtain an expression for the derivative of the covariance estimator in the absence of spiking activity,

$$\lim_{\Delta t \to 0} \frac{x_{1|1} - x_{0|0}}{\Delta t} = \lim_{\Delta t \to 0} AW_{0|0} + W_{0|0} A^T + BB^T$$

$$
\begin{aligned}
&-W_{0|0}\sum_{j=1}^{C}\left[\left(\frac{\partial \log \lambda^j}{\partial x}\right)^T\left(\frac{\partial \log \lambda^j}{\partial x}\right)\right.\\
&\left.+\frac{\partial^2 \log \lambda^j}{\partial x \partial x^T}\right]_{x=\exp(A\Delta t)x_{0|0}} W_{0|0}+O(\Delta t)\\
&= AW^- + W^-A^T + BB^T - W^-\sum_{j=1}^{C}\left[\frac{\partial^2 \lambda^j}{\partial x \partial x^T}\right]_{\hat{x}^-}\hat{W}^-. \qquad \text{(S.10)}
\end{aligned}
$$

We can construct a similar expression for the derivative of the mean estimator with respect to time when neurons are not spiking,

$$
\begin{aligned}
\lim_{\Delta t\to 0}\frac{x_{1|1}-x_{0|0}}{\Delta t} &= \lim_{\Delta t\to 0} Ax_{0|0} - W_{1|1}\sum_{j=1}^{C}\left[\lambda^j\left(\frac{\partial \log \lambda^j}{\partial x}\right)^T\right]_{\exp(A\Delta t)x_{0|0}} + O(\Delta t)\\
&= A\hat{x}^- - \hat{W}^+\sum_{j=1}^{C}\left[\lambda^j\left(\frac{\partial \log \lambda^j}{\partial x}\right)^T\right]_{\hat{x}^-}. \qquad \text{(S.11)}
\end{aligned}
$$

We obtain a continuous time stochastic differential equation describing the covariance estimator by combining equations S.6 and S.10, and another describing the mean estimator by combining equations S.7 and S.11,

$$
dx = Ax^-dt + \hat{W}^+\sum_{j=1}^{C}\left[\left(\frac{\partial \log \lambda^j}{\partial x}\right)^T(dN^j - \lambda^j dt)\right]_{\hat{x}^-}, \qquad \text{(S.12)}
$$

$$
\frac{d\hat{W}}{dt} = AW^- + W^-A^T + BB^T - \sum_{j=1}^{C} W^-\left(\frac{d^2\lambda^j}{dxdx^T} + S^j\frac{dN^j}{dt}\right)_{\hat{x}-} W^-, \qquad \text{(S.13)}
$$

where,

$$
S^j = \begin{cases} \left(W^- - \left(\dfrac{\partial^2 \log \lambda^j}{\partial x \partial x^T}\right)^{-1}\right)^{-1} & \text{if } \left(\dfrac{\partial^2 \log \lambda^j}{\partial x \partial x^T}\right) \neq 0 \\ 0 & \text{otherwise} \end{cases}.
$$

From any starting point, this system of differential equations completely describes the evolution of the Gaussian approximation to the posterior density in time.

In some rare cases, this system of equations can be solved exactly. For example, if the state is fixed, that is $A = 0$ and $B = 0$, and the firing model is a generalized linear model with a logarithmic link function, $\lambda(t) = \exp(\alpha + \beta x(t))$, and no spike is observed in some interval $[t_0, t]$, then the solution in that interval is given by,

$$\hat{x} = \beta^{-1}\left[-\frac{1}{2}\log(2W_0\lambda_0^{-1}\beta^2 t + \lambda_0^{-2}) - \alpha\right], \tag{S.14}$$

$$\hat{W} = (2\lambda_0 W_0^{-1}\beta^2 t + W_0^{-2})^{-\frac{1}{2}}. \tag{S.15}$$

When a spike occurs $\hat{x}$ jumps to $\hat{x} + \beta\hat{W}$ and follows equation S.14 with new initial conditions given by the estimates at the time of the spike. $\hat{W}$ does not jump at spike times since $d^2 \log \lambda/dx^2 = 0$, but it does follow a new trajectory determined by equation S.15 with the new initial conditions. In most cases, however, it is not possible to solve the differential equations given by S.12 and S.13 symbolically.

II. Simulation details for neural decoding example

The hand movement trajectories were simulated using an AR(100) process, with autoregressive parameters suggested from previous analyses of a monkey performing continuous reaching movements to randomly appearing targets (behavioral methods in Truccolo, Eden, Fellows, Donoghue and Brown (2005)). Specifically, to simulate an arm movement with speed, $v(t)$, and direction, $\phi(t)$, the state vector, $x(t) = \begin{bmatrix} v(t)\cos(\phi(t)) \\ v(t)\sin(\phi(t)) \end{bmatrix}$, was drawn randomly at discrete time points using the AR(100) equation:

$$x(t_k) = \sum_{i=1}^{100} \begin{bmatrix} a_i & 0 \\ 0 & b_i \end{bmatrix} x(t_{k-1}) + \begin{bmatrix} \sigma_x \\ \sigma_y \end{bmatrix} G_k, \tag{S.16}$$

where t_k are discrete time points spaced a 1 msec intervals, $G_k \sim N(0, I)$ is drawn from a zero-mean bivariate Gaussian distribution, $\sigma_x = 1.35 \cdot 10^{-6}$, $\sigma_y = 1.76 \cdot 10^{-6}$, and a_i and b_i are the autoregressive parameters for movement in the x and y directions respectively, whose values are given in Figure S1. This autoregressive process was initialized with a sequence of zeros. The values of $x(t)$ at times different from the 1 ms discretized times were defined as the linear interpolation between the values at the discrete time points.

We simulated spiking activity from neural insensity models of the form:

$$\lambda^i(t|H_t) = \exp\left\{\alpha^i + \beta^i|v(t+150\text{ms})|\cos\left(\phi(t+150\text{ms}) - \phi^i_{\text{pref}}\right) + \sum_{\tau=1}^{130\text{ms}} \gamma^i_\tau \Delta N^i_{[t-\tau, t-(\tau-1))}\right\}, \tag{S.17}$$

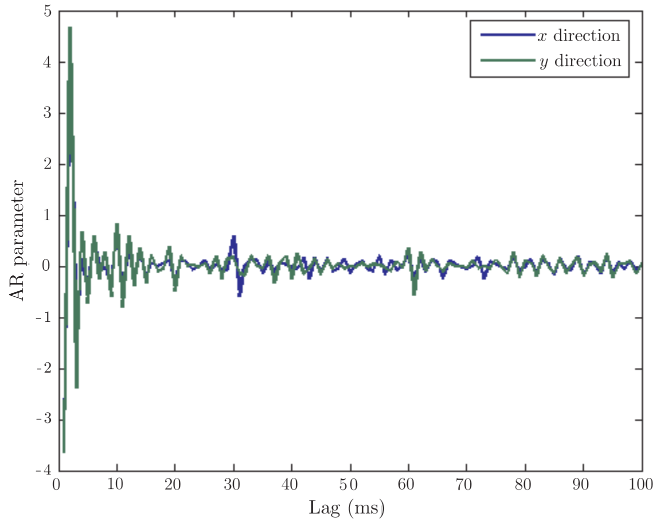

Figure S1. Autoregressive parameters used to generate smooth arm trajectories.

where as before, $v(t)$ and $\phi(t)$ are the speed and direction of the arm movement, $\Delta N^i_{[a,b)}$ is the total number of spikes fired by the ith neuron in the interval $[a, b)$, and α^i, β^i, ϕ^i_{pref}, $\{\gamma^i_\tau\}^{130}_{\tau=1}$ are the model parameters related to the baseline firing rate, velocity modulation, preferred direction and history dependence of the neuron. This form of kinematic tuning is based on a model by Moran and Schwartz (1999).

We simulated 20 neurons generated by selecting α^i, β^i, ϕ^i_{pref}, $\{\gamma^i_\tau\}^{130}_{\tau=1}$ randomly over a range of realistic parameter values identified during previous analyses. Specifically, $\alpha^i \sim N(5, 1)$, $\beta^i \sim N(0.05, 0.0001)$, $\phi^i_{\text{pref}} \sim U([-\pi, \pi])$, and $\gamma^i_\tau \sim N(\hat{\gamma}\tau, .01)$, where $\hat{\gamma}_\tau$ are mean history parameters taken from Figure 2B of Truccolo, Eden, Fellows, Donoghue and Brown (2005). For each such neuron, spikes were generated by drawing a sample from an exponential distribution, and time-rescaling using the conditional intensity $\lambda^i(t|H_t)$ (Brown, E. N., Barbieri, R., Ventura, V., Kass R. E. and Frank L. M. (2001)) to compute the simulated interspike intervals (ISIs).

Using this simulated spiking activity, we constructed point process likelihoods for each neuron, and estimated each parameter by maximum likelihood. The neural intensity models of the form of equation S.17 fall within the class of generalized linear models (GLM) allowing for straightforward estimation using a method such as iteratively reweighted least squares. Figure S2B shows the resulting maximum likelihood estimates of the spatial parameters from 120 sec. of simulated spiking activity from each neuron. The estimated receptive

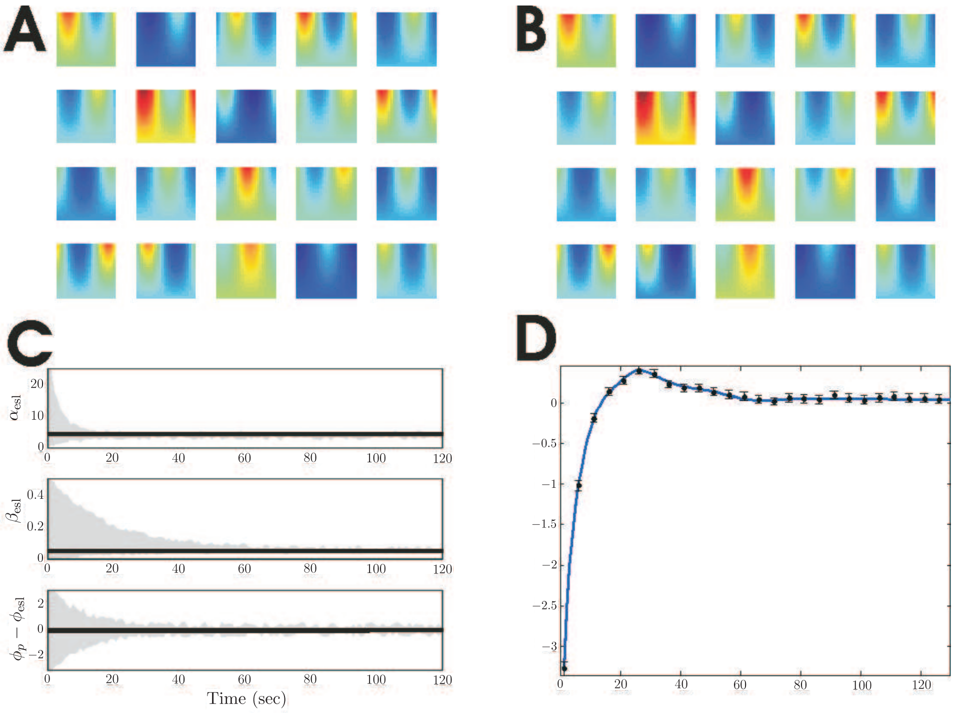

Figure S2. Encoding analysis. (A) True receptive field properties for each of the 20 simulated neurons as a function of arm direction (x-ordinate) and speed (y-ordinate). (B) Maximum likelihood estimates of receptive fields for each simulated neuron based on spiking activity over 120 sec. (C) Convergence of spatial receptive field parameter estimates as a function of amount of time observed. Black line represents true parameter values, gray region represents 95% confidence region of encoding estimates generated by simulating data 1,000 times. (D) True (blue line) and estimated history parameters from neuron 1, based on 120 sec of spiking observations.

field properties are nearly identical to their true values. Figure S2C shows how these estimates converge to their true values as a function of the amount of data observed for a single neuron. Here we simulated 1,000 trials of spiking activity for the same neurons and constructed 95% confidence regions for the resulting estimates, shown in gray. Within 60 seconds, these confidence regions converge nearly exactly to the true values for each of these parameters. The estimates of the history parameters take longer to estimate accurately. Figure S2D shows the true values of these parameters and the maximum likelihood estimates following 20 minutes of simulated spiking data.

Figure S3. illustrates the results of a goodness-of-fit analysis on the estimated neural intensity models. The spike times for each neuron over 120 seconds of

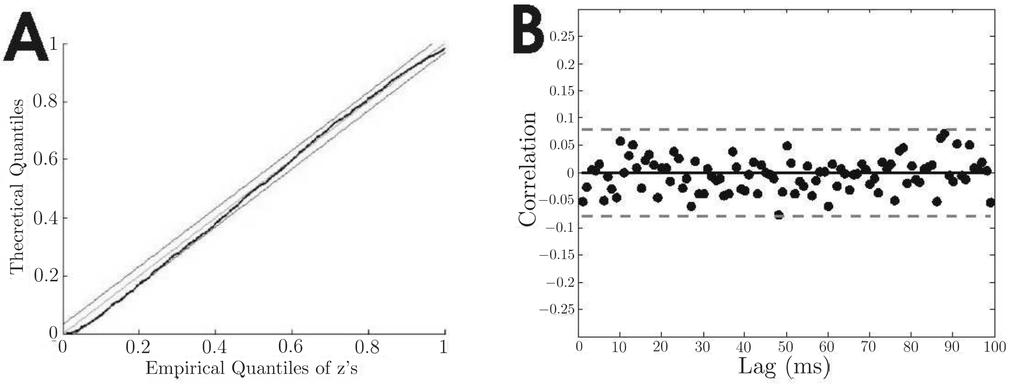

Figure S3. Illustration of goodness-of-fit analysis for first estimated neural receptive field. Spike times over 120 seconds of movement were rescaled according to the estimated intensity function. A) KS plot of rescaled empirical CDF versus exponential CDF. The plot remains within the 95% confidence interval for the KS statistic at all times. B) Correlation coefficient of rescaled times as a function of spike lag.

movement were rescaled according to their estimated neural intensity. Figure S3A shows a KS plot for the first neuron. The KS plot remains within its 95% confidence interval, suggesting that the estimated models are able to accurately describe the statistical structure in the observed spiking activity. Figure S3B shows that this rescaled sequence of spike times are uncorrelated across many lags, suggesting that the estimated neural intensity model is able to disentangle history dependent correlations in the spiking activity. Examining all 20 simulated neurons in this manner, we found that 18/20 fit completely within the 95% KS confidence bounds, suggesting that this estimation method is appropriate across all the simulated data.

Next, we used the continuous-time approximate filter in equations S.12 and S.13 to estimate an arm trajectory from the simulated spike data. We adopt a state space model for the arm movement trajectory as in equation 3.1, with $x(t) = \begin{bmatrix} v(t)\cos(\phi(t)) \\ v(t)\sin(\phi(t)) \end{bmatrix}$, $A = 0$ and $B = 10 \cdot I$. The initial value for the mean estimator was drawn randomly from $\hat{x}(0) \sim N(0, 4 \cdot I)$, and the initial variance estimator was set to $4 \cdot I$.

The filtering equations were solved numerically using an Euler's method at a time step of 1 msec, for non-spike intervals, and computing the jumps directly at the spike times. Notice that the state equation used to construct the trajectory estimate is different from the model used to generate the arm trajectories, given by equation S.16. Whereas it would be possible to improve the decoded estimates by using the true model generating the data, in real neural systems no such model will be known. We use the simple state model above to construct the filter estimates to emulate this situation. The accuracy of the resulting decoded trajectories suggests that the filter works well even with this model misspecification.

Department of Mathematics and Statistics, 111 Cummington St., Boston University, Boston, MA 02215, U.S.A.

E-mail: tzvi@bu.edu

Department of Brain and Cognitive Sciences, Harvard/MIT Division of Health Sciences and Technology, Massachusetts Institute of Technology, Cambridge, MA, U.S.A.

E-mail: brown@neurostat.mgh.harvard.edu

(Received April 2007; accepted March 2008))