Statistical methods for understanding neural computation

Liam Paninski

Department of Statistics and Center for Theoretical Neuroscience
Columbia University
http://www.stat.columbia.edu/∼liam
liam@stat.columbia.edu

June 24, 2009

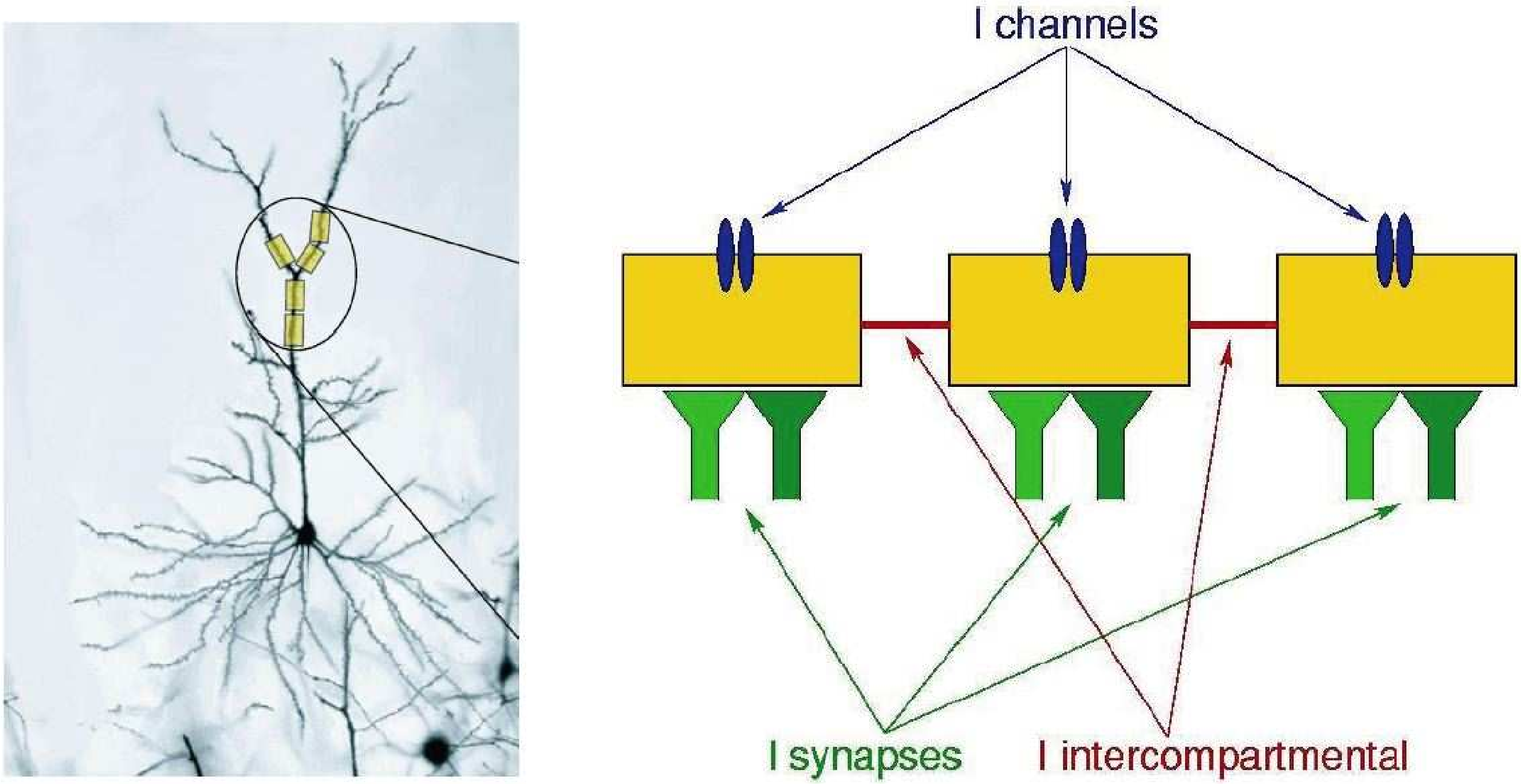
Support: NIH CRCNS, Sloan Fellowship, NSF CAREER, McKnight Scholar award.

Some exciting open challenges for statistical neuroscience

- inferring biophysical neuronal properties from noisy recordings
- reconstructing the full dendritic spatiotemporal voltage from noisy, subsampled observations
- estimating subthreshold voltage given superthreshold spike trains
- extracting spike timing from slow, noisy calcium imaging data
- reconstructing presynaptic conductance from postsynaptic voltage recordings
- inferring connectivity from large populations of spike trains
- decoding behaviorally-relevant information from spike trains
- optimal control of neural spike timing

— to solve these, we need to combine the two classical branches of computational neuroscience: dynamical systems and neural coding

An inverse problem: inferring cable equation parameters



Can we recover detailed biophysical properties?

- Active: membrane channel densities
- Passive: axial resistances, “leakiness” of membranes
- Dynamic: spatiotemporal synaptic input

Estimating biophysical parameters from $V(x,t)$

$$C\frac{dV_i}{dt} = I_i^{\text{channels}} + I_i^{\text{synapses}} + I_i^{\text{intercompartmental}}$$

$$I_i^{\text{channels}} = \sum_c \bar{g}_c g_c(t)(E_c - V_i(t))$$

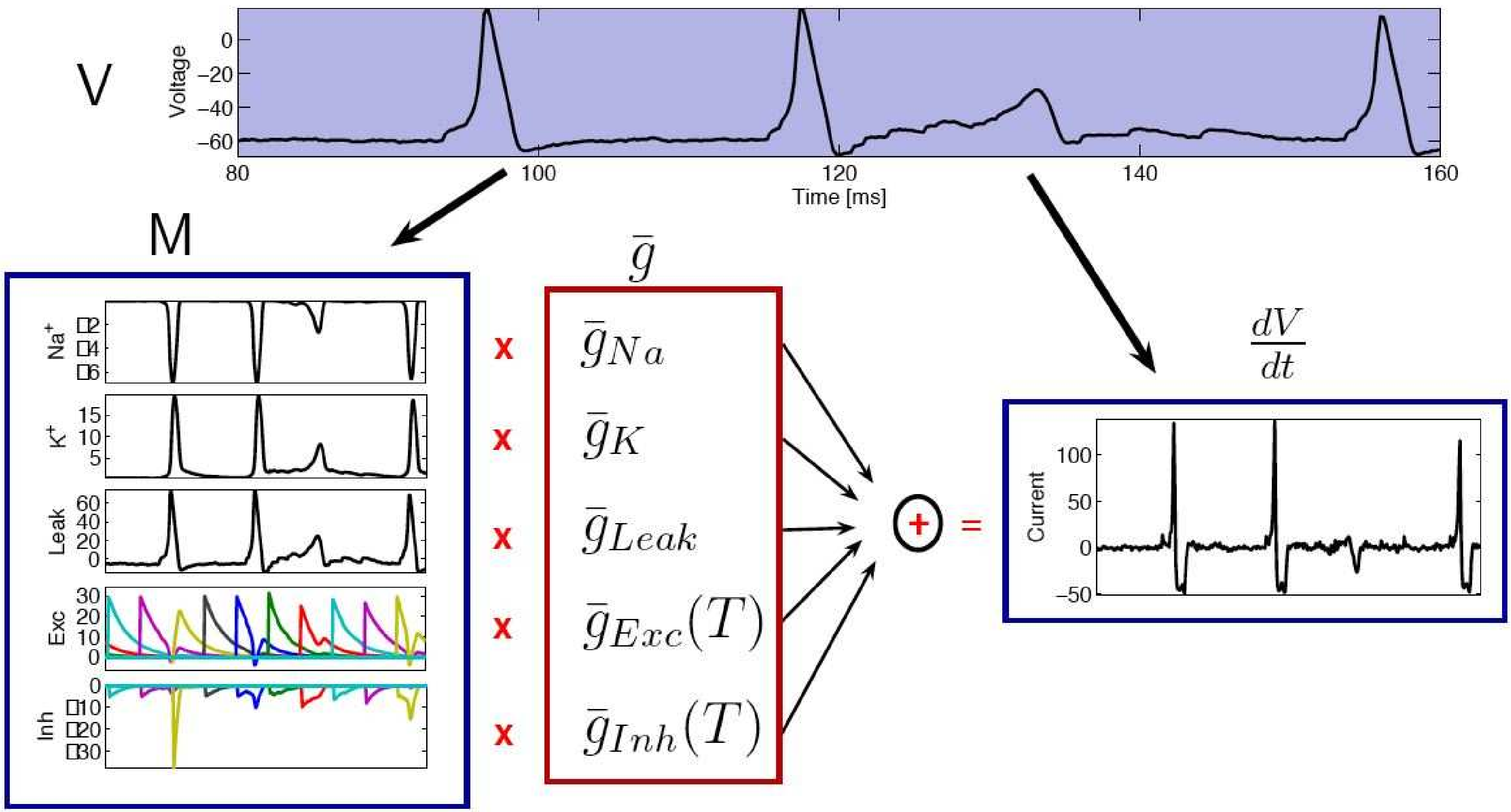
$$I_i^{\text{synapses}} = \sum_s (\xi_s * k_s)(t)(E_s - V_i(t))$$

$$I_i^{\text{intercompartmental}} = \sum_a g_a \Delta V_a(t)$$

Key point: **if** we observe full $V_i(t)$ + cell geometry, channel kinetics known + current noise is Gaussian,

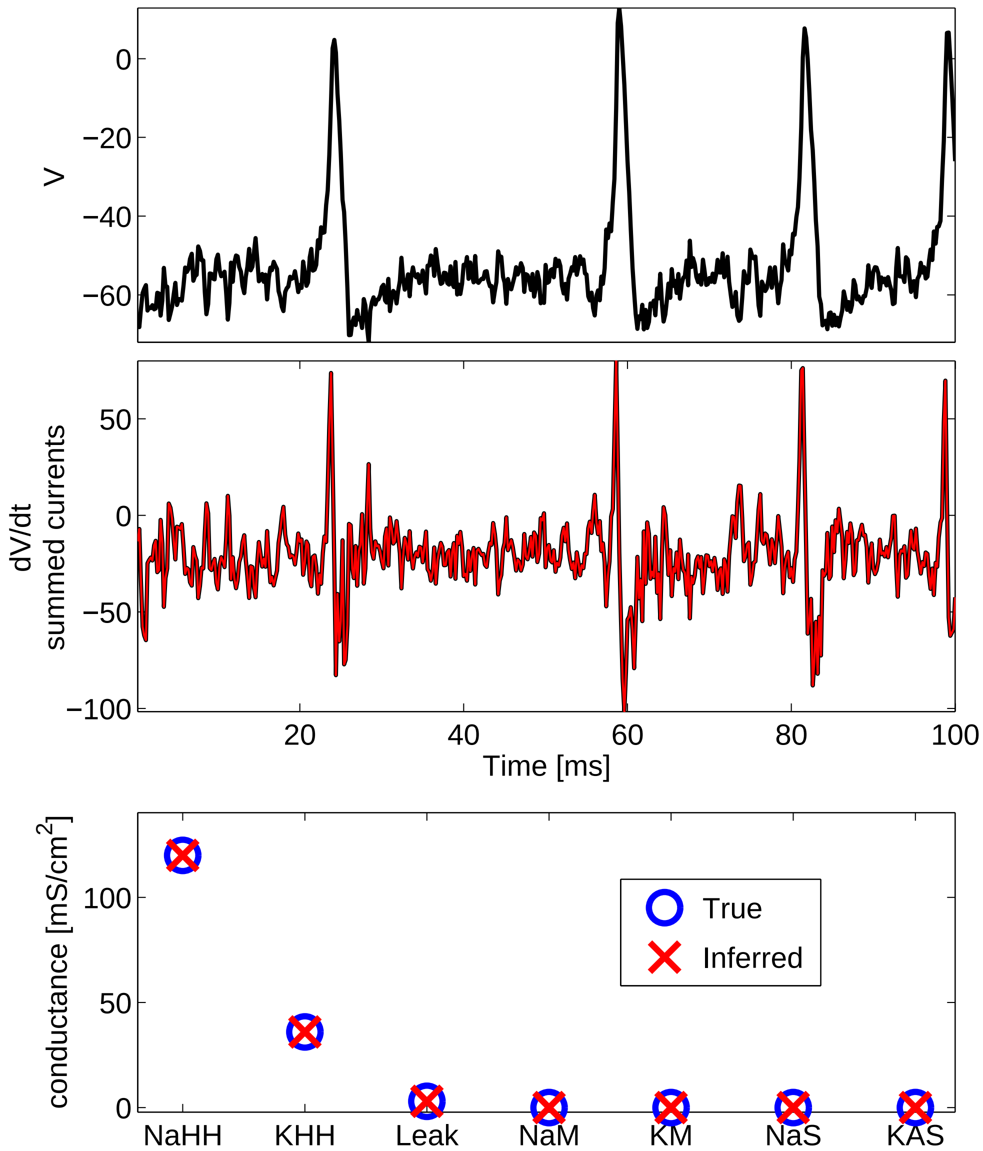
then estimating unknown parameters is standard convex nonnegative regression problem (albeit high-d): $\min_{\theta \geq 0} ||Y - X\theta||^2$.

Estimating channel densities from $V(t)$



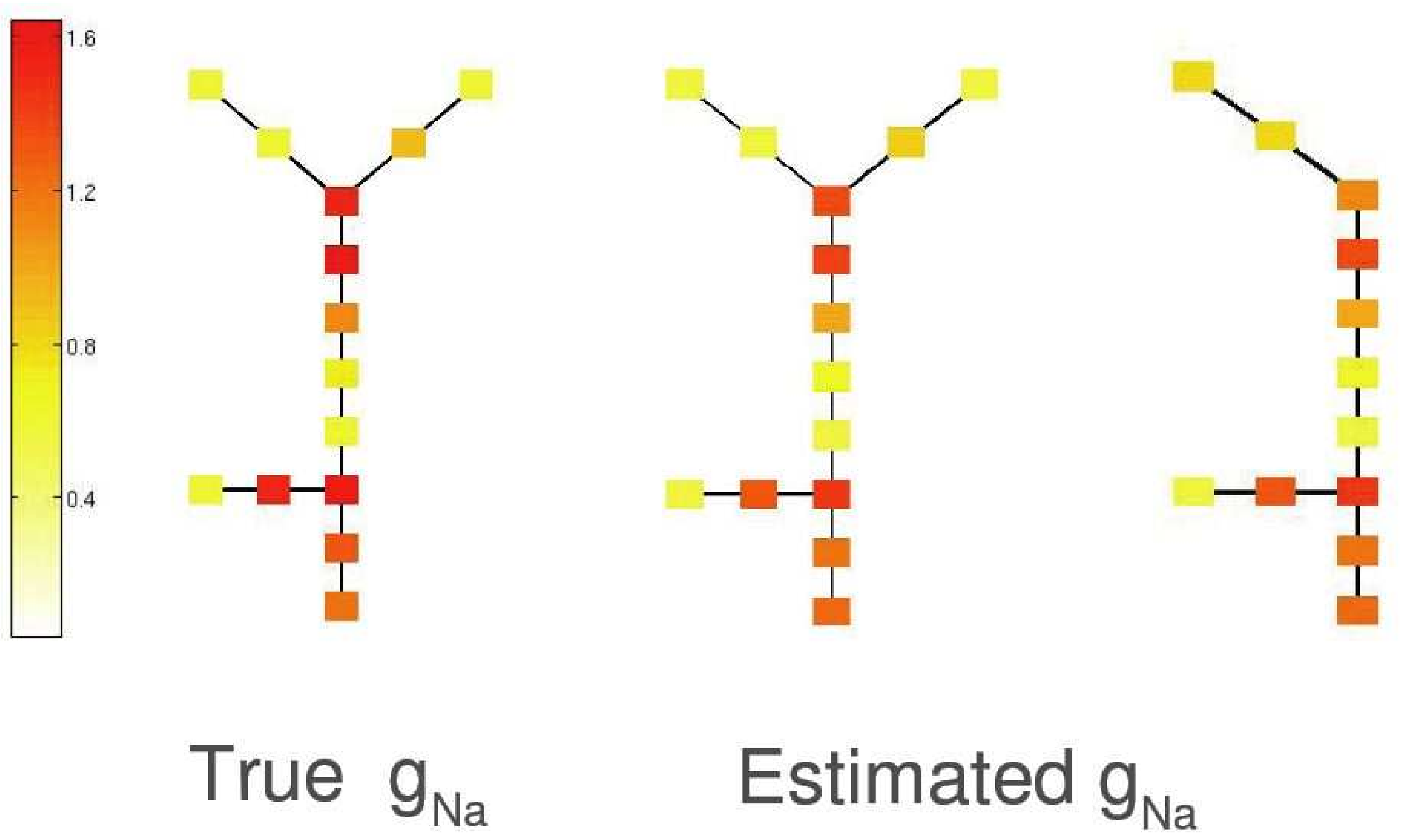
(Huys et al., 2006)

Estimating channel densities from $V(t)$



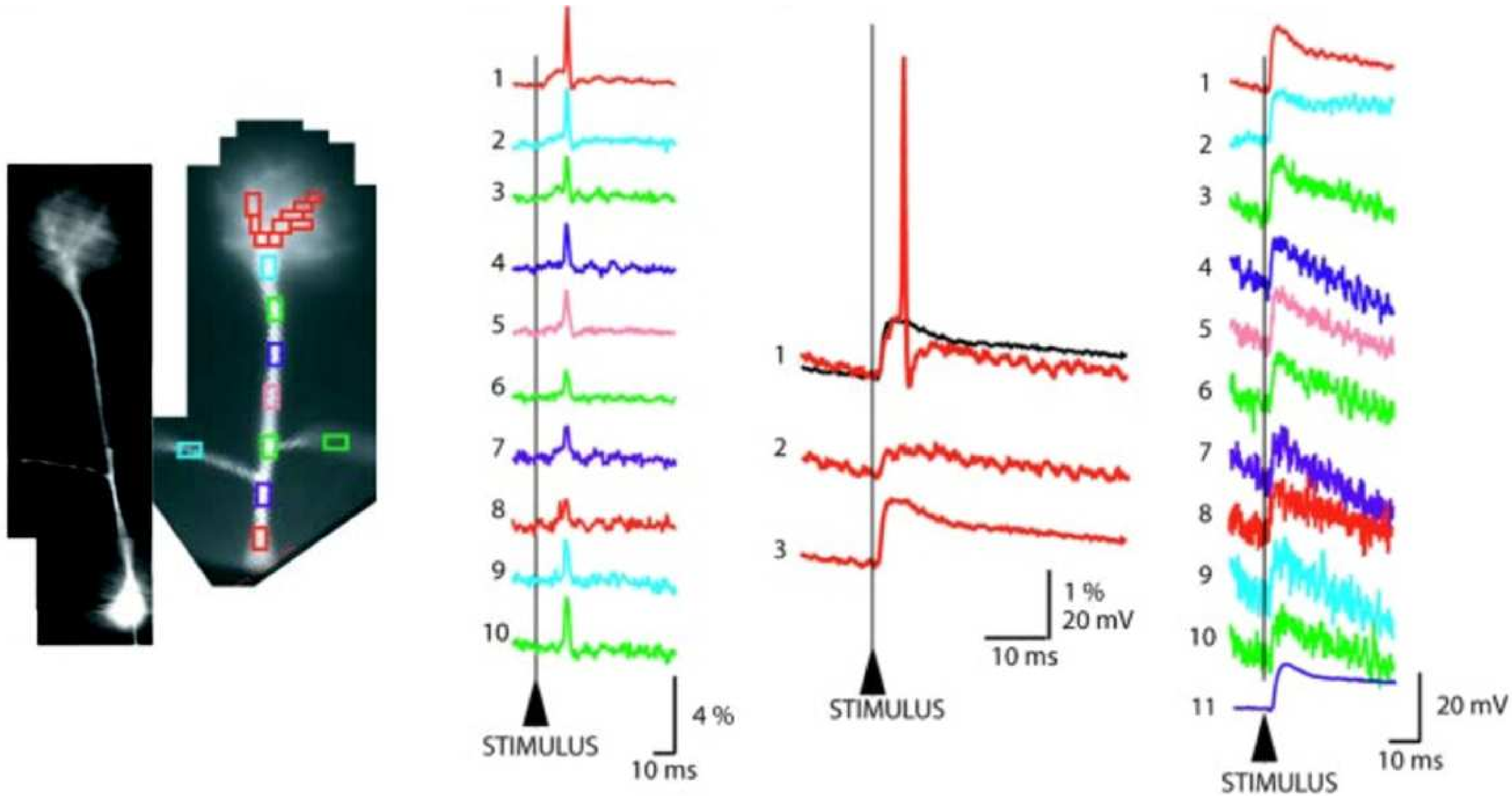
Estimating non-homogeneous channel densities

$$I_i^{\text{channels}} = \sum_c \bar{g}_c g_c(t)(E_c - V_i(t))$$



The filtering problem

Spatiotemporal imaging data is very exciting, but we have to deal with noise and intermittent observations.



(Djurisic et al., 2004; Knopfel et al., 2006)

Basic paradigm: the Kalman filter

Variable of interest, q_t, evolves according to a noisy differential equation (Markov process):

$$dq/dt = f(q_t) + \epsilon_t.$$

Make noisy observations:

$$y_t = g(q_t) + \eta_t.$$

We want to infer $E(q_t|Y)$: optimal estimate given observations. We also want errorbars: $Var(q_t|Y)$ quantifies how much we actually know about q_t.

If $f(.)$ and $g(.)$ are linear, and ϵ_t and η_t are Gaussian, then solution is classical: Kalman filter.

The forward recursion

We want $p(q_t|Y_{1:t}) \propto p(q_t, Y_{1:t})$. We know that

$$p(Q,Y) = p(Q)p(Y|Q) = p(q_1)\left(\prod_{t=2}^{T} p(q_t|q_{t-1})\right)\left(\prod_{t=1}^{T} p(y_t|q_t)\right)$$

To compute $p(q_t, Y_{1:t})$ recursively, just write out marginal and pull out constants from the integrals:

$$\begin{aligned} p(q_t, Y_{1:t}) &= \int_{q_1}\int_{q_2}\dots\int_{q_{t-1}} p(Q_{1:t}, Y_{1:t}) = \int_{q_1}\int_{q_2}\dots\int_{q_{t-1}} p(q_1)\left(\prod_{i=2}^{t} p(q_i|q_{i-1})\right)\left(\prod_{i=1}^{t} p(y_i|q_i)\right) \\ &= p(y_t|q_t)\int_{q_{t-1}} p(q_t|q_{t-1})p(y_{t-1}|q_{t-1})\int_{q_{t-2}}\dots\int_{q_2} p(q_3|q_2)p(y_2|q_2)\int_{q_1} p(q_2|q_1)p(y_1|q_1)p(q_1). \end{aligned}$$

So, just recurse

$$p(q_t, Y_{1:t}) = p(y_t|q_t)\int_{q_{t-1}} p(q_t|q_{t-1})p(q_{t-1}, Y_{1:t-1}).$$

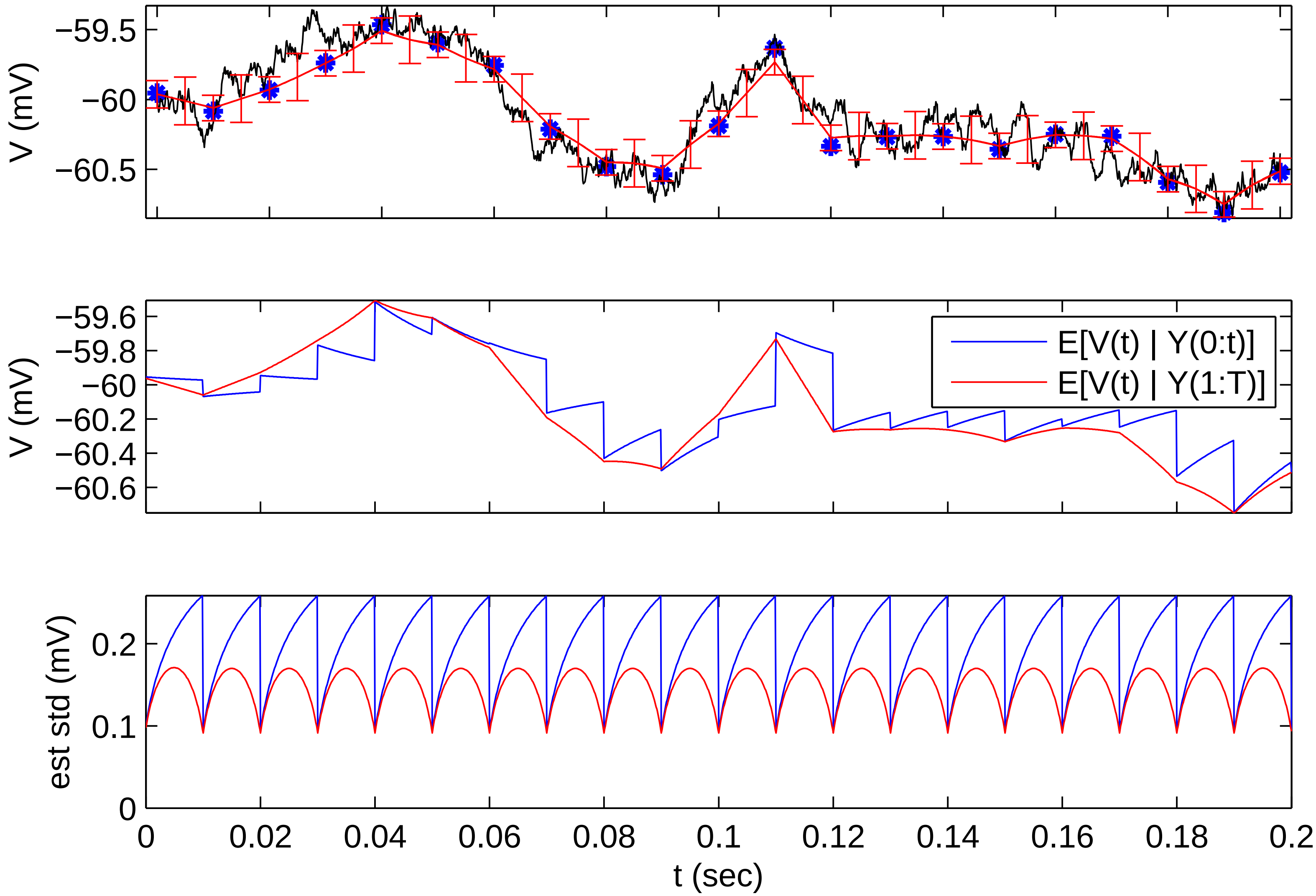
Linear-Gaussian case: requires $O(\dim(q)^3 T)$ time; just matrix algebra.

Approximate solutions in more general case, e.g., Gaussian approximations (Brown et al., 1998), or Monte Carlo ("particle filtering").

Key point: efficient recursive computations $\implies O(T)$ time.

Application: incomplete observations of $V(t)$

— Leaky integrator model: $dV/dt = g_l[V_l - V(t)] + \epsilon_t$



Multicompartmental case

Easy extension of Kalman method:

$$d\vec{V}/dt = A\vec{V}(t) + \vec{\epsilon}_t$$

$$\vec{y}(t) = B\vec{V}(t) + \vec{\eta}_t$$

Example:

$V_i(t)$ = voltage at compartment i

A = dynamics matrix (cable equation): includes leak ($A_{ii} = -g_l$) and inter-compartmental terms ($A_{ij} = 0$ for non-adjacent compartments)

B = observation matrix

Example: laser scanning

$B = B_t =$ single-node snapshot

(Loading hp07-KalmanSmootherMovie.mov)

(Huys and Paninski, 2009)

Example: multiple observations

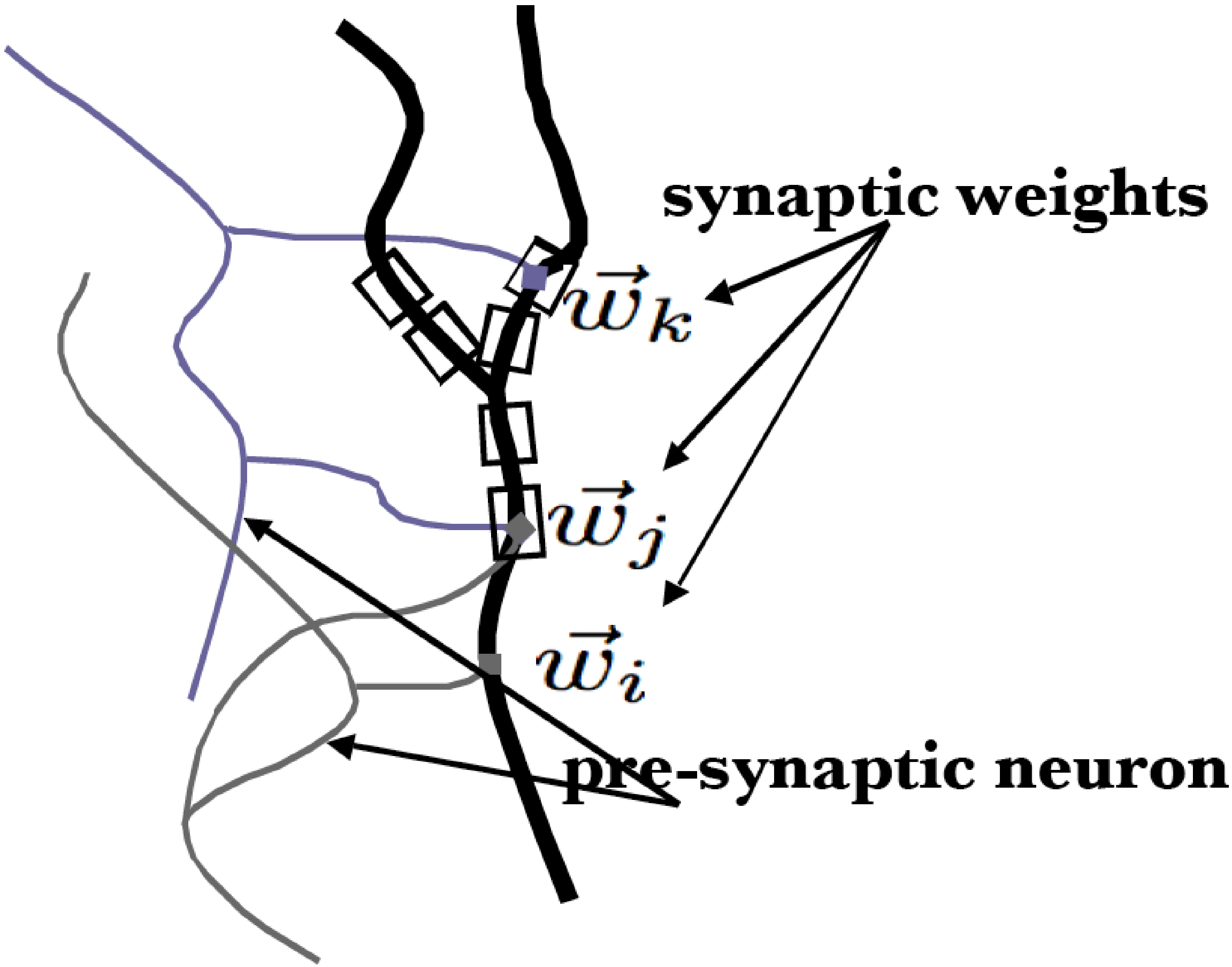
(Loading low-rank-speckle.mp4)

— special methods required to deal with large dendritic trees: $\dim(q_t)$ is very large (Paninski, 2009a).

Example: summed observations

(Loading low-rank-horiz.mp4)

Application: synaptic locations/weights



Application: synaptic locations/weights

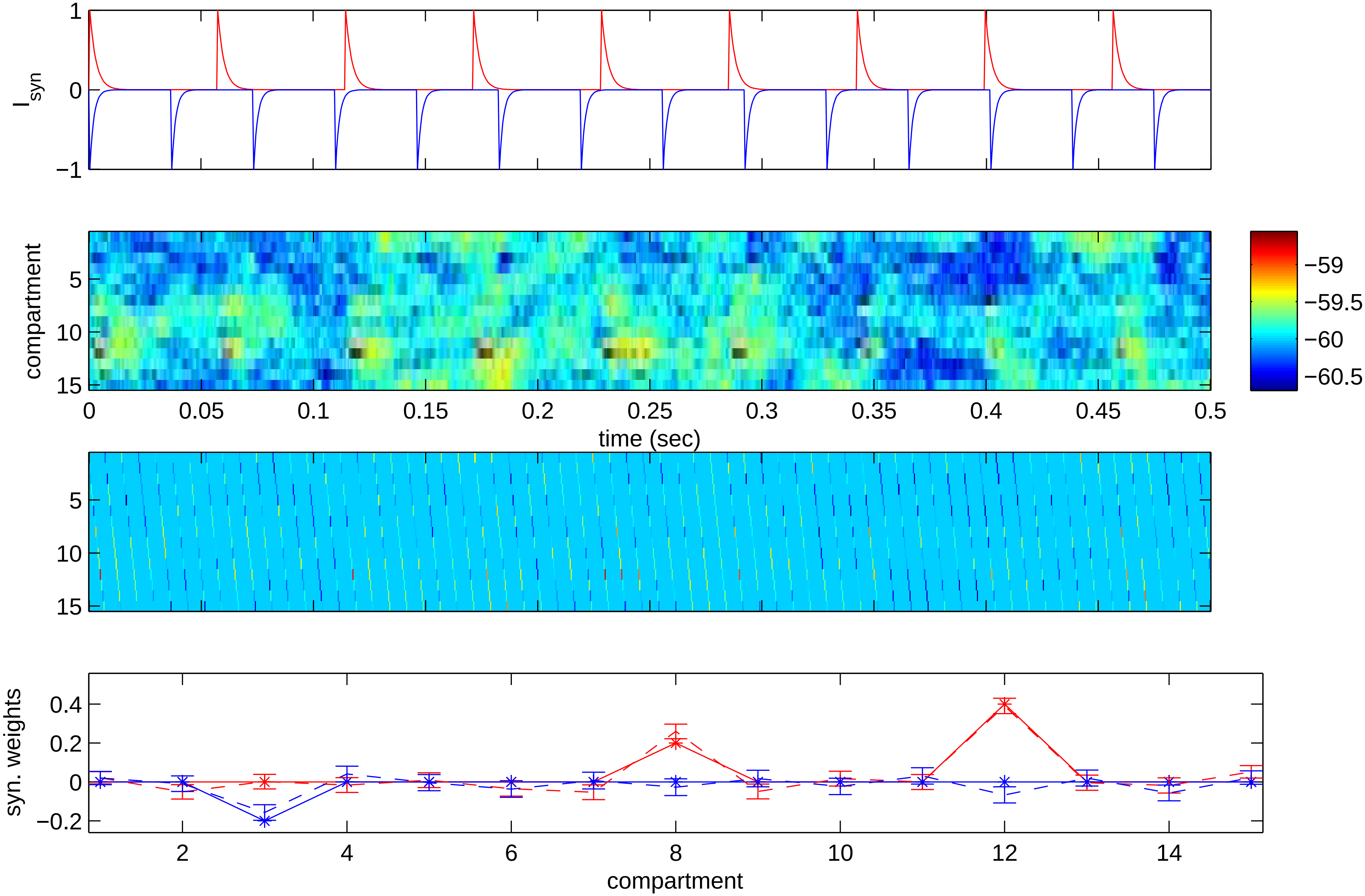
Including known terms:

$$d\vec{V}/dt = A\vec{V}(t) + W\vec{U}(t) + \vec{\epsilon}(t)$$

$U_j(t)$ = known input terms

Example: $U(t)$ are known presynaptic spike times, and we want to detect which compartments are connected (i.e., infer the weight matrix W).

Detecting synapses



(Paninski and Ferreira, 2008; Paninski et al., 2009)

Another application: neural prosthetics

q_t: hand position (red square); $E(q_t|Y_{1:t})$: green circle

y_t: vector of observed spike counts at time t from multiple simultaneously recorded motor cortical neurons

(Loading Kalman-neural-decoding.mp4)

(Wu et al., 2006; Wu et al., 2009)

Another look: computing the MAP path

We often want to compute the MAP estimate

$$\hat{Q} = \arg\max_Q p(Q|Y).$$

In standard Kalman setting, forward-backward recursions also compute MAP (because $E(Q|Y)$ and $\hat{Q}$ coincide if $p(Q|Y)$ is Gaussian).

More generally, write out the posterior:

$$\begin{aligned}
\log p(Q|Y) &= \log p(Q) + \log p(Y|Q) + const. \\
&= \sum_t \log p(q_{t+1}|q_t) + \sum_t \log p(y_t|q_t) + const.
\end{aligned}$$

Two basic observations:

- If $\log p(q_{t+1}|q_t)$ and $\log p(y_t|q_t)$ are concave, then so is $\log p(Q|Y)$.
- Hessian H of $\log p(Q|Y)$ is block-tridiagonal: $p(y_t|q_t)$ contributes a block-diag term, and $\log p(q_{t+1}|q_t)$ contributes a block-tridiag term.

Now recall Newton's method: iteratively solve $HQ_{dir} = \nabla$. Solving tridiagonal systems requires $O(T)$ time.

— computing MAP by Newton's method requires $O(T)$ time, even in highly non-Gaussian cases.

Constrained optimization

In many cases we need to impose constraints on q_t (e.g., nonnegativity). Easy to incorporate here, via interior-point (barrier) methods:

$$\begin{aligned}\arg\max_{Q \in C} \log p(Q|Y) &= \lim_{\epsilon \searrow 0} \arg\max_{Q} \left\{ \log p(Q|Y) + \epsilon \sum_t f(q_t) \right\} \\ &= \lim_{\epsilon \searrow 0} \arg\max_{Q} \left\{ \sum_t \log p(q_{t+1}|q_t) + \log p(y_t|q_t) + \epsilon f(q_t) \right\};\end{aligned}$$

$f(.)$ is concave and approaching $-\infty$ near boundary of constraint set C. The Hessian remains block-tridiagonal and negative semidefinite for all $\epsilon > 0$, so optimization still requires just $O(T)$ time.

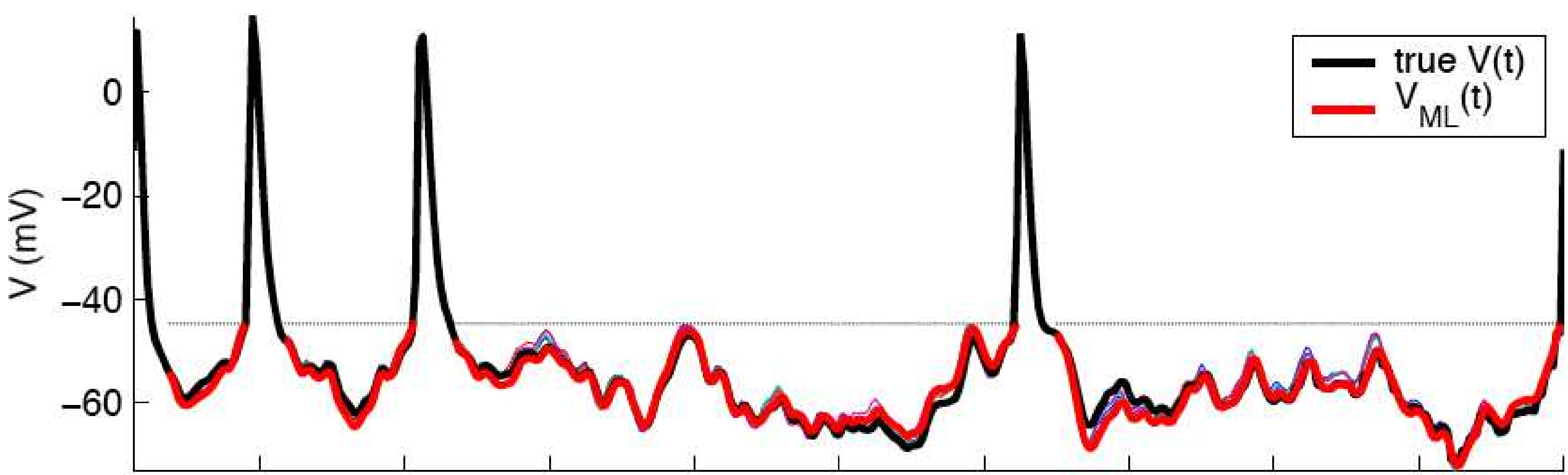
Example: computing the MAP subthreshold voltage given superthreshold spikes

Leaky, noisy integrate-and-fire model:

$$V_{t+dt} = V_t + \left(-\frac{V_t}{\tau} + I_t\right) dt + \sigma\sqrt{dt}\epsilon_t, \;\; \epsilon_t \sim \mathcal{N}(0,1)$$

Observations: $y_t = 0$ (no spike) if $V_t < V_{th}$; $y_t = 1$ if $V_t = V_{th}$

Hard threshold $\implies$ $p(V|Y)$ is very non-Gaussian: "corners" at $V_t = V_{th}$.

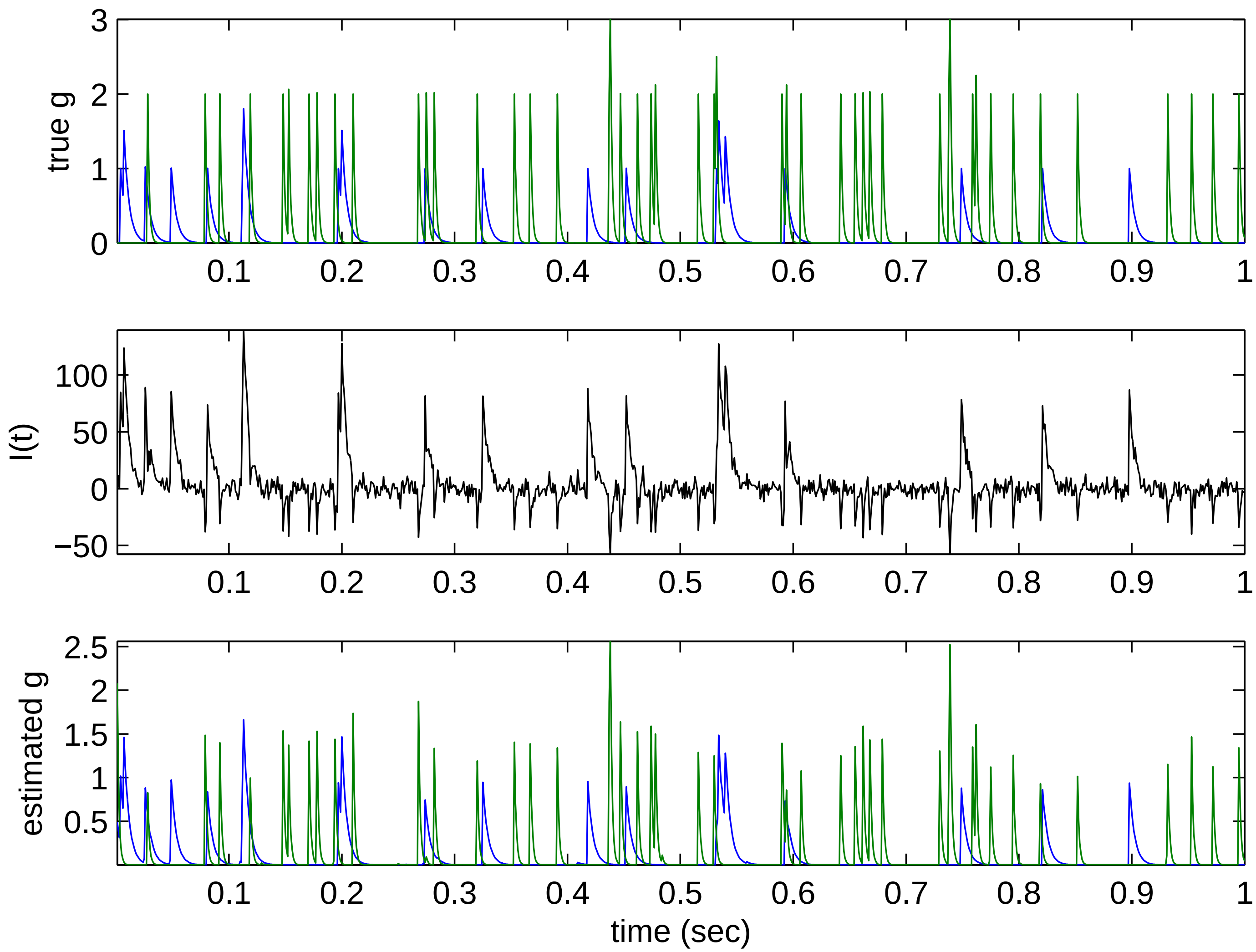


(Paninski, 2006)

Example: inferring presynaptic input

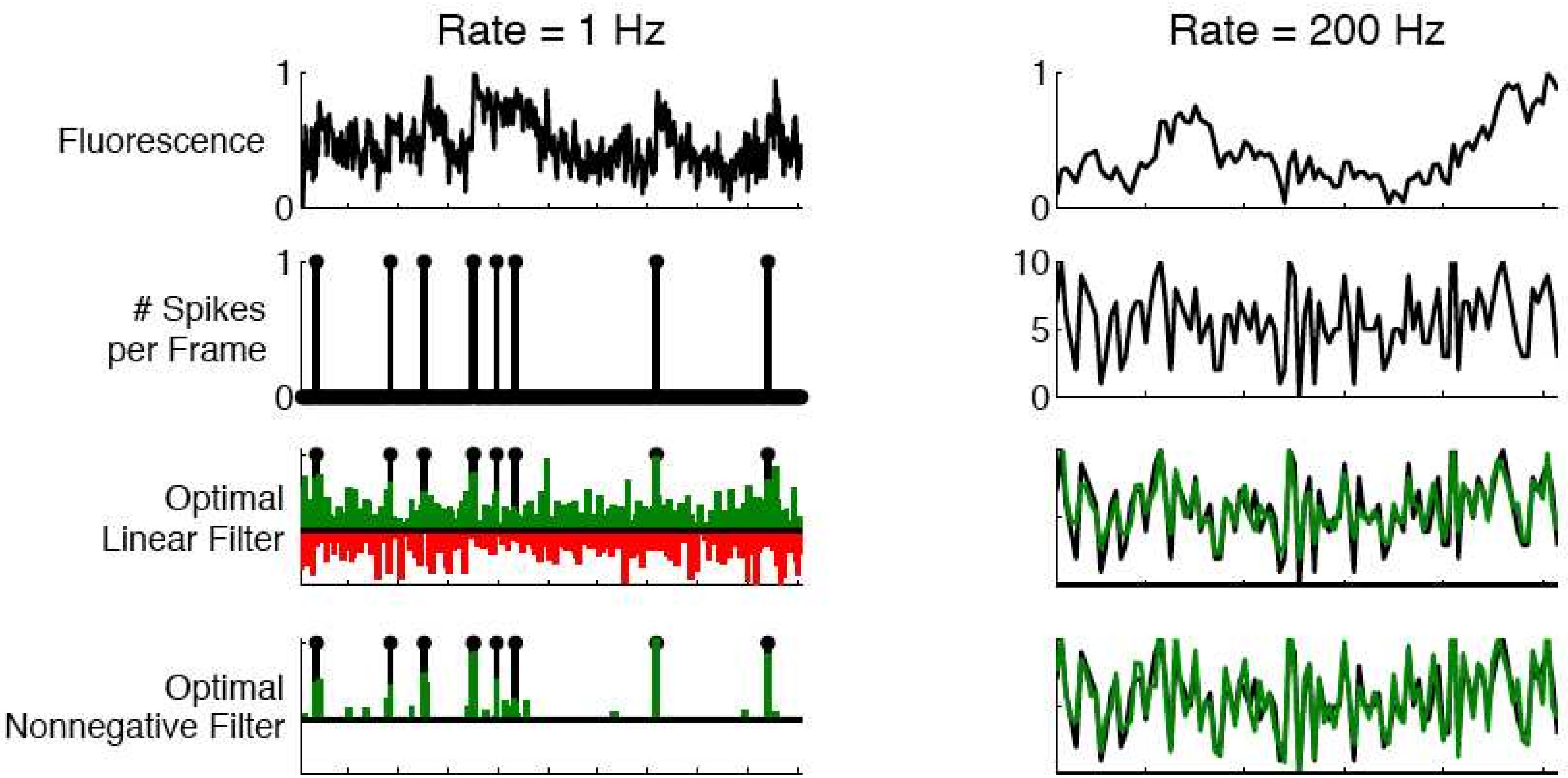
$$
\begin{aligned}
g_j(t+dt) &= g_j(t) - dt g_j(t)/\tau_j + N_j(t), \;\; N_j(t) \geq 0 \\
y_t &= I_t = \sum_j g_j(t)(V_j - V_t) + \epsilon_t
\end{aligned}
$$

Hidden state q_t: vector of conductances g_t (Paninski, 2009b)



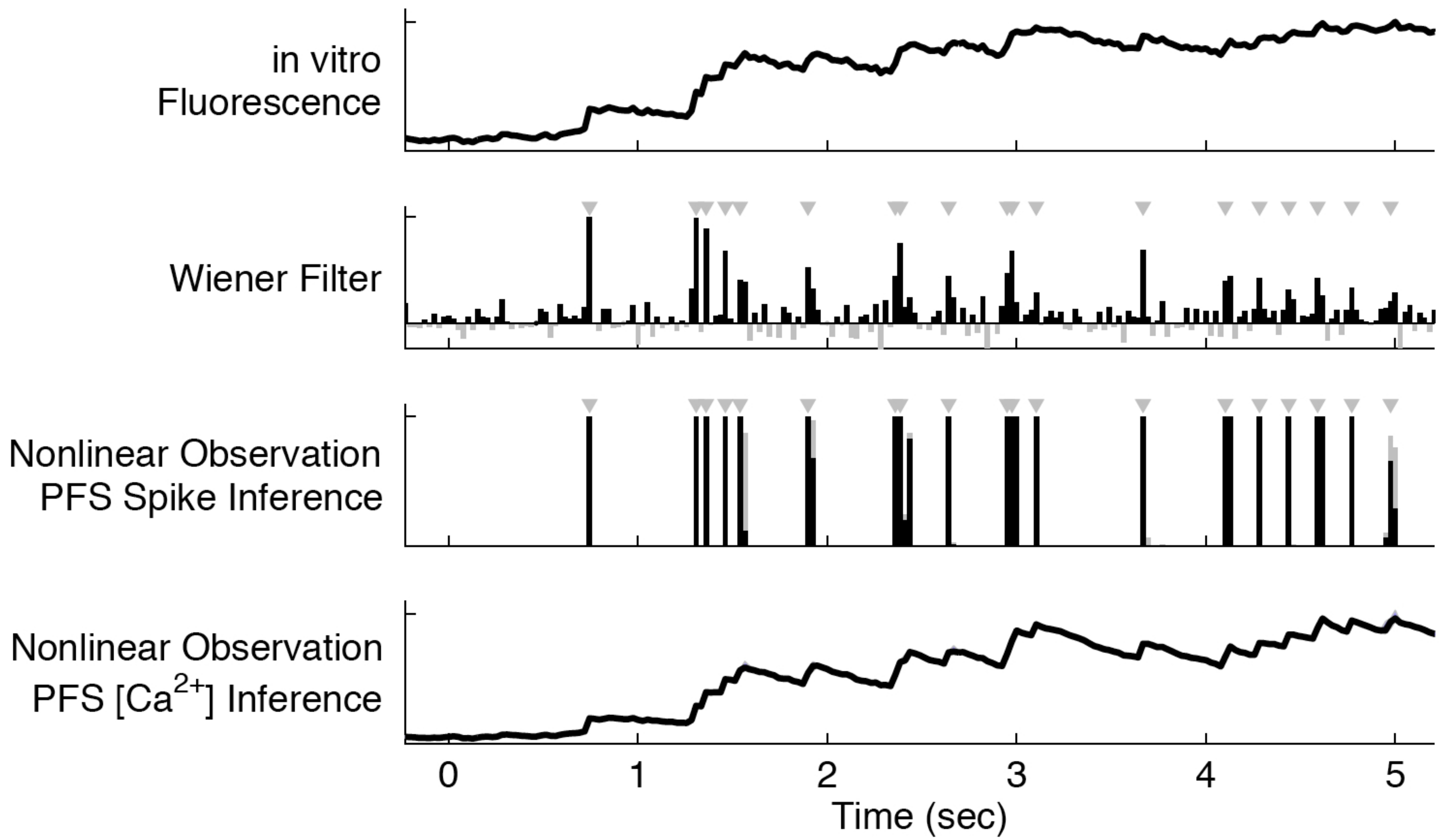
Example: inferring spike times from slow, noisy calcium data

$$C_{t+dt} = C_t - dtC_t/\tau + N_t;\ N_t > 0;\ y_t = C_t + \epsilon_t$$



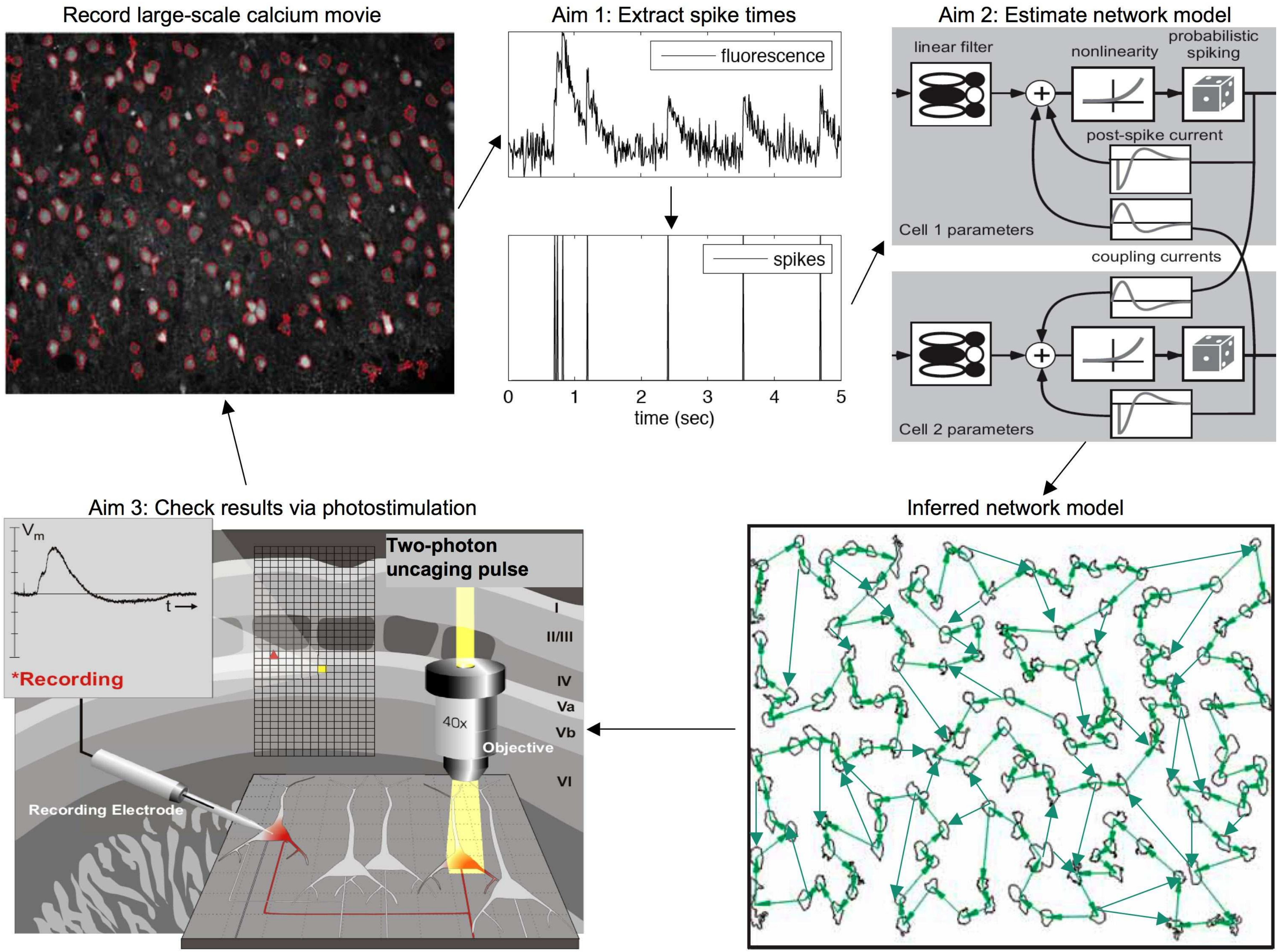
— nonnegative deconvolution is a recurring problem in signal processing (Vogelstein et al., 2008a).

Particle filter can extract spikes from saturated recordings



(Vogelstein et al., 2008b)

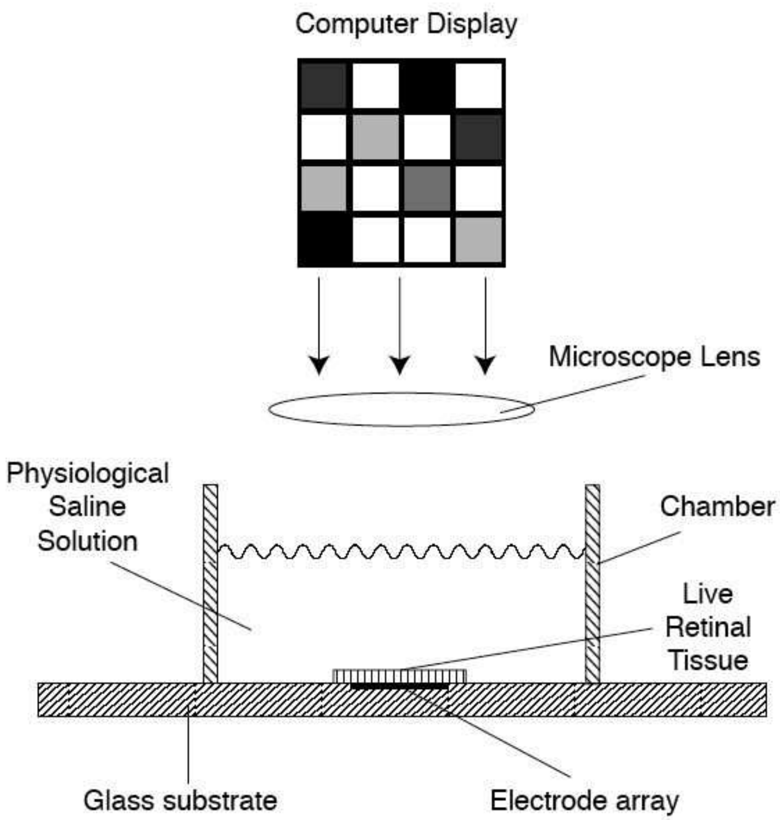
Next challenge: circuit inference

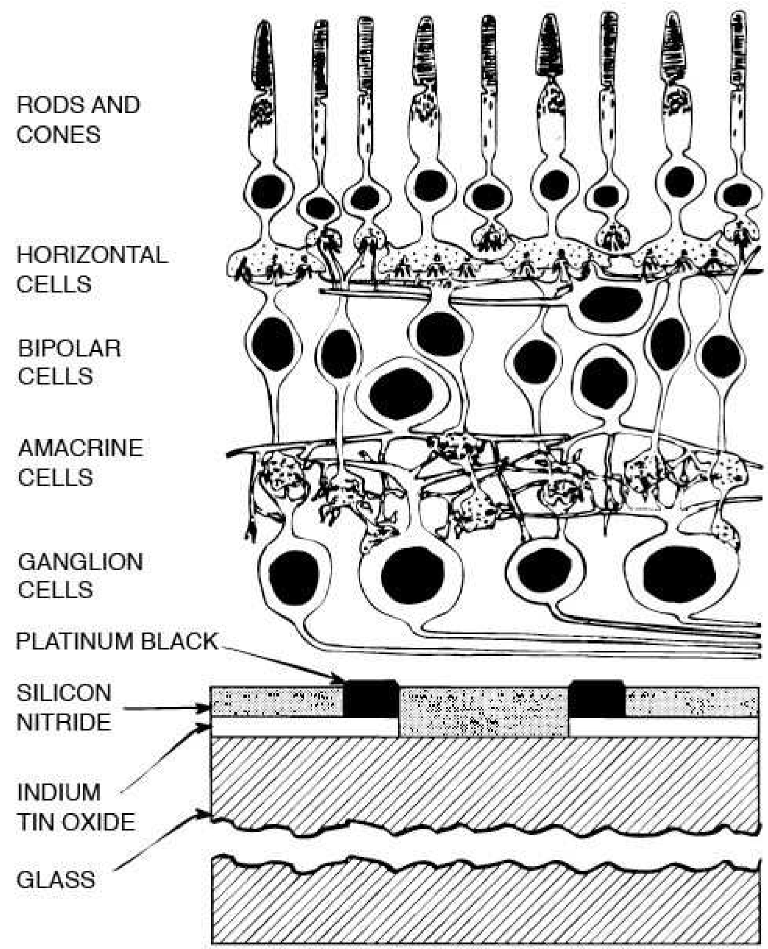


Part 2: modeling spike train data

Preparation: dissociated macaque retina (Chichilnisky lab)

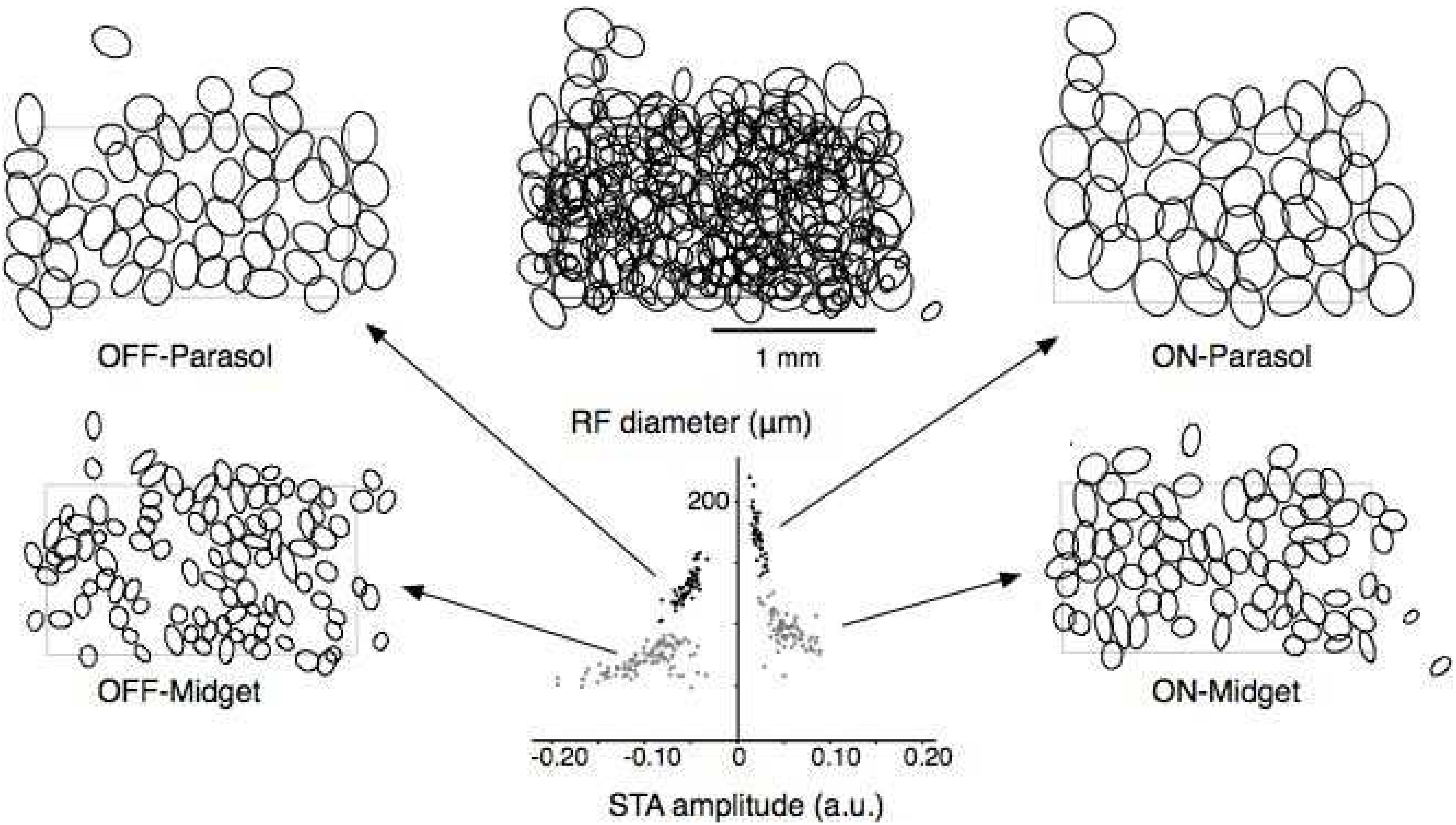
— extracellularly-recorded responses of populations of RGCs



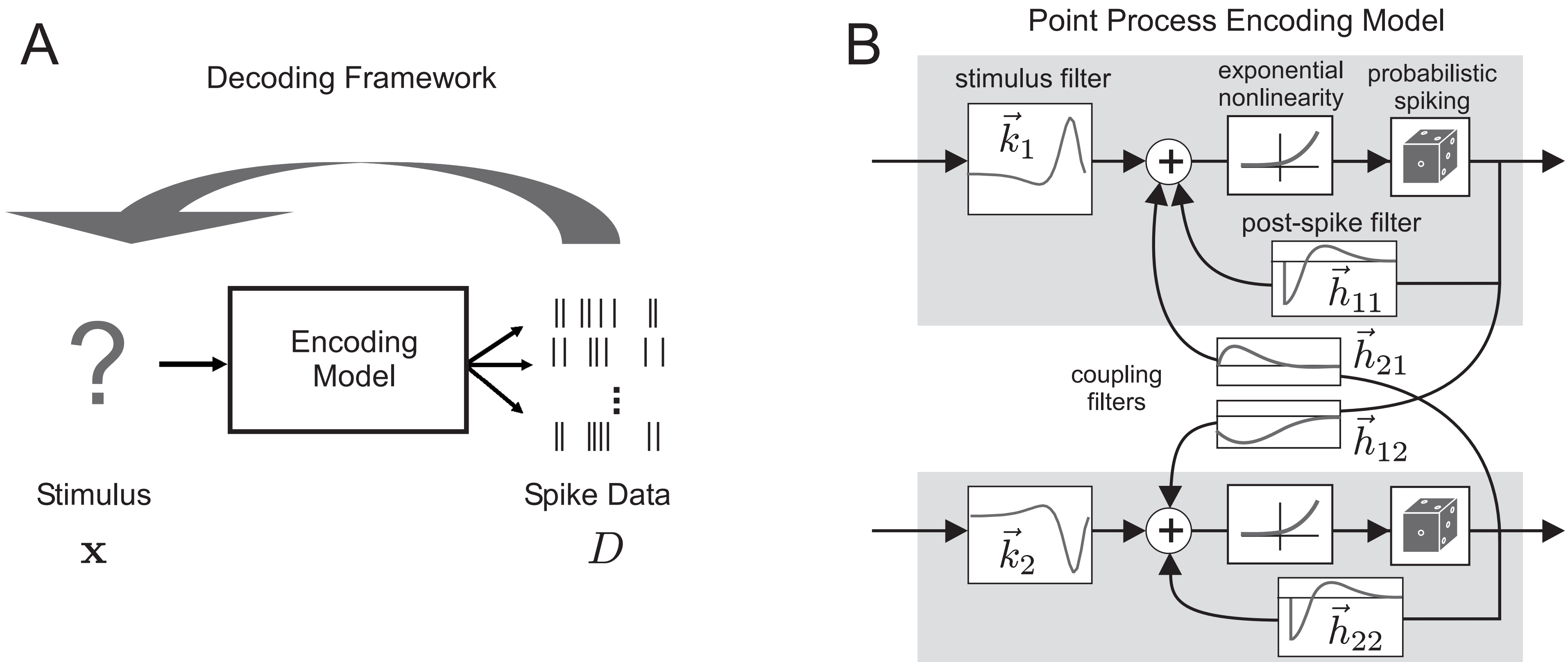


Stimulus: random spatiotemporal visual stimuli (Pillow et al., 2008)

Receptive fields tile visual space



Multineuronal point-process model



$$\lambda_i(t) = f\left(b_i + \vec{k}_i \cdot \vec{x}(t) + \sum_{i',j} h_{i',j} n_{i'}(t-j)\right),$$

(Paninski et al., 2007)

Point-process likelihood

$$\lambda_t = f(X_t\theta)$$

$$\log p(n_t|X_t,\theta) = \log Poiss(n_t;\lambda_t dt) = -f(X_t\theta)dt + n_t \log f(X_t\theta) + const$$

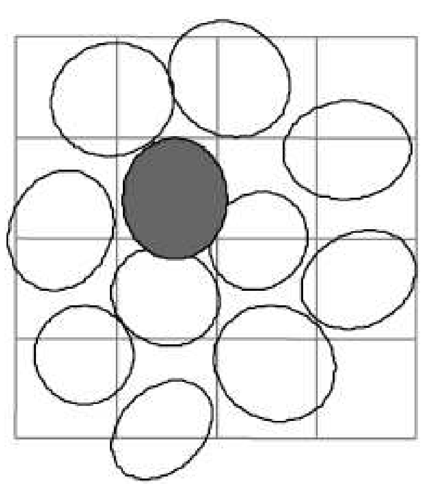
$$\log p(\{n_t\}|X,\theta) = \sum_t \log p(n_t|X_t,\theta).$$

Key points:

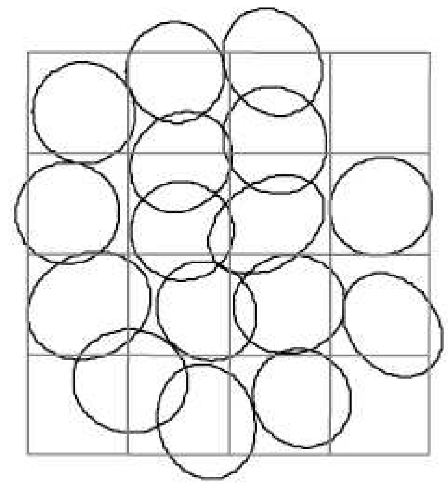
- f convex and log-concave $\implies$ log-likelihood concave in θ. Easy to optimize, so estimating θ is very tractable (Paninski, 2004; Truccolo et al., 2005).

- Easy to include priors $p(\theta)$ if $\log p(\theta)$ is concave: useful for smoothing/sparsening estimates

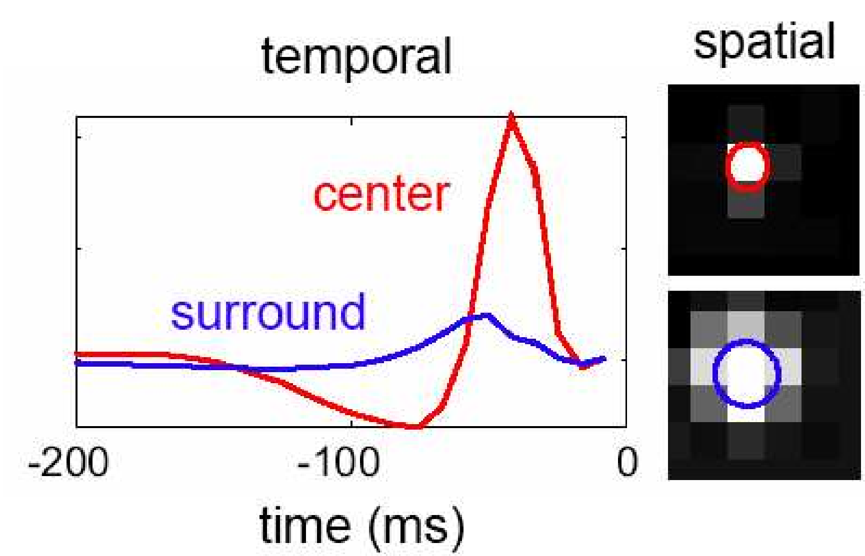
ON cells



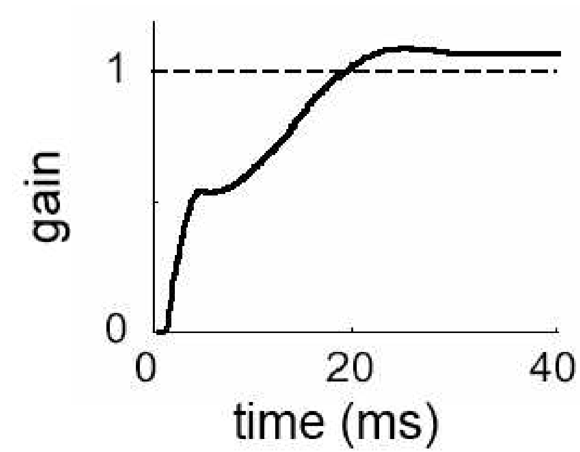
OFF cells



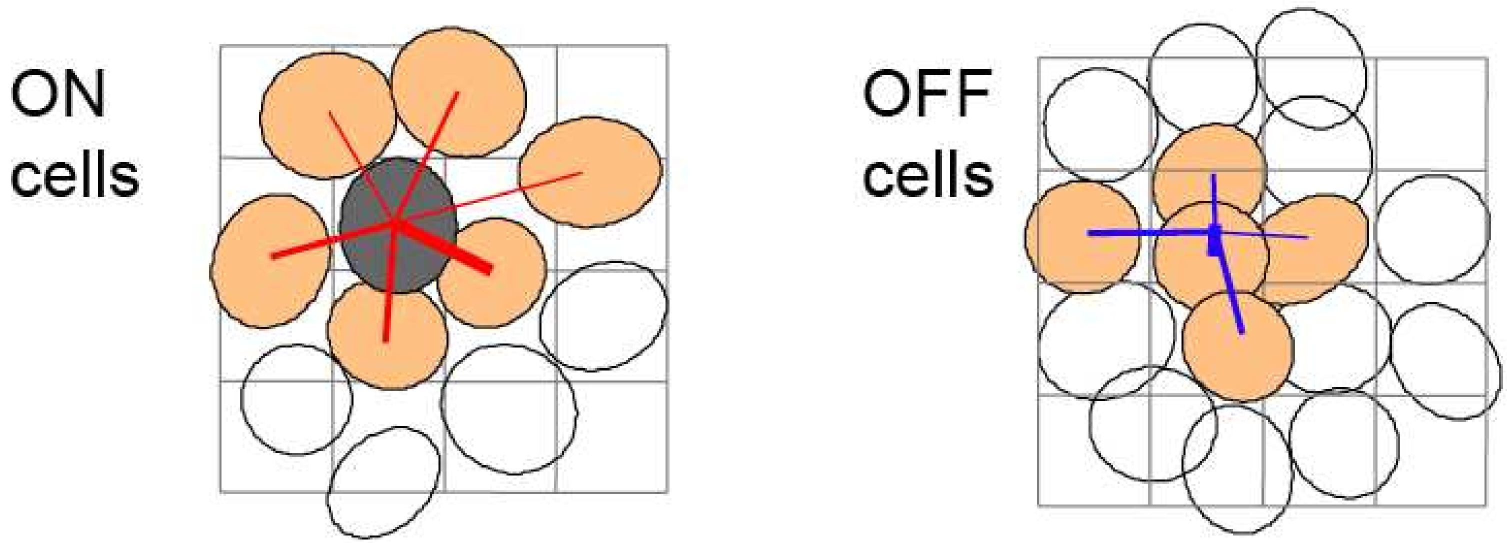
stimulus filter



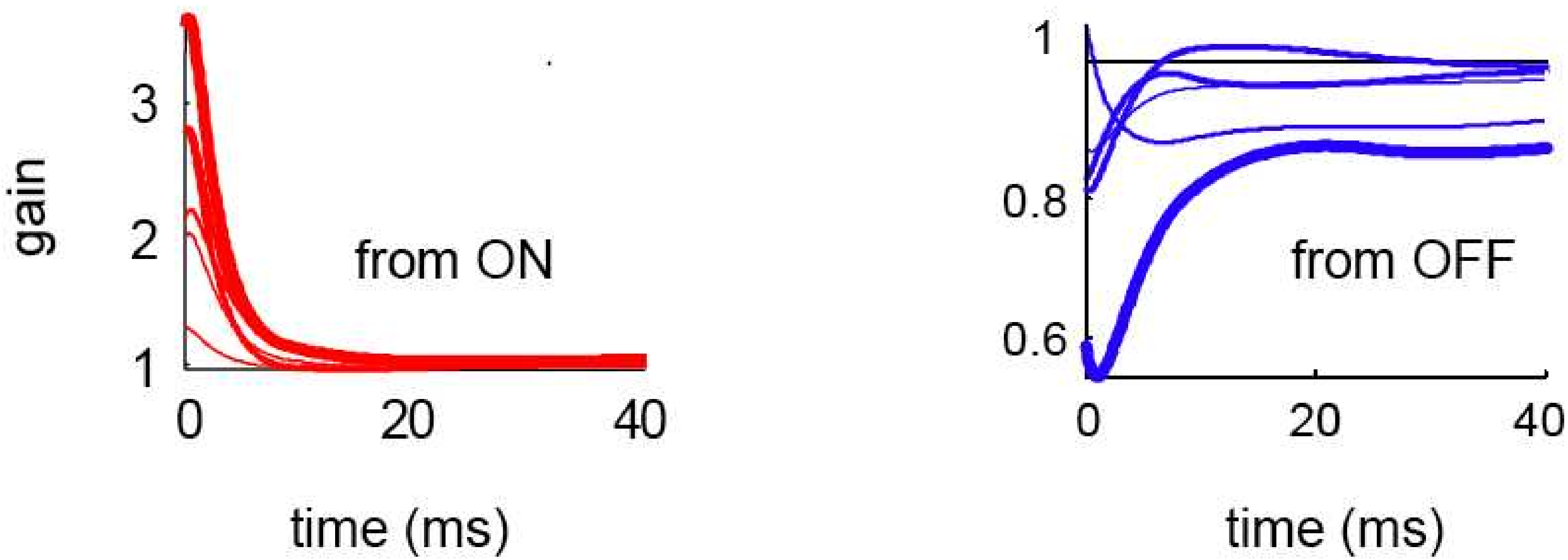
post-spike filter



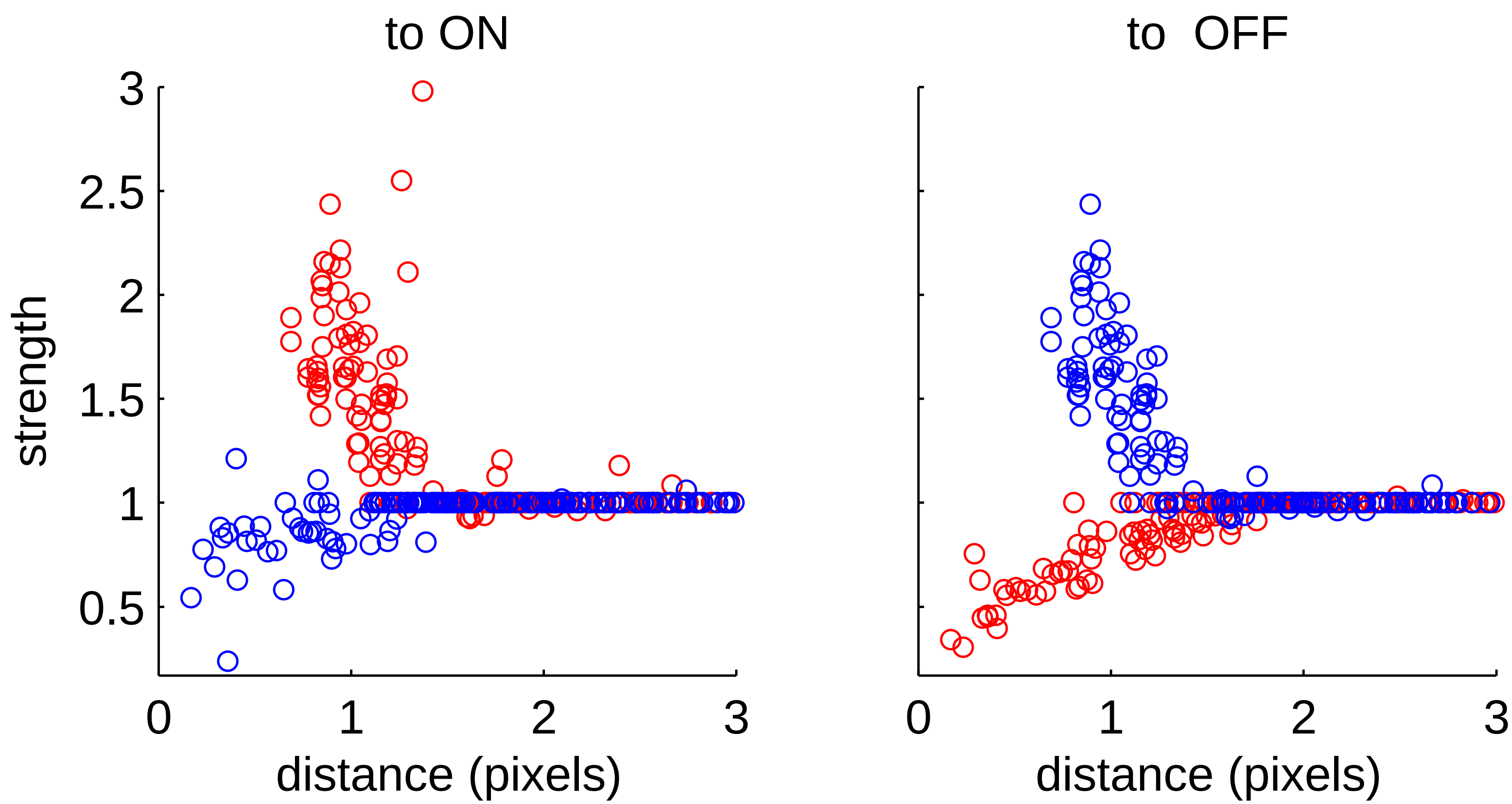
— θ_{stim} is well-approximated by a low-rank matrix (center-surround)



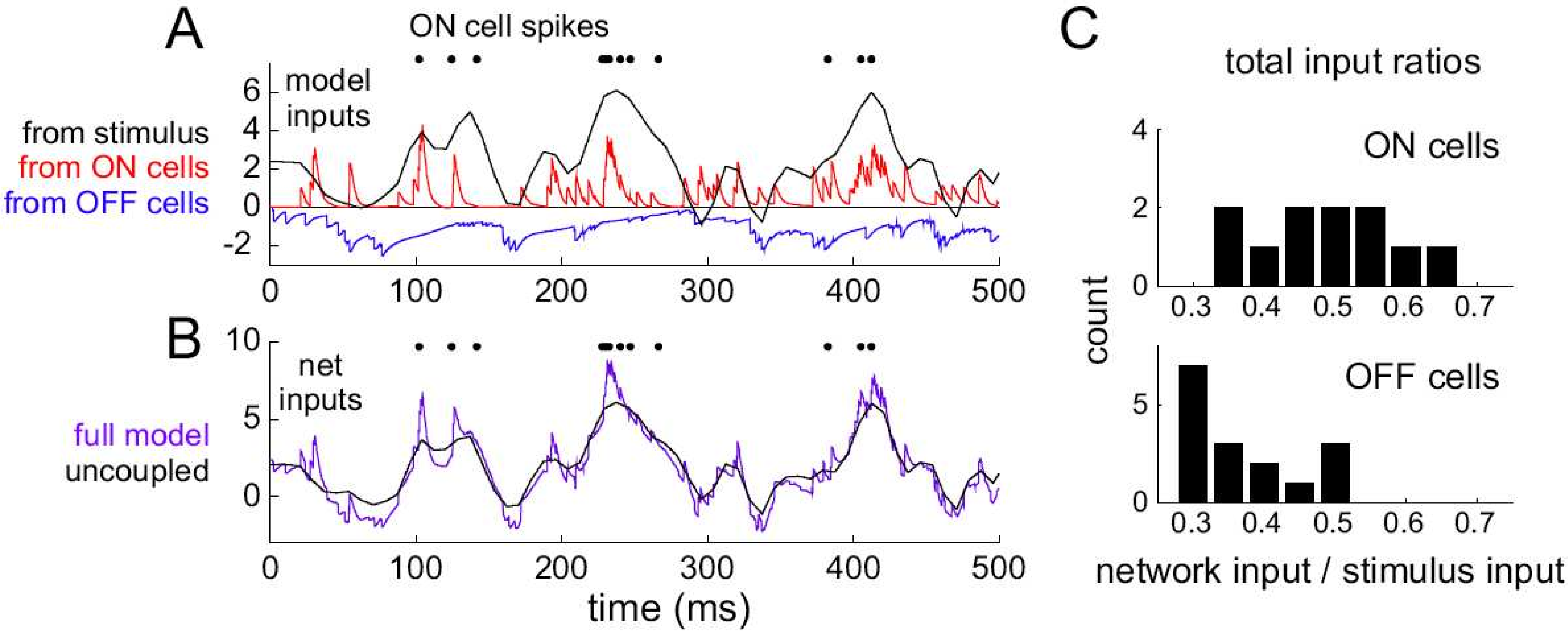
coupling filters



Nearest-neighbor connectivity



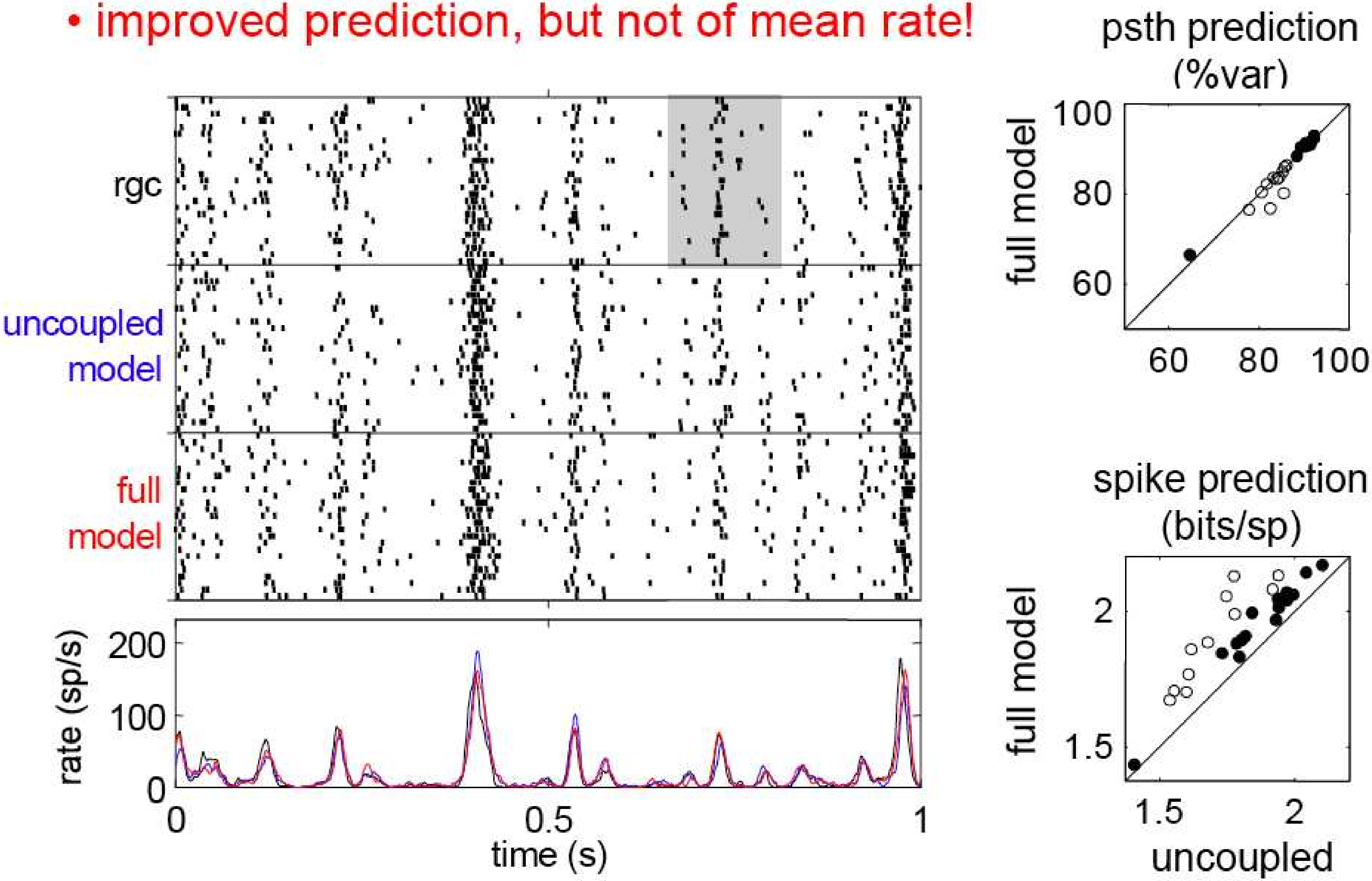
Network vs. stimulus drive



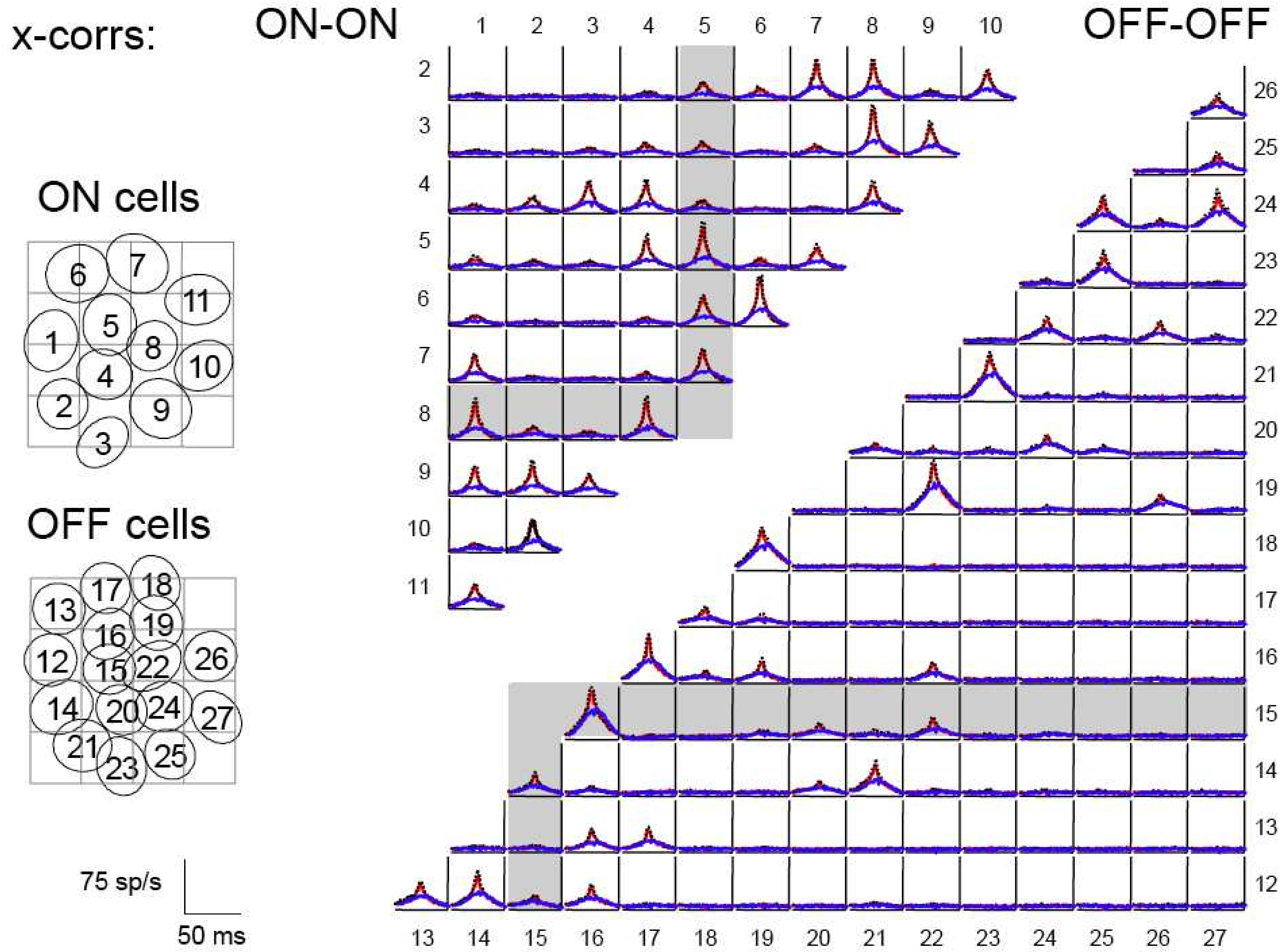
— Network effects are $\approx 50\%$ as strong as stimulus effects

Spike Train Prediction

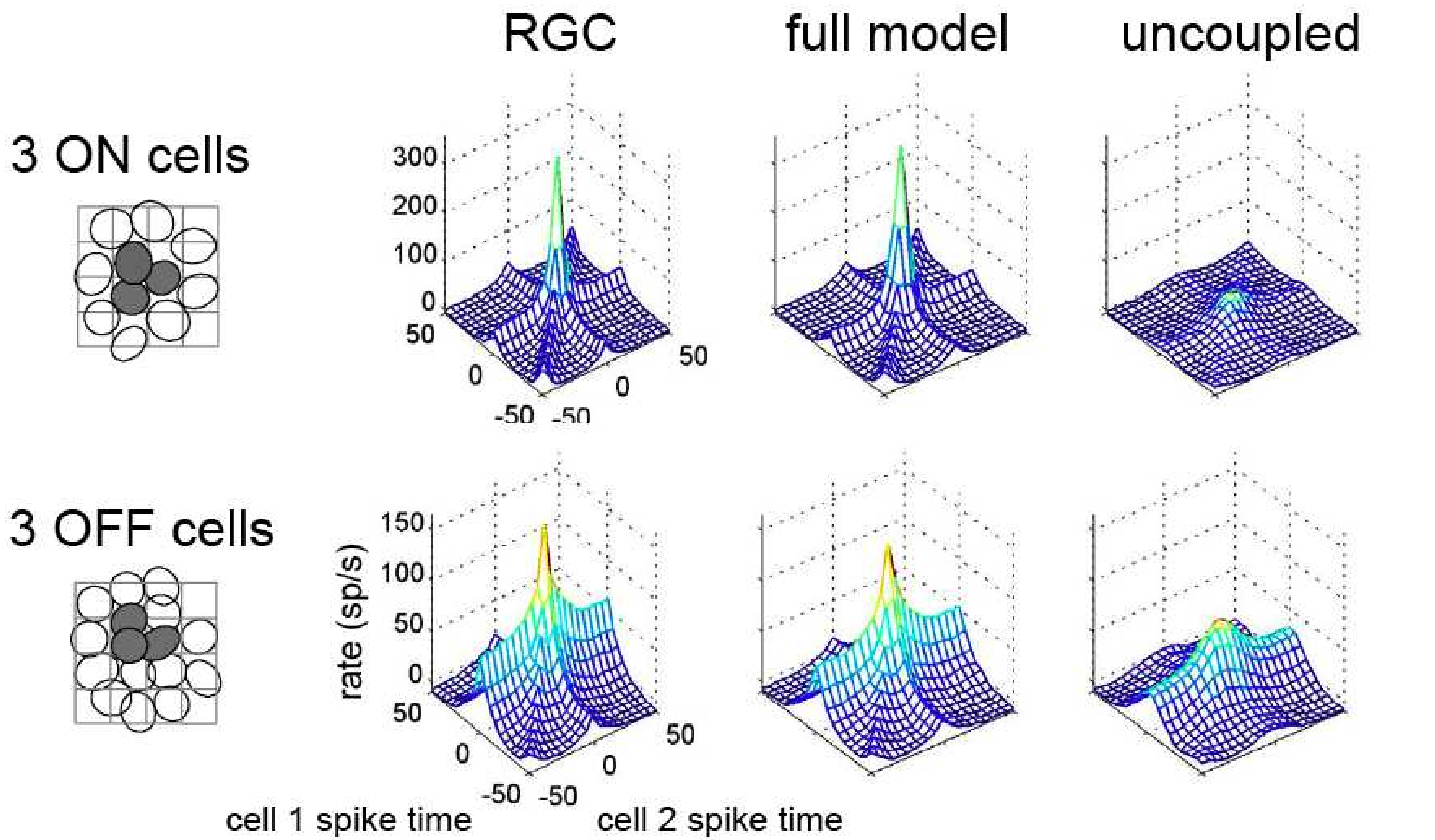
- improved prediction, but not of mean rate!



Model captures spatiotemporal cross-corrs

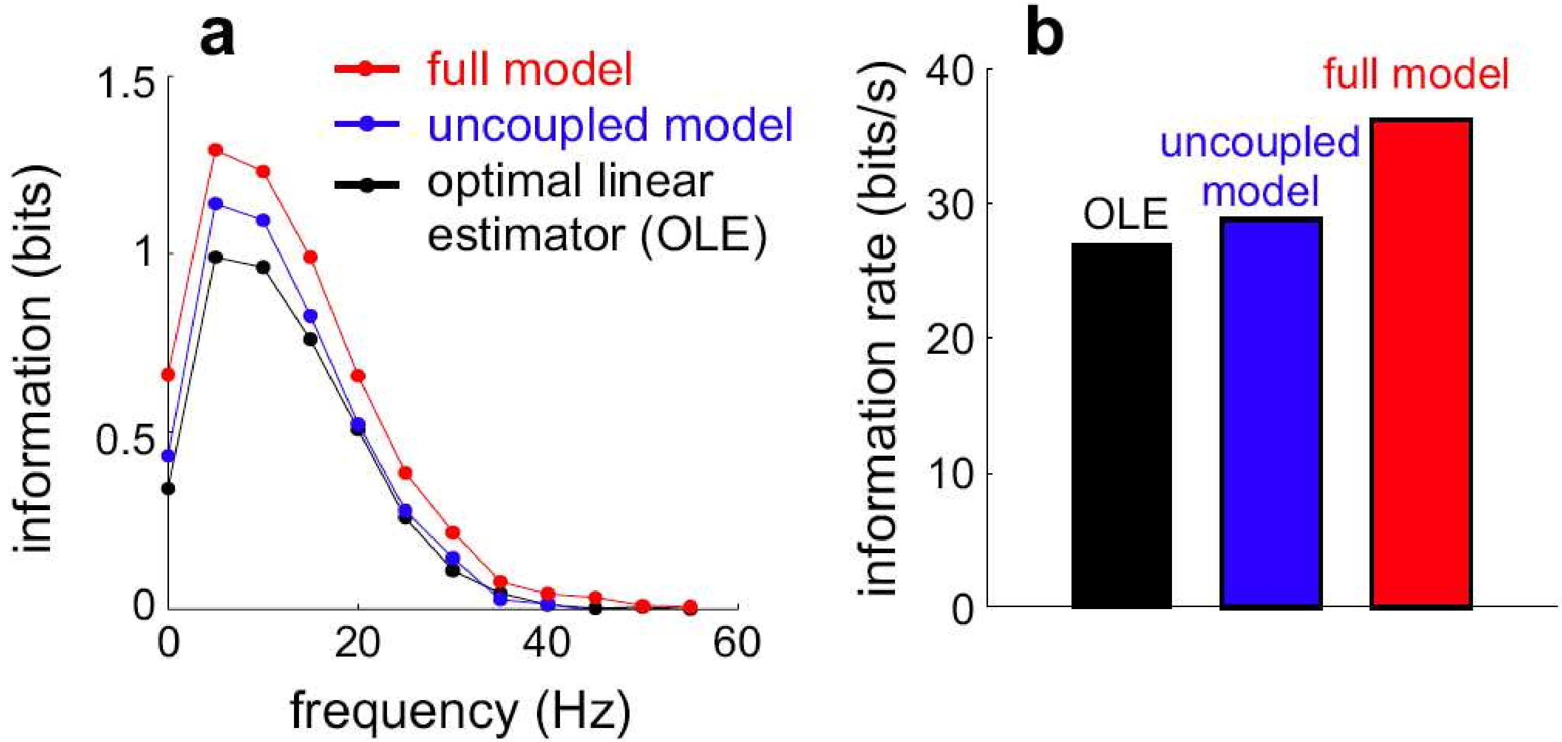


Triplet correlations



Optimal Bayesian decoding

$$E(\vec{x}|spikes) \approx \arg\max_{\vec{x}} \log P(\vec{x}|spikes) = \arg\max_{\vec{x}} [\log P(spikes|\vec{x}) + \log P(\vec{x})]$$

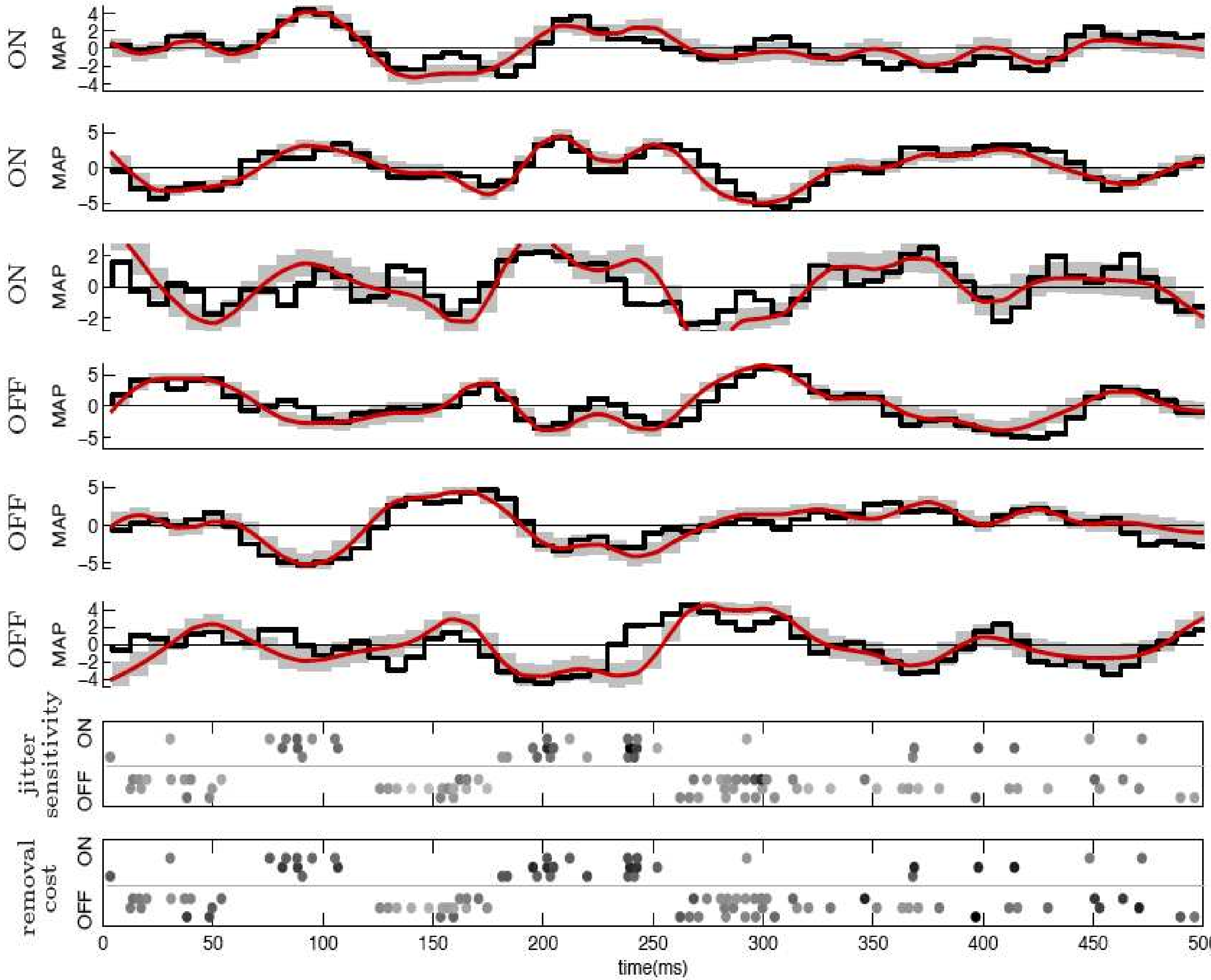


— Computational points:

- $\log P(spikes|\vec{x})$ is concave in $\vec{x}$: concave optimization again.
- Decoding can be done in linear time via standard Newton-Raphson methods, since Hessian of $\log P(\vec{x}|spikes)$ w.r.t. $\vec{x}$ is banded (Pillow et al., 2009).

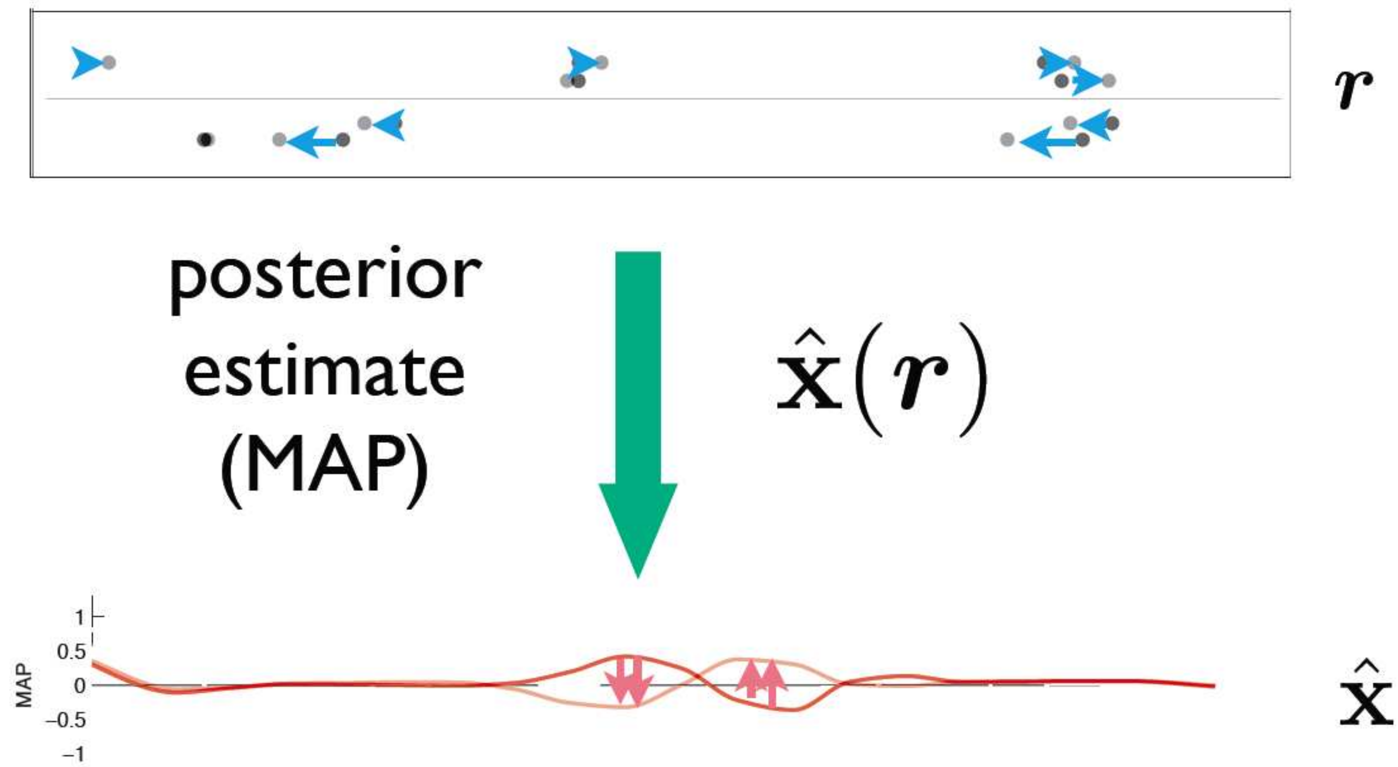
— Biological point: paying attention to correlations improves decoding accuracy.

Application: how important is timing?



— Fast decoding methods let us look more closely (Ahmadian et al., 2009)

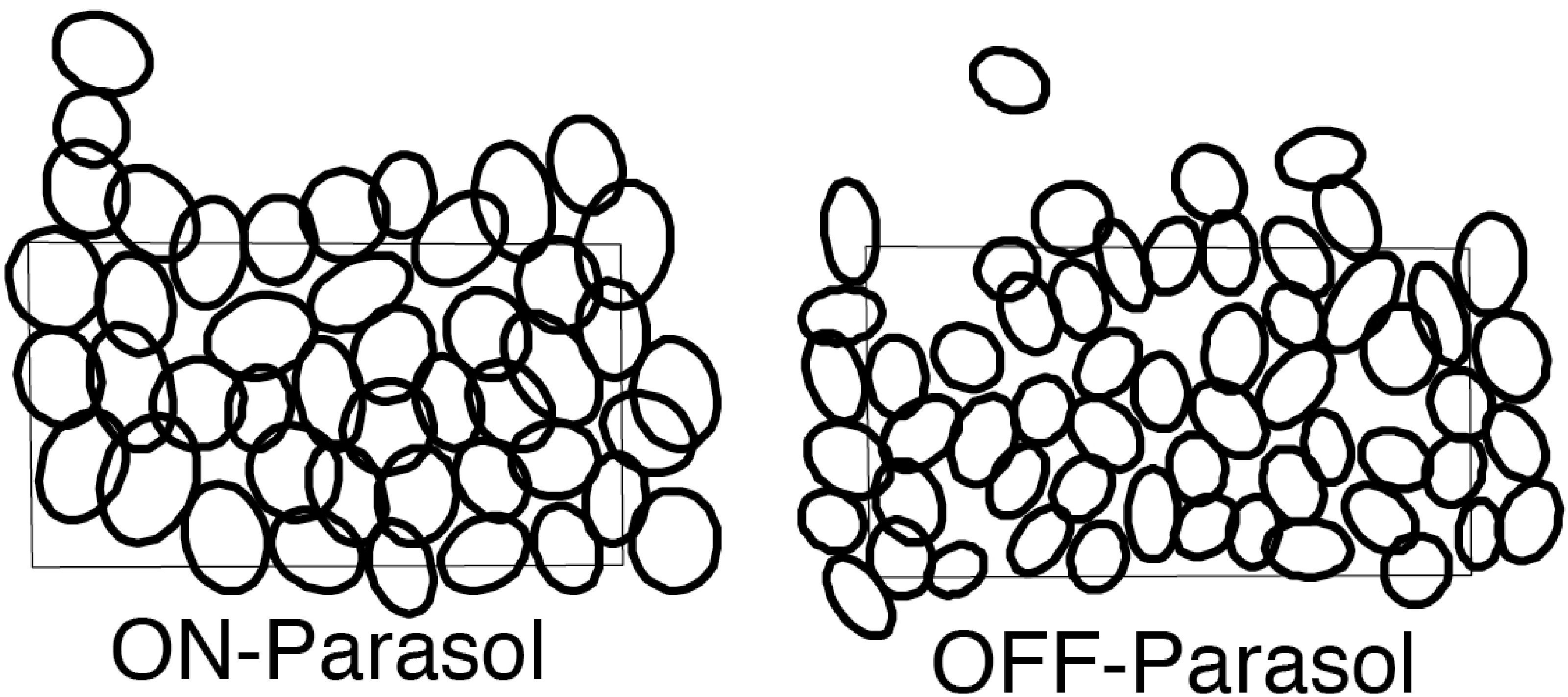
Constructing a metric between spike trains



$$d(r_1, r_2) \equiv d_x\left(\hat{x}(r_1), \hat{x}(r_2)\right)$$

Locally, $d(r, r + \delta r) = \delta r^T G_r \delta r$: interesting information in G_r.

Application: recurrent network modeling



— Do observed local connectivity rules lead to interesting network dynamics? What are the implications for retinal information processing? Can we capture these effects with a reduced dynamical model?

— Mean-field analysis (Toyoizumi et al., 2009)

Last example: optimal control of spike timing

How can we make a neuron exactly fire when we want it to fire?

Assume bounded inputs; otherwise problem is trivial.

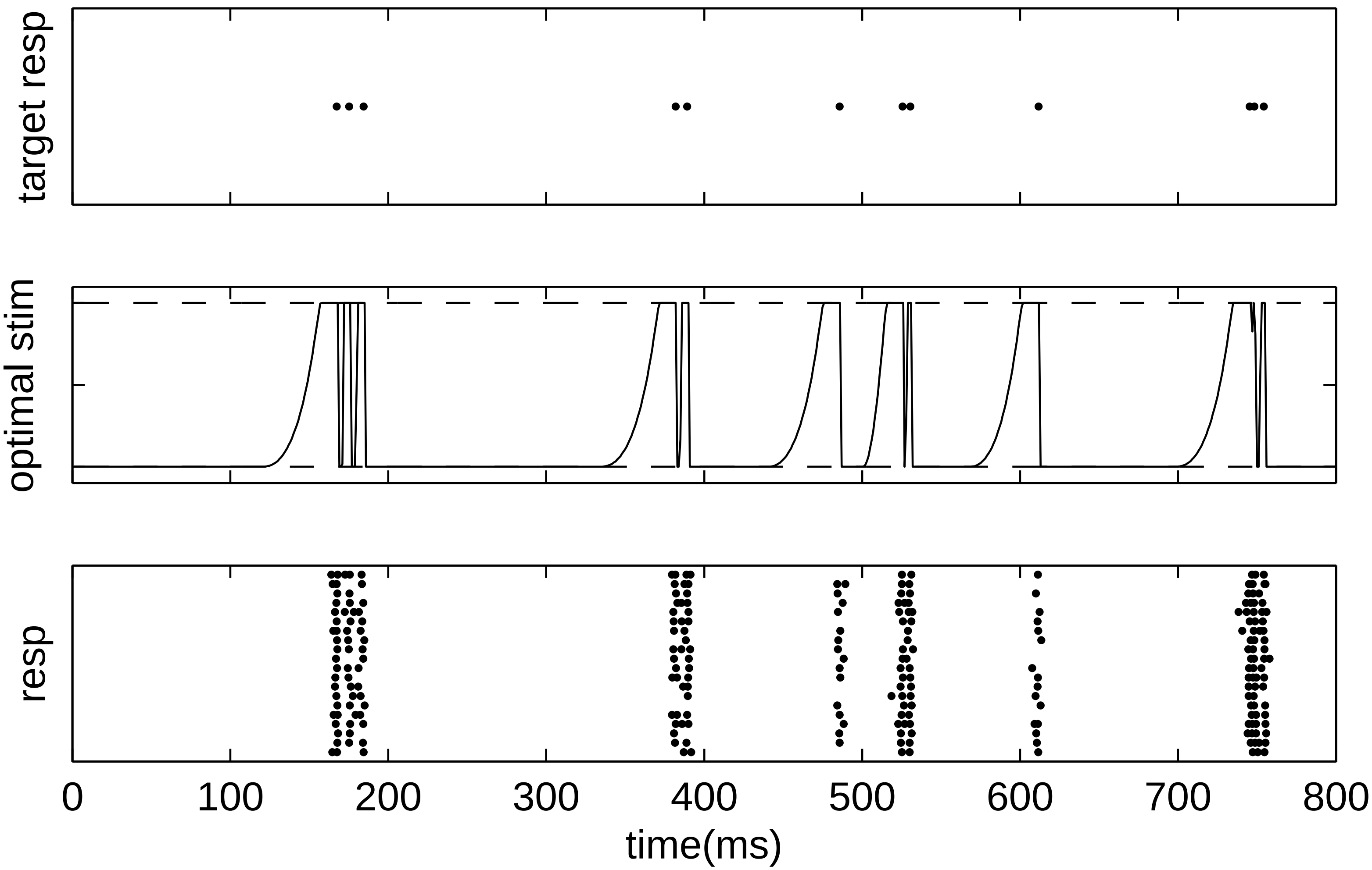
Start with a simple model:

$$
\begin{aligned}
\lambda_t &= f(V_t + h_t) \\
V_{t+dt} &= V_t + dt\,(-gV_t + aI_t) + \sqrt{dt}\sigma\epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0,1).
\end{aligned}
$$

Now we can just optimize the likelihood of the desired spike train, as a function of the input I_t, with I_t bounded.

Concave objective function over convex set of possible inputs I_t + Hessian is tridiagonal $\implies O(T)$ optimization.

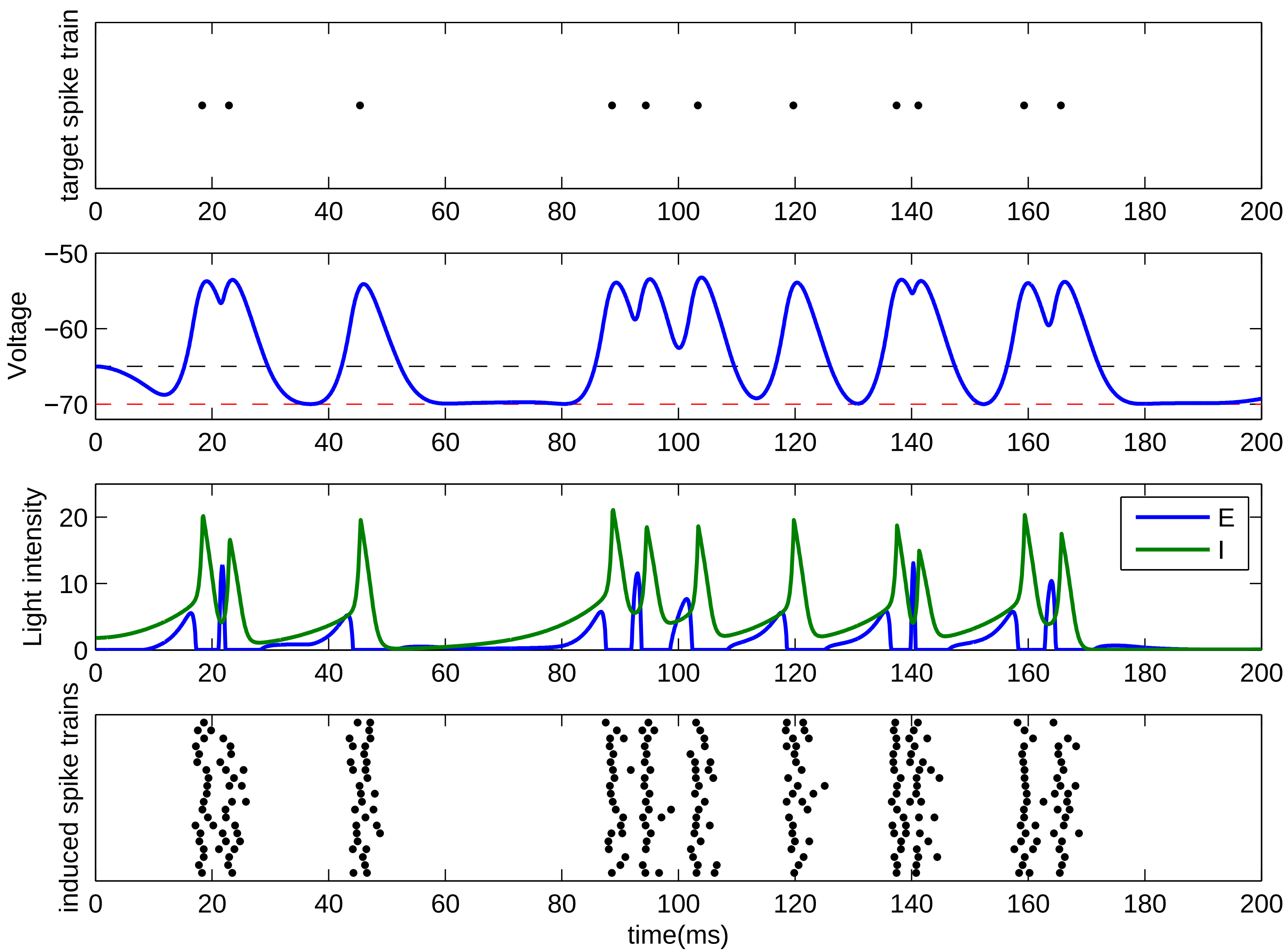
Optimal electrical control of spike timing



Optical conductance-based control of spiking

$$V_{t+dt} = V_t + dt\left(-gV_t + g_t^i(V^i - V_t) + g_t^e(V^e - V_t)\right) + \sqrt{dt}\sigma\epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0,1)$$

$$g_{t+dt}^i = g_t^i + dt\left(-\frac{g_t^i}{\tau_i} + a_{ii}L_t^i + a_{ie}L_t^e\right);\ g_{t+dt}^e = g_t^e + dt\left(-\frac{g_t^e}{\tau_i} + a_{ee}L_t^e + a_{ei}L_t^i\right)$$



References

Ahmadian, Y., Pillow, J., and Paninski, L. (2009). Efficient Markov Chain Monte Carlo methods for decoding population spike trains. *Under review, Neural Computation*.

Brown, E., Frank, L., Tang, D., Quirk, M., and Wilson, M. (1998). A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18:7411–7425.

Djurisic, M., Antic, S., Chen, W. R., and Zecevic, D. (2004). Voltage imaging from dendrites of mitral cells: EPSP attenuation and spike trigger zones. *J. Neurosci.*, 24(30):6703–6714.

Huys, Q., Ahrens, M., and Paninski, L. (2006). Efficient estimation of detailed single-neuron models. *Journal of Neurophysiology*, 96:872–890.

Huys, Q. and Paninski, L. (2009). Model-based smoothing of, and parameter estimation from, noisy biophysical recordings. *PLOS Computational Biology*, 5:e1000379.

Knopfel, T., Diez-Garcia, J., and Akemann, W. (2006). Optical probing of neuronal circuit dynamics: genetically encoded versus classical fluorescent sensors. *Trends in Neurosciences*, 29:160–166.

Paninski, L. (2004). Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15:243–262.

Paninski, L. (2006). The most likely voltage path and large deviations approximations for integrate-and-fire neurons. *Journal of Computational Neuroscience*, 21:71–87.

Paninski, L. (2009a). Fast Kalman filtering on dendritic trees. *In progress*.

Paninski, L. (2009b). Inferring synaptic inputs given a noisy voltage trace via sequential Monte Carlo methods. *Journal of Computational Neuroscience*, Under review.

Paninski, L., Ahmadian, Y., Ferreira, D., Koyama, S., Rahnama, K., Vidne, M., Vogelstein, J., and Wu, W. (2009). A new look at state-space models for neural data. *Submitted*.

Paninski, L. and Ferreira, D. (2008). State-space methods for inferring synaptic inputs and weights. *COSYNE*.

Paninski, L., Pillow, J., and Lewi, J. (2007). Statistical models for neural encoding, decoding, and optimal stimulus design. In Cisek, P., Drew, T., and Kalaska, J., editors, *Computational Neuroscience: Progress in Brain Research*. Elsevier.

Pillow, J., Ahmadian, Y., and Paninski, L. (2009). Model-based decoding, information estimation, and change-point detection in multi-neuron spike trains. *Under review, Neural Computation.*

Pillow, J., Shlens, J., Paninski, L., Sher, A., Litke, A., Chichilnisky, E., and Simoncelli, E. (2008). Spatiotemporal correlations and visual signaling in a complete neuronal population. *Nature*, 454:995–999.

Toyoizumi, T., Rahnama Rad, K., and Paninski, L. (2009). Mean-field approximations for coupled populations of generalized linear model spiking neurons with Markov refractoriness. *Neural Computation*, In press.

Truccolo, W., Eden, U., Fellows, M., Donoghue, J., and Brown, E. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble and extrinsic covariate effects. *Journal of Neurophysiology*, 93:1074–1089.

Vogelstein, J., Babadi, B., Watson, B., Yuste, R., and Paninski, L. (2008a). Fast nonnegative deconvolution via tridiagonal interior-point methods, applied to calcium fluorescence data. *Statistical analysis of neural data (SAND) conference.*

Vogelstein, J., Watson, B., Packer, A., Jedynak, B., Yuste, R., and Paninski, L. (2008b). Model-based optimal inference of spike times and calcium dynamics given noisy and intermittent calcium-fluorescence imaging. *Biophysical Journal*, In press; http://www.stat.columbia.edu/∼liam/research/abstracts/vogelstein-bj08-abs.html.

Wu, W., Gao, Y., Bienenstock, E., Donoghue, J. P., and Black, M. J. (2006). Bayesian population coding of motor cortical activity using a Kalman filter. *Neural Computation*, 18:80–118.

Wu, W., Kulkarni, J., Hatsopoulos, N., and Paninski, L. (2009). Neural decoding of goal-directed movements using a linear statespace model with hidden states. *IEEE Trans. Biomed. Eng.*, In press.