J Comput Neurosci (2012) 32:347–366
DOI 10.1007/s10827-011-0357-5

Optimal experimental design for sampling voltage on dendritic trees in the low-SNR regime

Jonathan Hunter Huggins · Liam Paninski

Received: 20 August 2010 / Revised: 8 July 2011 / Accepted: 28 July 2011 / Published online: 23 August 2011

Abstract Due to the limitations of current voltage sensing techniques, optimal filtering of noisy, undersampled voltage signals on dendritic trees is a key problem in computational cellular neuroscience. These limitations lead to voltage data that is incomplete (in the sense of only capturing a small portion of the full spatiotemporal signal) and often highly noisy. In this paper we use a Kalman filtering framework to develop optimal experimental design methods for voltage sampling. Our approach is to use a simple greedy algorithm with lazy evaluation to minimize the expected square error of the estimated spatiotemporal voltage signal. We take advantage of some particular features of the dendritic filtering problem to efficiently calculate the Kalman estimator's covariance. We test our framework with simulations of real dendritic branching structures and compare the quality of both time-invariant and time-varying sampling schemes. While the benefit of using the experimental design methods was modest in the time-invariant case, improvements of 25–100% over more naïve methods were found when the observation locations were allowed to change with time. We also present a heuristic approximation to the greedy algorithm that is an order of magnitude faster while still providing comparable results.

Action Editor: Wulfram Gerstner

J. H. Huggins (✉) · L. Paninski
Department of Statistics and Center for Theoretical Neuroscience, Columbia University, 1255 Amsterdam Ave., New York, NY 10027, USA
e-mail: jhh2143@columbia.edu

L. Paninski
e-mail: liam@stat.columbia.edu
URL: http://www.stat.columbia.edu/~liam

Keywords Optimal experimental design · State-space model · Voltage-sensitive imaging · Markov random field · Submodularity

1 Introduction

Understanding dendritic computation remains one of the principal open problems in cellular and computational neuroscience (Stuart et al. 1999; Spruston 2008; Sjostrom et al. 2008). The key challenge is the difficulty of recording physiological signals (particularly voltage) with sufficient spatiotemporal resolution on the dendritic tree. If we have the full spatiotemporal voltage signal then it is possible to use straightforward statistical methods to infer many biophysical quantities of interest (Morse et al. 2001; Wood et al. 2004; Huys et al. 2006), such as passive cable parameters, active properties, and in some cases even time-varying information, such as the rate of synaptic input. Unfortunately, technical challenges limit multiple-electrode recordings from dendrites to only a few electrodes, typically targeted far from the tips of dendritic branches (Stuart and Sakmann 1994; Cox and Griffith 2001; Cox and Raol 2004; Bell and Craciun 2005; Petrusca et al. 2007; Nevian et al. 2007; Homma et al. 2009). High-resolution two-photon imaging techniques provide more spatially-complete observations, but with significantly lower signal-to-noise (Djurisic et al. 2008; Homma et al. 2009; Canepari et al. 2010, 2011). In particular, high-resolution random-access voltage imaging techniques—for example, those based on acousto-optic deflection (AOD) (Vucinic and Sejnowski 2007; Reddy et al. 2008; Grewe and Helmchen 2009; Grewe et al. 2010)—have the potential to achieve kilohertz

recording rates in three dimensions. In this paper we focus on techniques applicable to this random-access, low-SNR case.

The technical limitations of current voltage measurement technologies lead to two sources of difficulty: 1) voltage data is incomplete (in the sense of only capturing a small portion of the full spatiotemporal signal) and 2) such data is available only in limited quantities for a single neuron (for example, in many cases only relatively short recording durations are possible due to photodamage). Statistical techniques offer a path to partially offset these difficulties by providing methods to de-noise the data and infer the voltage in unobserved compartments while also maximizing the amount of information that can be extracted from the data that can be gathered with currently available methods. As discussed in previous work (Huys and Paninski 2009; Paninski 2010), state-space filtering methods such as the Kalman filter are an attractive choice for addressing these concerns because they allow us to explicitly incorporate the known biophysics of dendritic dynamics, along with the known noise properties of the measurement device.

In addition to providing a method for estimating the spatio-temporal dendritic voltage given limited observations, the Kalman filter offers a framework for addressing the second difficulty: namely, that only limited spatial voltage data can be collected from a single neuron. We can make each measurement as informative as possible by developing an optimal experimental design (Federov 1972; Chaloner and Verdinelli 1995; Krause et al. 2008b; Lewi et al. 2009; Seeger 2009). Optimal experimental design requires solving two problems in a computationally efficient manner. First, we must evaluate the quality of a given proposed design. Second, we must efficiently search over a large space of candidate designs. This paper proposes solutions to both of these problems. To address the first problem we present an efficient implementation of the Kalman filter-smoother. For dendritic trees which have on the order of $N \sim 10^4$ compartments, the standard implementations of the Kalman filter-smoother are impractical because they require $O(N^3)$ time and $O(N^2)$ space. We therefore extend the results of Paninski (2010) to approximately calculate the filter-smoother estimator covariance matrix as a low rank perturbation to the steady-state (zero-SNR) solution in $O(N)$ time and space. Using these computed covariance matrices we can easily calculate the expected mean-squared error, the mutual information, and other design metrics. To efficiently search the space of possible designs, we utilized "lazy greedy" methods from the literature on submodular optimization (Nemhauser et al. 1978; Krause et al. 2007, 2008b; Das and Kempe 2008). These lazy evaluation methods proved critical to making the optimization tractable. We also present an even faster heuristic approximation to the full greedy algorithm that gives comparable performance in the case of both time-stationary and time-varying sampling schemes.

We begin in Section 2 below by discussing the formulation of the Kalman filter for our task and outlining the derivation of the fast backward smoother. Next we discuss our approach to optimal experimental design in Section 3. In Section 4 we present our results, including the effectiveness of our greedily selected sampling schemes in both time variant and invariant cases on a number of neuronal geometries. Conclusions and possible directions for future research are considered in Section 5.

2 Fast computation of objective functions for experimental design

When taking voltage measurements on a dendritic tree using, for example, laser-based scanning techniques (Vucinic and Sejnowski 2007; Grewe and Helmchen 2009), there is flexibility in designing a sampling scheme. If an experimentalist knows beforehand that the data will be processed using a Kalman filter, then we can exploit this fact to select observation locations so that the filter will be able to recover as much information as possible from those observations. A reasonable choice for an objective function is the weighted mean squared error (MSE) of the estimated spatiotemporal voltage, which is equivalent to the weighted sum of the variances of the smoothed mean $\mu_t^s = E(V_t|Y_{1:T})$:

$$\upsilon_w(\mathcal{O}) = \sum_{t=0}^{T}\sum_{i=1}^{N} w(i,t)[C_t^s]_{ii}, \tag{1}$$

where T is the number of time steps in the observation sequence, N is the number of compartments in the dendritic tree, $w(i,t)$ is the weight on compartment i at time t (i.e., $w(i,t)$ can be chosen to reflect the importance of compartment i to the experimenter), $C_t^s = Cov(V_t|Y_{1:T})$ is the optimal estimator covariance matrix at time t, and $\mathcal{O}$ is the set of selected sample locations. Thus, the diagonal of C_t^s represents the uncertainty in our predicted voltages at time t given all the observed data; this quantity will play a central role in our analysis, as discussed in more detail below.

The MSE is not the only possible metric for experimental design based on C_t^s. Instead of minimizing the summed MSE, we could minimize the maximum weighted MSE summed over all time points:

$$\sum_{t=0}^{T} \max_{i\in\{1\ldots N\}} w(i,t)[C_t^s]_{ii},$$

or just the maximum weighted MSE:

$$\max_{i,t} w(i,t)[C_t^s]_{it};$$

see, e.g., Krause et al. (2007) for additional discussion of this minimax approach. A third possibility is to maximize the mutual information $I(V_{1:T}; Y_{1:T})$ between the true underlying voltages V and the observed data Y (Cover and Thomas 1991). In the Kalman setting, it turns out that this mutual information may be computed efficiently via a forward-backward filter-smoother approach (see Appendix for details).

Thus, whatever our choice of objective function, we need to compute quantities related to the filter-smoother estimator covariance C_t^s. Our first task, therefore, is to develop efficient methods for calculating this quantity.

2.1 The fast low-SNR Kalman filter-smoother

Our analysis is based on a simple linear-Gaussian (Kalman) model for the noisy dynamics of the dendritic voltage and the observed data. This Kalman model serves as crude approximation, of course, but it is a reasonable starting point, particularly for subthreshold passive dynamics; see Huys and Paninski (2009) for more detailed background and discussion. In the state space voltage filtering formulation of the Kalman filter, the hidden state vectors V_t are the true voltages. If we discretize the tree into N compartments then V_t has dimensionality N. For the dynamics and observation equations we take

$$V_{t+dt} = AV_t + \epsilon_t,\ \epsilon_t \sim \mathcal{N}(0, \sigma^2 dt I) \tag{2}$$

$$y_t = B_t V_t + \eta_t,\ \eta_t \sim \mathcal{N}(0, W_t). \tag{3}$$

Here A is an $N \times N$ matrix that implements the backward-Euler implementation of the cable equation (see below), ϵ_t denotes a Gaussian noise source that perturbs the cable equation stochastically on each time step, and $\mathcal{N}(\mu, C)$ denotes a Gaussian density of mean μ and covariance C. The vectors y_t are the measurements on the dendritic tree, B_t is a matrix that specifies how the observations are related instantaneously to the voltage vector, and the covariance matrix W_t defines the noisiness of those observations. A single non-zero entry (set to unity) in column i of a row of B_t indicates that an observation was made in the i-th compartment at time t. Additional non-zero terms in the column introduce a blurring effect into the measurement. Thus, each observation in the set $\mathcal{O}$ from Eq. (1) corresponds to a row of B_t, along with the corresponding elements of the covariance matrix W_t. Note that we will only consider temporally uncorrelated noise here, though generalizations are possible (Paninski 2010).

We define A using a backward Euler (implicit) discretization for the cable equation, as discussed in Paninski (2010): in the noiseless case,

$$V_{t+dt}(x) = V_t(x) + dt \left(-g_x V_{t+dt}(x) + \sum_{w\in N(x)} a_{xw} \left[V_{t+dt}(w) - V_{t+dt}(x) \right] \right). \tag{4}$$

where $V_t(x)$ denotes the voltage in compartment x at time t, g_x is the membrane conductance in compartment x, $N(x)$ is the set of compartments adjoining x, and a_{xw} is the intercompartmental conductance between compartments x and w. We use the implicit time discretization because it is well-known that the forward Euler method (not described here) for solving the cable equation is unstable for large values of adt (Hines 1984; Press et al. 1992). Writing the cable equation in matrix-vector form gives

$$V_{t+dt} = V_t + KV_{t+dt}$$

for an appropriate matrix K of "Hines" form (Hines 1984); rearranging slightly gives

$$V_{t+dt} = (I - K)^{-1} V_t.$$

It is straightforward to replace A in Eq. (2) with $(I - K)^{-1}$.

It is worth noting explicitly here that we will assume that all of the parameters mentioned above are known, or more generally, that they can be estimated

experimentally in an on-line manner (i.e., while the preparation is still stable). Thus we will assume that the anatomical structure of the neuron (encapsulated by the connectivity graph $N(x)$) can be measured at the beginning of the experiment, and that reasonable estimates of the biophysical parameters g_x, a_{xw}, W_t, etc., may be obtained either *a priori* or via a version of the expectation-maximization approach discussed in Huys and Paninski (2009). In practice, we found that the exact values of these parameters, taken over a biophysically reasonable range, did not meaningfully impact the output of the optimal designs computed below. We acknowledge that on-line reconstruction of dendritic trees remains a challenging task; however, significant research into automated dendrite reconstruction methods is underway. For example, the powerful TREES MATLAB toolbox was recently introduced for exactly this type of task (Cuntz et al. 2010). Similarly, Losavio et al. (2008) demonstrate excellent qualitative and quantitative performance using their ORION software for fully automatic morphological reconstruction. In addition, competitions such as the Digital Reconstruction of Axonal and Dendritic Morphology (DIADEM) Challenge[1] continue to encourage research in this area.

Now that the model has been specified, we may derive an efficient smoothed backward Kalman recursion using methods similar to those employed by Paninski (2010) for the forward Kalman recursion. We assume that the forward mean

$$\mu_t^f = E(V_t|Y_{1:t})$$

and covariance

$$C_t^f = Cov(V_t|Y_{1:t})$$

(where $Y_{1:t}$ denotes all of the observed data $\{y_s\}$ up to time t) have already been computed by such methods. (Note that the forward recursion alone is insufficient to compute the MSE, which is a function of C_t^s and therefore of the observation times and locations both before and after the current time t.) In particular, we assume C_t^f has been approximated as a low rank perturbation to the steady state covariance matrix C_0, in the form

$$C_t^f \approx C_0 + U_t D_t U_t^T, \tag{5}$$

where D_t is an $n \times n$ matrix with $n \ll N$ and $U_t D_t U_t^T$ is a low-rank matrix.

[1] http://www.diademchallenge.org/

The equilibrium covariance C_0 can be computed from the forward Kalman recursion for the covariance matrix (Durbin and Koopman 2001),

$$C_t^f = \left[(AC_{t-dt}^f A^T + \sigma^2 dt I)^{-1} + B_t^T W_t^{-1} B_t \right]^{-1}, \tag{6}$$

by taking the zero-SNR steady state limit (i.e., where $B_t = 0$ and $C_t = C_{t-dt}$), which satisfies the discrete Lyapunov equation

$$AC_0 A^T + \sigma^2 dt I = C_0. \tag{7}$$

In this case the equation has the explicit geometric series solution (Brockwell and Davis 1991)

$$C_0 = \sigma^2 dt \sum_{i=0}^{\infty} A^{2i} = \sigma^2 dt \left(I - A^2\right)^{-1}, \tag{8}$$

since A is symmetric and stable (i.e., all eigenvalues are less than 1). See Paninski (2010) for details on calculating C_t^f and efficiently multiplying by C_0 and C_0^{-1} without having to represent these large matrices explicitly. In short, we can take advantage of the fact that the matrix K that implements the backward Euler propagation is symmetric and tree-tridiagonal (also known as "Hines" form (Hines 1984)): all off-diagonal elements K_{xw} are zero unless the compartments x and w are nearest neighbors on the dendritic tree. We can then use efficient sparse matrix divide methods to multiply by $(I - K)^{-1}$.

Before proceeding, let us describe an intuitive justification for the low rank approximation (Eq. (5)). If we make k observations at $t = 1$ then, by an application of the Woodbury matrix lemma to Eq. (6), Eq. (5) holds exactly with U_1 having rank k. If we make no further observations ($B_t = 0$ for all $t > 1$), then C_t^f follows the update rule

$$\begin{aligned} C_t^f &= AC_{t-1}^f A^T + \sigma^2 dt I \\ &= A[C_0 + U_{t-1} D_{t-1} U_{t-1}^T] A^T + \sigma^2 dt I \\ &= C_0 + AU_{t-1} D_{t-1} U_{t-1} U_{t-1} A^T, \end{aligned}$$

where the third equality follows from Eq. (7). Iterating the equation gives

$$C_t^f = C_0 + A^{t-s} U_s D_s U_s^T (A^{t-s})^T,$$

where s denotes the time of the last observation. Since A is stable, the second term will decay exponentially; thus, for $t - s$ sufficiently large, we can discard some dimensions of the perturbation $AU_{t-1} D_{t-1} U_{t-1} U_{t-1} A^T$ without experiencing much error in C_t^f. In the case that additional observations become available with each time step t, a similar phenomenon governs the behavior of C_t^f: long-ago observations are eventually forgotten,

due to the exponential decay caused by the double multiplication $AC_t^f A_T$. See Paninski (2010), Paninski et al. (2011) for further empirical justification that the error introduced by the low-rank approximation is, in fact, small.

Now, to calculate the low-rank approximation for the smoother, we repeatedly express portions of the classical recursion for the smoothed covariance (Shumway and Stoffer 2006)

$$C_t^s = C_t^f + C_t^f A^T \left[C\left(V_{t+1}|Y_{1:t}\right)\right]^{-1} \times \left[C_{t+1}^s - C\left(V_{t+1}|Y_{1:t}\right)\right]\left[C\left(V_{t+1}|Y_{1:t}\right)\right]^{-1} AC_t^f$$

in low rank form or, in the final step, as a low rank correction to the steady state covariance matrix,

$$C_t^s \approx C_0 + P_t G_t P_t^T,$$

which is justified by the same logic as the forward case. This approximation allows us to calculate C_t^s in $O(N)$ time and space instead of the $O(N^3)$ time and $O(N^2)$ space required by standard implementations. However, some $O(n^3)$ and $O(nN)$ operations remain, where n is the rank of the correction term P_t above; as motivated above, this is roughly proportional to the number of observations per time step, scaled by the observation SNR. (Indeed, an operational definition of the "low-SNR" regime here is that $n \ll N$, in which case the approximate filter discussed here is qualitatively more efficient than the standard implementation.) See Appendix A for the full derivation, and Paninski et al. (2011) for further discussion.

It is important to recall that in the linear-Gaussian case considered here, the posterior covariances C_t^s do not depend on the observed data y_t. Thus, once we know the system parameters (A, W_t, and B_t) we can optimize the experimental design without having to update our designs "on the fly" as we observe new data y_t; as discussed recently in, e.g., Lewi et al. (2009), this online experimental design optimization is considerably more computationally challenging than the problem we will focus on here.

In the case that B_t is time-invariant (stationary), there is a further speed-up that can be applied: the forward recursion C_t^f will converge to a limit, after which we no longer need to compute U_t and D_t. Specifically, we stop recomputing U_t and D_t after some time t_{conv}, when the instantaneous rate of change of D_t drops below some threshold τ:

$$\|diag(D_{t-dt}^f) - diag(D_t)\| < \tau,$$

where $\|\mathbf{v}\|$ is the 2-norm of the vector $\mathbf{v}$. A similar procedure can be used to check for the convergence of C_t^s, significantly speeding up the backward computation. This idea may also be applied in the case of periodically-varying observations B_t, though we have not pursued this direction in depth.

3 Selecting observation locations via submodular optimization methods

In this paper we will focus on using the weighted variance (Eq. (1)) to optimize the sampling scheme. For the sake of clarity we limit our discussion to the unweighted case,

$$w(i,t) = 1 \quad \forall i,t,$$

for the moment. All the results in this section can be extended to the weighted case in a straightforward manner. Our objective function therefore has the simpler form

$$\upsilon(\mathcal{O}) = \sum_{t=0}^{T}\sum_{i=1}^{N}[C_t^s]_{ii}.$$

Selecting the optimal set of observations $\mathcal{O}$ is NP-hard in general (Das and Kempe 2008; Krause et al. 2008a), and therefore we must rely on approximate methods to design an optimal sampling scheme. There is a significant body of work on maximizing objective functions that are *submodular* (see, e.g., Krause et al. 2008b and references therein). Submodularity is a diminishing returns property: the more observations added, the smaller the increase achieved by each additional observation. Let $\mathcal{S}$ be the set of observations from which to choose. Formally, some real-valued function $F(\cdot)$ is submodular if for $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{S}$ and $e \in \mathcal{S}\backslash\mathcal{B}$, it holds that $F(\mathcal{A} \cup \{e\}) - F(\mathcal{A}) \geq F(\mathcal{B} \cup \{e\}) - F(\mathcal{B})$. In other words, including the element e in the argument set $\mathcal{O} \subseteq \mathcal{S}$ increases $F(\mathcal{O})$ less as $\mathcal{O}$ becomes larger. The advantage is that greedy methods, surprisingly, are guaranteed to lead to fairly good optimizers of submodular functions: specifically, greedy optimizers perform within a constant factor of the best possible optimizers (which will typically be exponentially more difficult to compute; Nemhauser et al. 1978), as long as the function is submodular and monotonic.

It can be shown (Krause and Guestrin 2005; Seeger 2009) that the mutual information $I(V;Y)$ in our model is submodular (considered as a function of subsets of

the observation vector Y), as long as the noise covariance matrices W_t are diagonal (i.e., the observations are conditionally independent given V). This makes intuitive sense: as more observations are added in this setting, additional observations will contribute smaller amounts of new information. Since the mutual information and MSE are closely related in linear-Gaussian models (the former involves the log-determinant of the posterior covariance matrix, while the latter involves the trace of the same matrix), it is reasonable to conjecture that the following "variance reduction" function is submodular:

$$\begin{aligned}\rho(\mathcal{O}) &= \upsilon(\emptyset) - \upsilon(\mathcal{O}) \\ &= \sum_{t=0}^{T}\sum_{i=1}^{N}([C_0]_{ii} - [C_t^s]_{ii}) \\ &= \sum_{t=0}^{T} tr(C_0 - C_t^s) \\ &\approx \sum_{t=0}^{T} tr(C_0 - (C_0 + P_t G_t P_t^T)) \\ &= -\sum_{t=0}^{T} tr(P_t G_t P_t^T), \end{aligned} \tag{9}$$

where $tr(\cdot)$ denotes the trace of its matrix argument and we have taken advantage of the low rank approximation of C_t^s to eliminate the need to calculate C_0.[2] Note in addition that this variance reduction function is monotonically increasing: the more observations we make, the smaller the posterior variance in this model.

As noted above, for any monotonically increasing submodular function we can apply a simple greedy algorithm to obtain the optimality guarantees of Nemhauser et al. (1978) (cf. Krause et al. 2008b). During each iteration, the algorithm selects the observation location that provides the greatest increase in the objective function. This greedy algorithm can be improved by "lazily" re-evaluating only those elements with the potential to maximize the greedy objective on any iteration (Krause et al. 2008b). More specifically, let $F(\cdot)$ be a submodular objective function. For iteration i with established observation set $\mathcal{O}_{i-1}$ and observation location $e_j \in \mathcal{S}$, put $\delta_{i,j} = F(\mathcal{O}_{i-1} \cup \{e_j\}) - F(\mathcal{O}_{i-1})$ and $\Delta_{i,j} = \max_{l<j}\{\delta_{i,l}\}$. In other words, $\delta_{i,j}$ is the increase in $F(\cdot)$ if e_j is chosen as the i-th observation and $\Delta_{i,j}$ is the maximum such value for e_l examined before e_j. Then on iteration $i+1$ calculate $\delta_{i+1,j}$ if and only if $\delta_{i,j} \geq \Delta_{i+1,j}$ and otherwise put $\delta_{i+1,j} = \delta_{i,j}$. We do not always have to calculate $\delta_{i+1,j}$ because, by the submodularity of $F(\cdot)$, $\delta_{i+1,j} \leq \delta_{i,j}$, so if $\delta_{i,j} \leq \Delta_{i+1,j}$ then $\delta_{i+1,j} \leq \Delta_{i+1,j}$ and we can conclude a priori that e_j will be discarded. In other words, during each loop we keep track of the largest improvement in the objective function we have found so far. We only evaluate the objective value obtained by adding some observation e_j if the improvement we calculated for that element previously is greater than the largest improvement. This is because we know by the submodularity of $F(\cdot)$ that the improvement we get by adding e_j to the observation set will never increase when a different element is included, so there is no possibility that it will be the best observation to select during this iteration. This procedure greatly reduces computation because for most e_j the change in the objective will be lower than the maximum improvement, so only a few e_j have to be considered on each iteration.

While in many cases the variance reduction $\rho(\cdot)$ defined above is submodular (Das and Kempe 2008; Krause et al. 2008a), we found empirically that in our case $\rho(\cdot)$ is not quite submodular (see Section 4.4 for details). Nevertheless, the greedy algorithm (and the lazy implementation) proved to be quite effective in practice, as we will see below.[3]

3.1 Considerations when optimizing with non-stationary observations

State-of-the-art laser-based dendritic voltage measurement methods allow for time-varying sampling schemes (c.f. Vucinic and Sejnowski 2007 and references in Section 1). Conceptually, in the case of a time-stationary observation scheme, we can view the observations as *almost* simultaneous, in the sense that the microscope scans the locations in $\mathcal{O}$ within some small time step dt, then repeats the process T times. Alternatively, it is straightforward to apply the greedy algorithm to the case in which B_t is time-varying. Compared to the stationary case, instead of only giving a spatial location, each observation element of $\mathcal{O}$ has an additional

[2] Given P_t and G_t, we can calculate Eq. (9) in $O(NT)$ time, because $tr(P_t G_t P_t^T)$ can be computed in $O(N)$ time:

$$tr(P_t G_t P_t^T) = tr(L_t L_t^T) = sum(sq(L_t)),$$

where $L_t = P_t G_t^{1/2}$, $sq(\cdot)$ indicates squaring the argument component-wise, and we take advantage of the fact that G_t is $n \times n$ diagonal matrix with $n \ll N$.

[3] Our implementation of the algorithm is based on A. Krause's Matlab Toolbox for Submodular Function Optimization (Krause 2010) (available at http://www.cs.caltech.edu/~krausea/sfo/index.html).

temporal component and instead of optimizing N spatial locations, the algorithm optimizes over NT spatiotemporal locations (T is again the number of time steps in the observation sequence).

A slight modification to the greedy algorithm is necessary, however. In general, we want to enforce a limit of k observations at each time step because the measurement apparatus is limited to some finite number of (near) simultaneous observations. Therefore, after the k-th observation has been selected by the algorithm at time step t, the remaining observations at that time should no longer be considered. An incidental benefit of this constraint is that as the algorithm proceeds, many potential observation elements are eliminated from consideration, beyond those already eliminated by implementing lazy evaluation, which further speeds up the optimization. Empirically, we found the effect was particularly strong because the greedy algorithm tended to choose observations from the same time step consecutively. The same time steps were chosen because observations near $t = T/2$ provide the most information since this choice maximizes the information gained about past and future voltages; hence the greedy algorithm tended to start with observations near $t = T/2$ and work away from that time.

An additional speed-up—in the same spirit as the convergence of C_t trick used in the stationary B_t case—can be made to the Kalman filter in the case of time varying observations. If we begin by making a single observation at time t_1 then we can begin the forward recursion at that time and C_t^f will converge to C_0 at some future time t_1^f. For the backward smoothing, C_t^s will quickly converge, but will then need to be recalculated again beginning at t_1^f until it again converges for some time $t_1^s < t_1$. Once the greedy algorithm selects the best observation location at some time t_1, we can continue to add new observations in a similar manner. For some second observation at t_2, there are three cases. If $t_2 = t_1$ then we proceed exactly as before. If $t_2 < t_1$ there are two subcases: if $t_2^f < t_1^s$, then there is no interaction between the two observations on the forward and backward recursions. Otherwise $t_2^f \geq t_1^s$ and recalculation of the covariance matrices will be required for times $t_2^s < t < t_1^f$. The $t_2 > t_1$ case is similar.

4 Results

We applied the optimization methods described above to data simulated with two representative neuronal geometries: a rabbit starburst amacrine cell ("THINSTAR") and a rat hippocampal pyramidal cell ("c73164").[4] For the dynamics Eq. (2) we set $dt = 1$ msec and $\sigma^2 = 1$; for the observation Eq. (3) W_t was set to the identity matrix scaled by .005; for the cable Eq. (4), ranges for the intercompartmental coupling a_{xw} and the membrane conductance g_x were based on the geometry of the cell (including the compartment lengths and diameters) and empirically derived values for biophysical parameters (Koch 1999; Dayan and Abbott 2001). The number of time steps T was set to 20. The simulation parameters are summarized in Table 1.

We begin in Fig. 1 by examining the relative magnitudes of the prior variances (i.e., the diagonal elements of the equilibrium covariance matrix C_0). C_0 can be computed quite explicitly in the limit as $dt \to 0$. We define $K' = K/dt$, so K' does not have any dt dependence. Using Eq. (8) and the substitution $A = (I - K)^{-1} = (I - dtK')^{-1}$ gives

$$\begin{aligned}\lim_{dt\to 0} C_0 &= \lim_{dt\to 0} \sigma^2 dt(I - (I - dtK')^{-2})^{-1} \\ &= \lim_{dt\to 0} \sigma^2 dt(-2dtK' - 3dt^2(K')^2)^{-1} \\ &= \lim_{dt\to 0} -\sigma^2(2K' + 3dt(K')^2)^{-1} \\ &= -\frac{\sigma^2}{2}(K')^{-1}, \end{aligned} \tag{10}$$

where the second line follows from taking the second order Taylor approximation in dt. $(K')^{-1}$ in Eq. (10) has a natural interpretation as the transfer impedance matrix (Zador and Pearlmutter 1993). When we compute C_0 for the starburst amacrine and pyramidal cells (Fig. 1), we find that the maximum variance was approximately 60% larger than the minimum variance. The variance increases for compartments farther away from the soma, so compartments near the tips of the dendritic branches have significantly higher variance than those close to the soma, with the greatest variance at the ends of the tips.

Observing near the tips, therefore, has the potential to provide the largest total variance reduction. This is illustrated directly in Fig. 2, which shows the variances of a subtree of the pyramidal cell when zero, one, two and three observations are made. Note how the second observation, which is only two compartments away from a tip, has the greatest effect, not only reducing the variance of the observed compartments but also significantly decreasing the variance

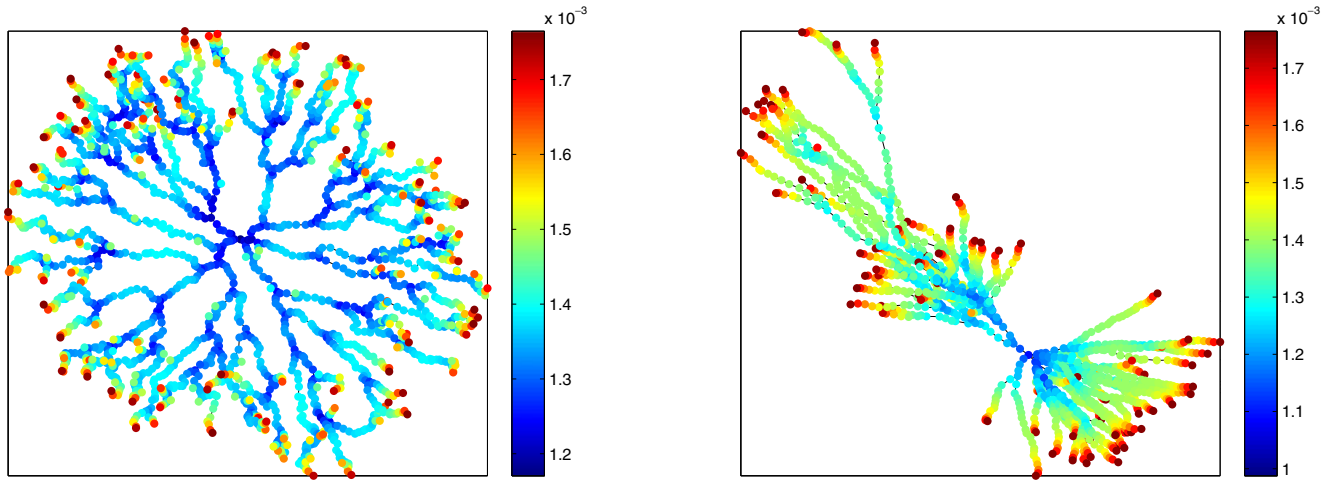
[4] Both geometries are available at http://neuromorpho.org (Ascoli 2006).

Table 1 Simulation parameters for Eqs. (2)–(4)

Parameter	Description	Value(s)
N	Number of compartments	1419 (c73164) and 2133 (THINSTAR)
T	Number of time steps	20
dt	Time step length (in milliseconds)	1
W_t	Scale of observation noise	.005
a_{xw}	Intercompartmental coupling	1250–5000 (c73164) and $1–4 \times 10^5$ (THINSTAR)
g_x	Membrane conductance	100

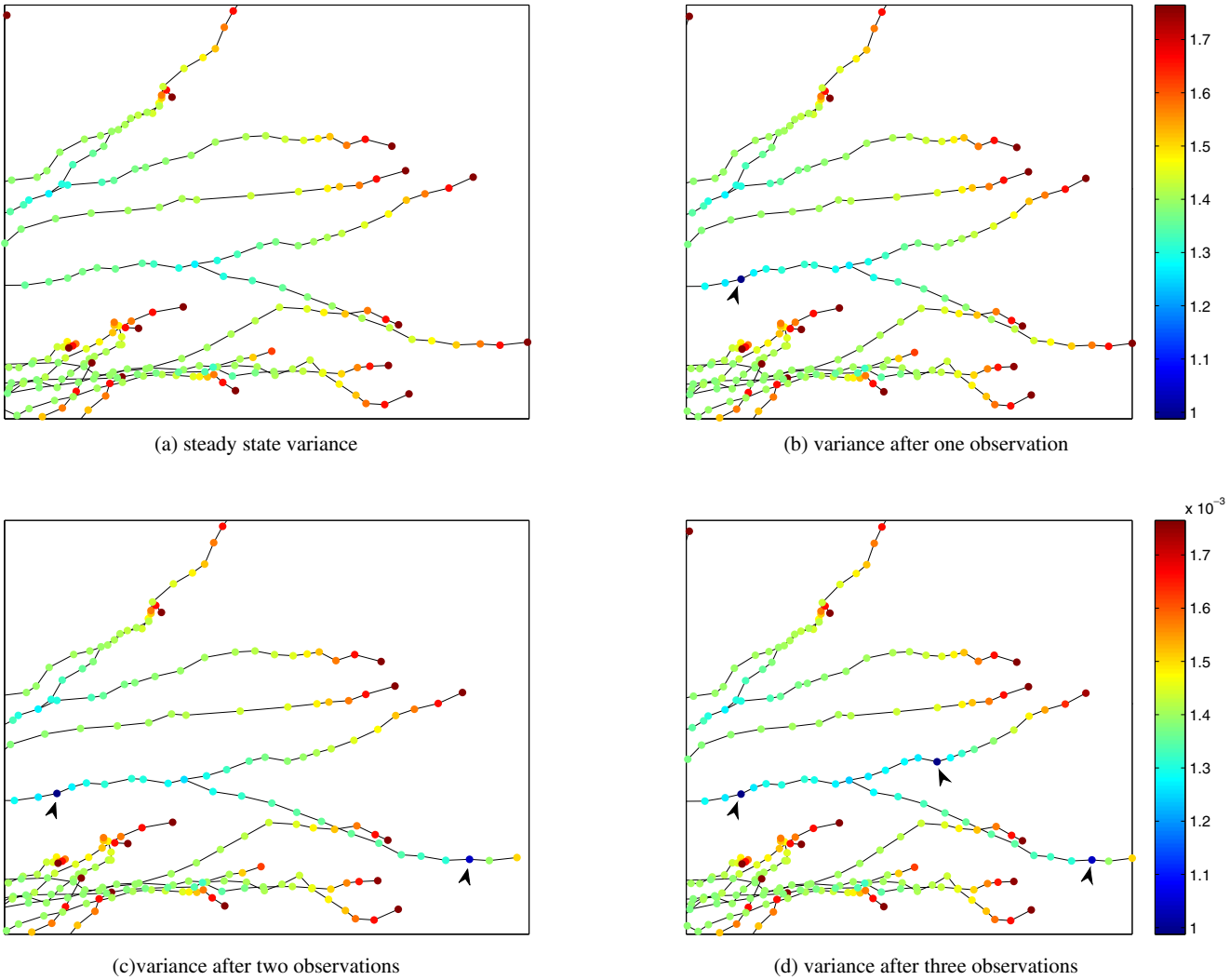
of neighboring compartments. This effect is less pronounced in the observations farther away from a tip. Figure 3 makes the pattern clear: the colors in each compartment in the second panel indicate the total variance reduction produced by making a single observation at that compartment. The total variance reduction is highest near, but not at, the tips (since by making observations not quite at the tips we can decrease the posterior variance of more compartments), so that is where we would expect the greedy algorithm to favor making observations. Figure 4 shows exactly this phenomenon. It plots the first 100 locations chosen by the lazy greedy algorithm for the starburst and pyramidal geometries. The locations are colored in an effort to indicate the order in which the locations were chosen by the algorithm. Some observation clumping is noticeable with the starburst amacrine cell. The clumping indicates that, because of the low SNR setting, multiple samples near the same location must be made in order for the smoother to make accurate predictions.

Figure 3(b) shows that the greatest *initial* variance reduction is produced by observations near the tips—exactly the locations sampled by greedy algorithm, as seen in Fig. 4. This observation motivated us to try a heuristic approximation to the full greedy algorithm using only these initial variance reductions. We were further motivated by Fig. 2 (cf. Fig. 9), which shows that observations only locally affect the variance of compartments, so the initial variance reduction should be a good approximation to the true variance reduction even for later iterations of the algorithm. The heuristic works as follows. First, calculate the initial variance reduction for each possible observation site. (Naively, this would take $O(N^2)$ time, since we need to compute



(a) THINSTAR (b) c73164

Fig. 1 The relative magnitudes of the prior variance for two neuronal geometries. The color of each compartment i indicates the corresponding prior variance, $[C_0]_{ii}$. Moving away from the soma, the variance increases. Thus, compartments near the tips of the dendritic branches have significantly higher variance than those close to the soma, with the greatest variance at the ends of the tips. The variance is also lower near sections with many branch points, a phenomenon particularly noticeable in the starburst amacrine cell. The neuronal geometry for panel (**a**) is taken from the "THINSTAR" file (rabbit starburst amacrine cell); see Bloomfield and Miller (1986) for details. The geometry for panel (**b**) is taken from the "c73164" file (rat hippocampal pyramidal cell); see Ishizuka et al. (1995) for details



(a) steady state variance

(b) variance after one observation

(c)variance after two observations

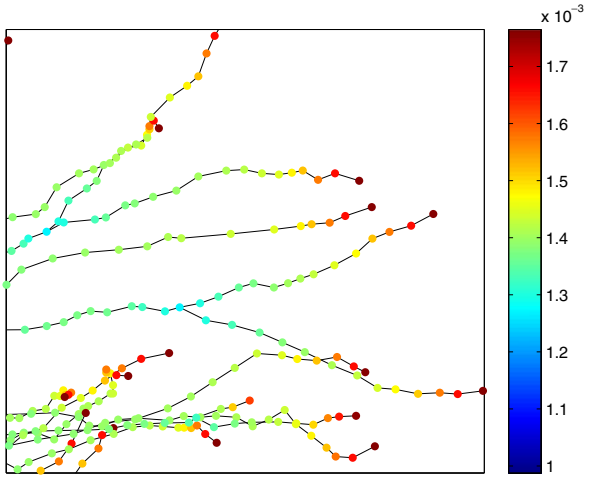
(d) variance after three observations

Fig. 2 Computing the variance reduction. (**a**) The prior variance on a sub-branch; this is a zoomed-in view of Fig. 1(b). (**b**)–(**d**) The posterior variance on the tree after one (**b**), two (**c**), and three (**d**) observation sites are selected. Coloring and scale in all subplots are as in Fig. 1(b). While each observation (the location is indicated by an *arrow*) provides a reduction in the variance in the observed compartment and its neighbors, the effect is strongest near the second observation, which was made near a dendritic tip. This result makes intuitive sense because the prior variances of compartments near the tips are significantly higher than those of other compartments

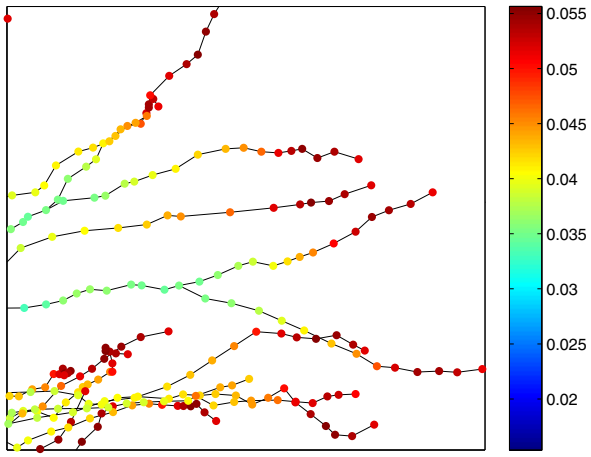
the reduction in variance at each compartment i, following an observation at each compartment j; however, since each observation only affects a few neighboring compartments, it is possible to perform this computation in just $O(N)$ time. See Paninski et al. (2011) for a more detailed discussion in a related application.) Next, select observations in order of this initial variance reduction (largest first), downweighting compartments that are close to any compartments which have already been selected (where the downweighting function at compartment i is chosen to match the spatially local dip in variance following an observation at i, as illustrated in Fig. 2). This heuristic produced results that were qualitatively identical to the original algorithm (Fig. 8), with much better computational scaling, as discussed in Section 4.3 below.

4.1 Including non-uniform weighting and variance terms

In many cases it may be physiologically desirable to bias the algorithm to favor observation locations closer to the soma. We discussed optimization methods using the weighted objective function Eq. (1); Fig. 6 illustrates



(a) steady state variance



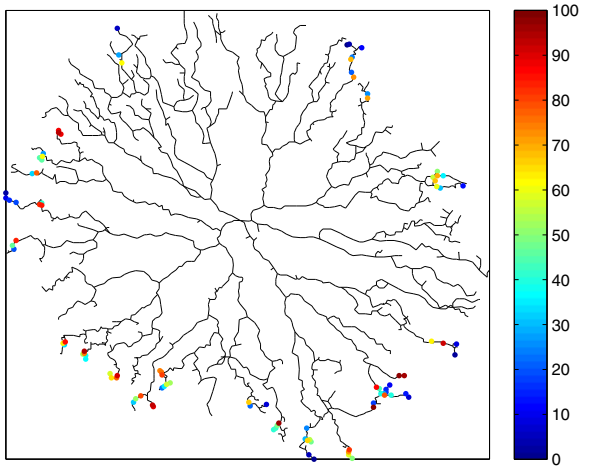
(b) total variance reductions

Fig. 3 (**a**) The prior variances of each compartment in a subtree of the pyramidal cell (cf. Fig. 1(b)). (**b**) The color in each compartment indicates the total variance reduction produced by making observations solely at that compartment. Compared with the prior variances in (**a**), the maximum variance reductions are not at the tips of the branches, but near the tips, since observing at a tip effectively provides information about fewer compartments than does observing near the tip

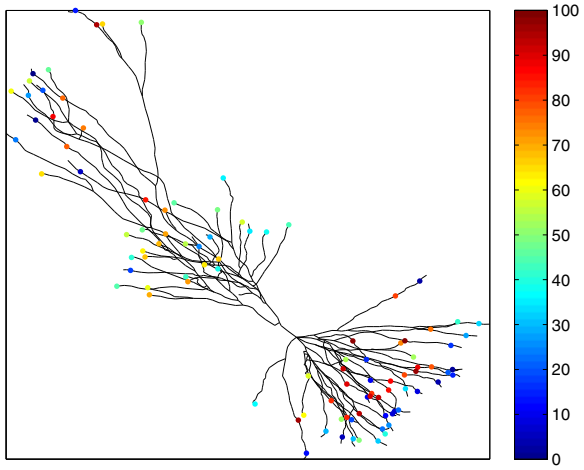
how an exponential weighting function might affect the output of the greedy algorithm. The weighting is a function of the distance $d(i)$ along the tree between the observation location i and the soma. The exponential weighting term took the following form:

$$w_{\exp}(i) = w_{\min} + (w_{\max} - w_{\min}) \exp\left(-\alpha \frac{d(i)}{d_{\max}}\right), \qquad (11)$$

where $w_{\max}$ is the maximum weighting and $w_{\min}$ is a lower bound on the weighting; $d_{\max}$ is the maximum distance from the soma (75 for THINSTAR, 71 for c73164); and α controls the strength of the bias toward the soma. Similar results were observed for other (non-exponential) weighting functions. Because there is an implicit free scaling parameter we can parameterize $w_{\min}$ and $w_{\max}$ by their ratio: $w_r \equiv \frac{w_{\max}}{w_{\min}}$. We tried a variety of values for both w_r and α and found that they

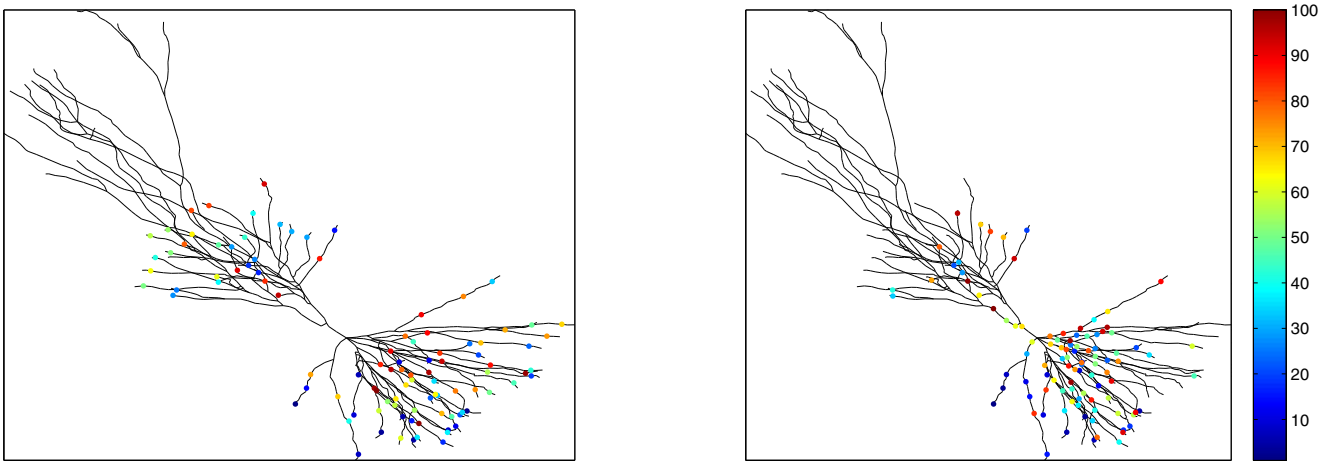


(a) THINSTAR



(b) c73164

Fig. 4 Sampling scheme generated by greedily selecting 100 observation locations. The colors—beginning with *dark blue* and ending with *dark red*—indicate the order in which the locations were selected by the greedy algorithm. Note that the algorithm preferentially samples from the periphery of the dendritic tree, as predicted by Fig. 3



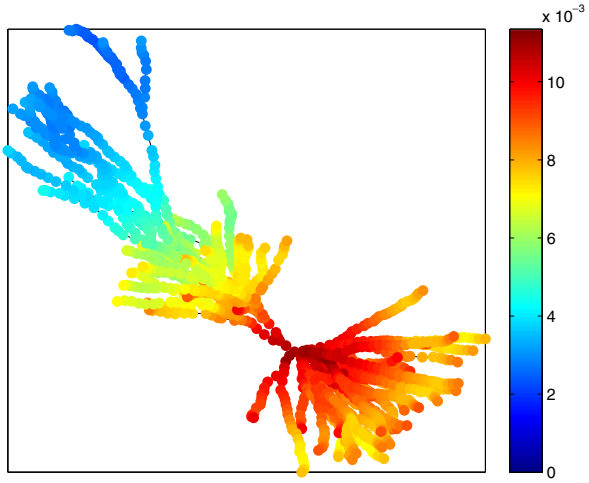
(a) α= 4 (b) α= 10

Fig. 5 First 100 observation locations selected by the greedy algorithm when an exponential weighting term Eq. (11) is applied to the variance reduction. For $\alpha = 4$, the observation locations are qualitatively similar to the unweighted case, except they tend toward the soma a bit more. There is no longer any sampling along the long branches reaching up into the upper lefthand corner, yet there are also no observations very near the soma. The trend continued for larger α values. For $\alpha = 10$ there was a growing bias toward the soma and locations very near the soma were selected earlier. This trend of earlier selection suggests that when fewer observations will be made a stronger weighting bias would be necessary to induce the desired sampling locations. Similar trends were found when w_r was varied (data not shown)

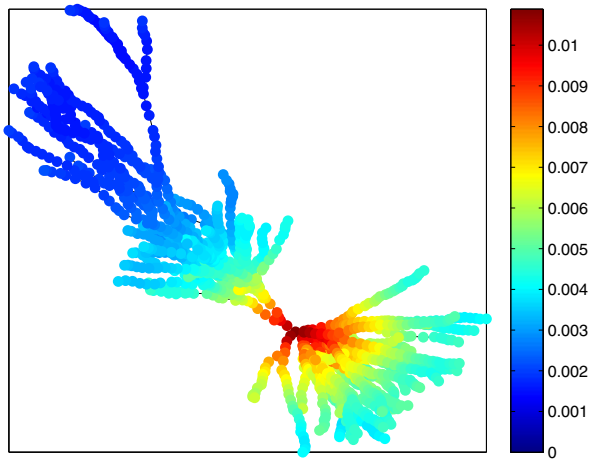
had similar effects on the qualitative output. Thus, we will restrict our discussion to the effect of α on the results and fix $w_r = 10$.

As expected, increasing α biased observations toward clustering near the soma more. Yet even with $\alpha = 10$ there are still many observations near the tips of the dendrites (Fig. 5). As α increases the greedy algorithm also tends to choose locations close to the soma earlier. Weighting thus provides a flexible and effective way to bias the sampling scheme toward the parts of the dendrite that are of greatest interest. Figure 6 provides some insight into this behavior: though the variances of compartments near the soma are weighted much more highly than those on the periphery, the relative strength of the variance reduction near the periphery still maintains a (weak) bias toward the periphery, even for large values of the weighting strength α.

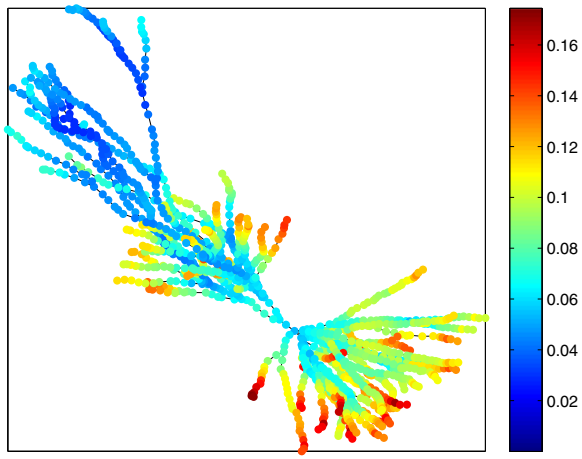
Many voltage imaging techniques make noisier measurements near the tips of dendrites than near the soma because of the reduction in dendrite circumference as a function of distance from the soma, leading to a reduction in SNR (which is proportional to the membrane area within the image focus). We explored this effect by varying the observation noise variance W_t as a function of the distance from the soma. Figure 7 shows the results of a simulated experiment in which W_t grows linearly with the distance from the soma. We find that indeed, higher peripheral noise leads to more samples at the soma. Somewhat more surprisingly, as the ratio of the noise variance at the dendrite tips to that at the soma increases, the optimized observation locations begin to appear together, in a clumpy manner. These results can be explained by the interaction of two competing factors. First, as already discussed, peripheral observations are the most informative, so peripheral locations are preferred even if they have slightly higher noise. However, because of the higher noise, each observation provides much less information, so multiple observations near the same location are required to obtain a comparable SNR level. The second factor is that higher noise at the periphery leads to more observations near the soma. Eventually the latter factor overwhelms the former as the ratio between the noise and the soma and the periphery is increased and all the observations clump around the soma (as seen in Fig. 7). Also recall that these simulations are performed in a low-SNR setting, where many samples may be taken from the same location with limited redundancy (since each observation provides only a limited amount of information); in experiments with higher SNR (i.e., smaller values of the observation noise W_t), this clumping of the observation locations was reduced (since in the higher-SNR setting, multiple observations from the same location provide



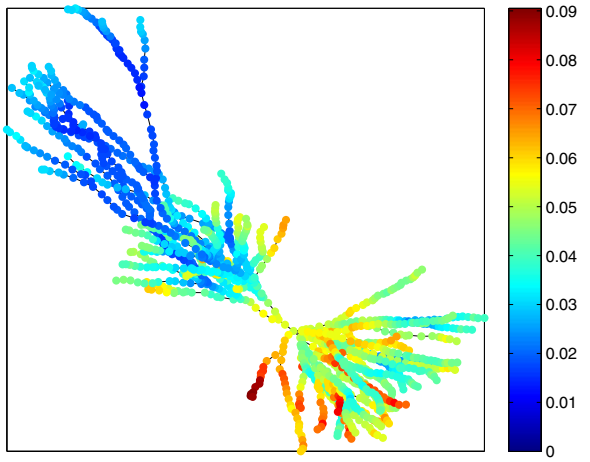
(a) exponentially weighted prior variance: $\alpha = 4$



(b) exponentially weighted prior variance: $\alpha = 10$



(c) exponentially weighted variance reduction: $\alpha = 4$



(d) exponentially weighted variance reduction: $\alpha = 10$

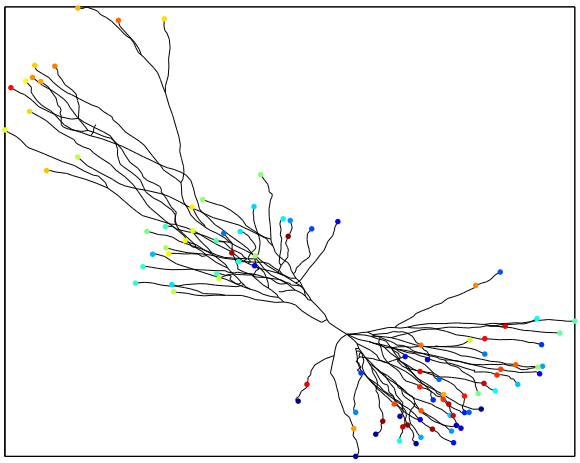
Fig. 6 *Top*: exponentially weighted prior variance for two settings of the bias term α from Eq. (11). *Bottom*: initial variance reductions using exponential weighting for two settings of the bias term α. The colorings are qualitatively very different from the weighted variances shown in the top row. The differences between the two figures indicate how much greater the variance reduction near the periphery is compared to the reduction at the soma, since it overwhelms the weighting function, even for a large bias term

redundant information, and an optimal sampler will prefer to observe many distinct locations instead of sampling repeatedly from nearby locations; data not shown).

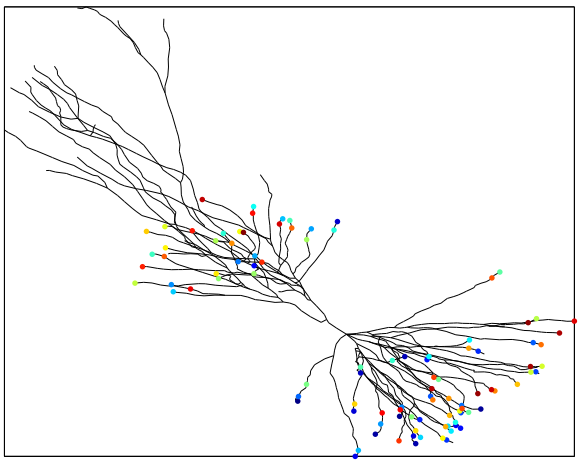
4.2 Quantitative results

Figure 8 compares the performance of the greedy algorithm against a baseline method for selecting observation locations (random location sampling) as well as the heuristic approximation described previously. Other methods such as selecting evenly spaced locations were also tried, but produced results comparable to choosing locations randomly, so we have not included them in our results. The variance reductions for the random method were averaged over 15 trials, so error bars are included for those results.

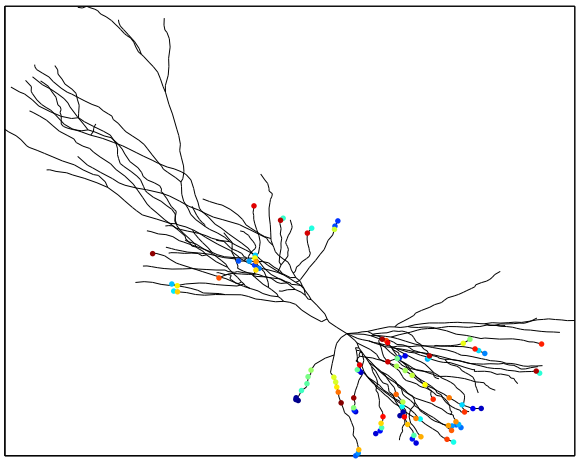
For the starburst geometry, improvements over randomly selected and evenly spaced locations ranged from 60% for small $k \sim 10$ (recall that k denotes the number of observations per time step) to 30% for large $k \sim 100$. For the pyramidal geometry, improvements



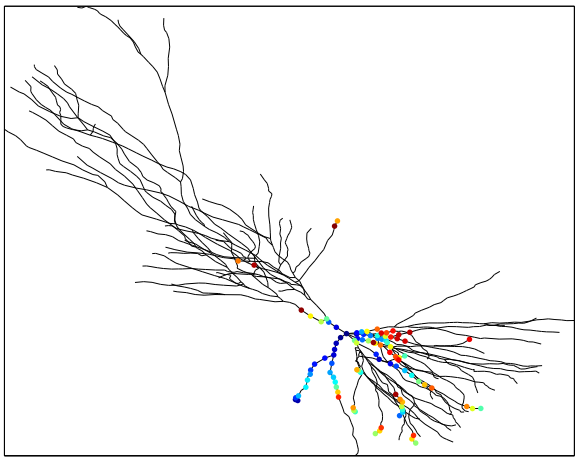
(a) max/min noise ratio 2:1



(b) max/min noise ratio 5:1



(c) max/min noise ratio 10:1

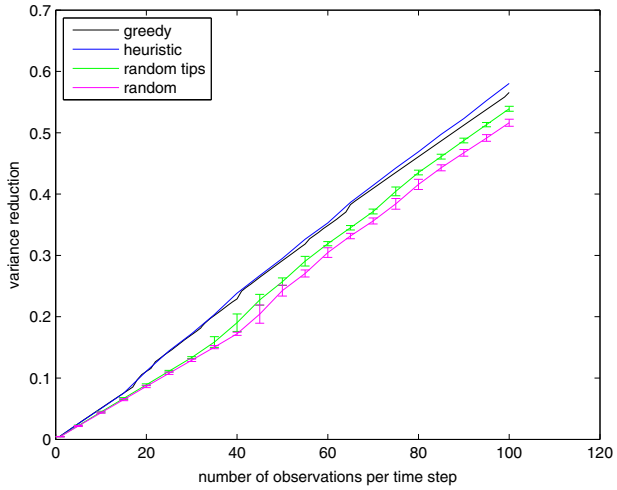


(d) max/min noise ratio 25:1

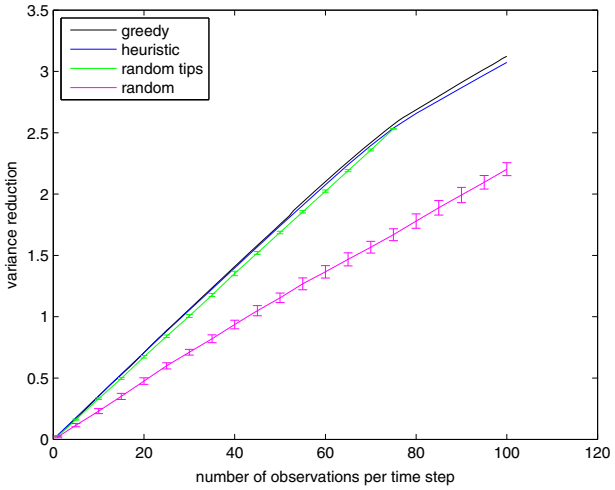
Fig. 7 100 observation locations selected by the greedy algorithm when the observation noise varied linearly with the distance of the compartment from the soma. The non-constant noise leads to clumping in the observation selection scheme. For smaller ratios the need for multiple observations near the same point due to increased noise produces adjacent observations on the periphery. As the ratio increases the higher noise levels on the periphery pushes the observations towards the soma

ranged from almost 100% to 55% over a similar range of k values. Based on the results of the greedy optimization—namely, that locations on the periphery were heavily favored for sampling—we decided to compare the performance of the greedy algorithm to the method of observing a random sample of the tips of the branches. The greedy algorithm was only 8% better for small k and negligibly better for larger k. Of course, sampling at the tips will no longer be optimal in the case of strongly varying weights or noise variances, as discussed in the preceding section; both the full greedy method and the faster heuristic are sufficiently general to cover these cases.

As expected based on the qualitatively similar outputs of the two algorithms, the heuristic performed as well as, and even slightly better than, the original greedy algorithm, suggesting that the original intuitions that motivated the heuristic were well-founded. These intuitions extend to the time-varying case, as seen in Fig. 9, which shows that the effect of each observation is local both in space and in time. Thus, it was natural to implement a non-stationary heuristic approximation along exactly the same lines as the stationary heuristic we have already described. The resulting nonstationary heuristic algorithm outperformed the stationary versions of both the greedy and heuristic algorithms, and



(a) THINSTAR

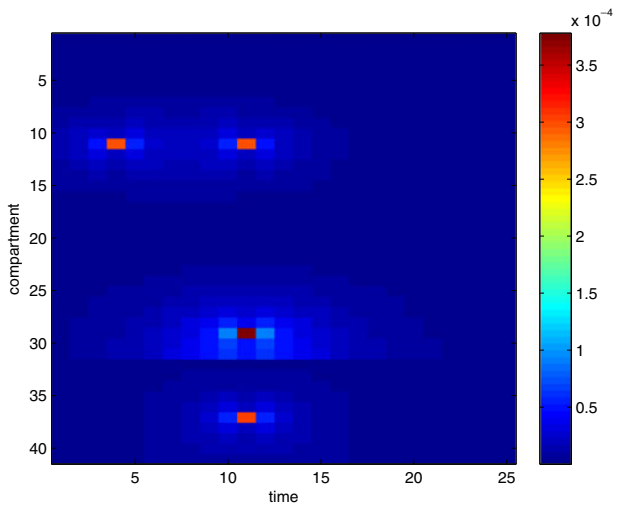


(b) c73164

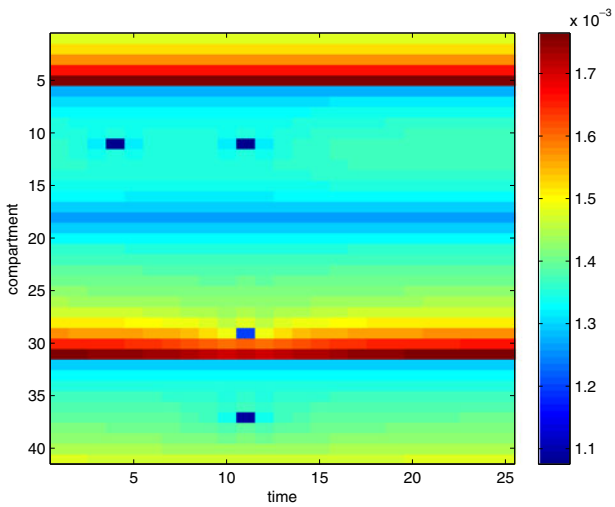
Fig. 8 Graph of the variance reduction vs. number of observations for the four observation location selection methods and two neuronal geometries. The *black line* is based on observations selected by the greedy algorithm. The *blue line* is for the time-stationary version of the heuristic that uses the variance reductions produced by the first iteration of the algorithm. The *green line* is the average of 15 trials of randomly selected observations at the tips of the dendritic branches. The *magenta line* is the average of 15 versions in which the observations were chosen at random; *error bars* indicate ± one standard deviation. The heuristic effectively reproduced the results of the full greedy algorithm at much lower computational cost due to the local effect of observations on the variance

was much faster than the full greedy method in the nonstationary case, as we discuss further below.

While the greedy algorithm led to relatively minor improvements over naïve methods in the time-stationary setting, larger improvements were visible in the non-stationary sampling case (Fig. 10). For THINSTAR (c73164), the non-stationary version had variance reductions 15–40% (60–90%) higher than the stationary version and 25–40% (30–90%) higher than non-stationary naïve methods. However, because



(a) Variance reduction from four observations



(b) Posterior variance from four observations

Fig. 9 Illustration of non-stationary sampling. We sampled sparsely in time at the three compartments that were observed in Fig. 3: we sampled just at one time point in two compartments and twice in the other. (**a**) The variance reduction in each compartment over time caused by taking the four samples. The observation from compartment 29, which is near a tip, has a significantly larger effect than the other three observations, as we have seen in the preceding figures illustrating the stationary sampler. (**b**) The posterior variance of each compartment after sampling

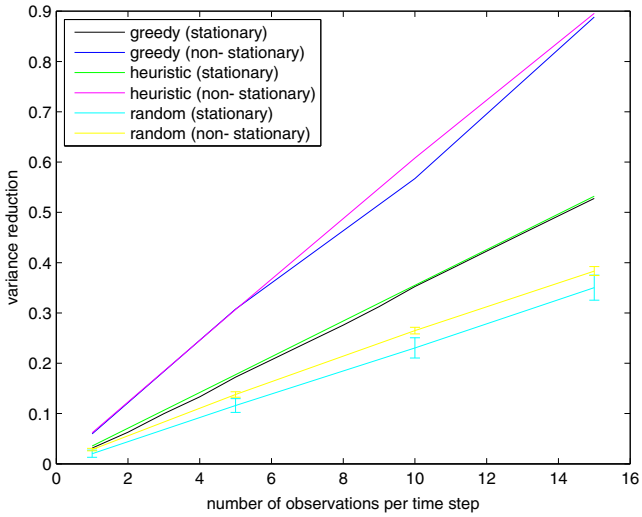


Fig. 10 Comparison of the variance reduction produced by using the full and heuristic greedy optimizations, with and without time variation in the sampling scheme, compared against random sampling. Allowing for non-stationary observations increased performance over the stationary case by 60–90% for c73164 and 15–40% for THINSTAR (data not shown), with larger benefits for smaller numbers of observations. While the non-stationary optimization using the full greedy was an order of magnitude slower, the non-stationary heuristic version still ran in under an hour, even for 100 observations per time step (data not shown here) and performed as well as the full greedy method

allowing time-varying B_t slows the optimization, we only ran the algorithm for k up to 15 (still with $T = 20$). This shortcoming does not extend to the non-stationary heuristic, which performed as well as the non-stationary full greedy algorithm but was scalable up to $k \gg 10$.

4.3 Computational requirements

Using lazy evaluation in the greedy algorithm proved to be crucial to making the optimization tractable. For THINSTAR, for example, after the initial iteration, when all $N = 2133$ compartments must be evaluated, each subsequent time step had on average lazy 28 evaluations while the non-lazy version requires $2134 - i$ on the i-th iteration, an improvement of about two orders of magnitude. The benefit is even greater than these numbers indicate, since the cost per evaluation for the first iteration is much less than for later iterations. This is because, while the bottleneck on the number of compartments N has been removed by using a low rank correction to calculate C_t^s, some $O(n^3)$ and $O(nN)$ operations remain, where n is proportional to the number of observations that are made at each time step. Thus, the true speed-up is closer to three orders of magnitude. Without lazy evaluation the greedy algorithm would be far too slow to be practical.

The cubic scaling of the computation time as a function of the number of observations k proved to be problematic for larger values of k. For $k = 10$ the runtime was 10 minutes and for $k = 30$ the runtime was 1.5 hours, while for $k = 100$ the runtime jumped to almost 2 days. If time-varying observation schemes were allowed a more severe version of the same trend was observed. For $k = 10$ the runtime was about the same as for the time-invariant case, but jumped to 20 hours for $k = 30$. (All timing experiments used the THINSTAR neuronal geometry and were run in MATLAB on Linux machines, each with 2 2.66 GHz Quad-Core Intel Xeon processors and 16 GB RAM.) The time varying implementation probably remains impractical without either an efficient parallelized implementation (which, it is worth noting, should be fairly straightforward here—since we can easily evaluate each of the top 32 experimental designs in parallel, for example—though we have not yet explored this direction systematically), or spatial downsampling or model reduction techniques such as those explored in Kellems et al. (2009).

These computational limitations are all overcome by using the heuristic approximation, which scales as $O(N)$, as discussed above; furthermore, parallelization techniques can also be applied easily to the heuristic method, since the initial variance reductions can be trivially evaluated in parallel. Thus we conclude that the heuristic algorithm provides a stable and fast method for computing good sampling designs, both in the stationary and non-stationary settings.

4.4 Non-submodularity of the variance reduction

Empirically we found that $\rho(\cdot)$ is not quite submodular. To see this, consider the sequence $\boldsymbol{\rho} = (\rho_1, \rho_2, \ldots)$ of the best variance reductions calculated by the lazy greedy algorithm for increasing k values and define the difference function on a vector $\mathbf{v} = (v_1, v_2, v_3, \ldots)$ as

$$\Delta\mathbf{v} = (v_2 - v_1, v_3 - v_2, v_4 - v_3, \ldots).$$

Then a necessary (but not sufficient) condition for $\rho(\cdot)$ to be submodular is that all elements of the second order difference, $\Delta^2\boldsymbol{\rho} := \Delta\Delta\boldsymbol{\rho}$, are non-positive. That is, each observation we add should provide a smaller increase in $\rho(\cdot)$ than the previous observation. For the starburst geometry we found that this condition only held 60% of the time (although it always held for the pyramidal geometry). However, if we relax the condition to allow slight violations, it held 85% of the time. Specifically, instead of requiring $\Delta^2\boldsymbol{\rho} \leq 0$, we permit $\Delta^2\boldsymbol{\rho}/\Delta\boldsymbol{\rho} \leq .1$, where division and comparison are done component-wise. That is, we ignore increases in the rate of variance reduction decrease that are less than 10% of

the current rate of decrease. This analysis suggests that most of the submodularity violations are fairly small.

Recall that the validity of lazy evaluation is contingent upon the submodularity of the objective function. If the objective function is not submodular then using lazy evaluation could decrease the quality of the optimization. We ran the greedy algorithm without lazy evaluation and found that the decrease in performance was negligible, costing less than 1% for the starburst geometry and less than .01% for the pyramidal geometry (in both cases for $1 \leq k \leq 30$, where k is the number of observations per time step).

5 Conclusion

We have presented a state-space filter framework for efficiently inferring and smoothing voltage measurements across large dendritic trees and for designing optimal voltage sampling schemes. This work extends (Paninski 2010), which considered only the forward dendritic Kalman filter and did not explore experimental design issues. Our low-rank perturbation methods allow for efficient computation of the smoothed covariance, which can be used to calculate a number of measures of experimental optimality. We have shown how to design an optimal sampling scheme using one such metric, the variance reduction, via a greedy algorithm with lazy evaluation.

Somewhat surprisingly, in the simplest case of spatially-constant noise, variance weighting, and stationary observation sets, the optimal greedy algorithm can be well-approximated by a much simpler algorithm in which we randomly select observations from the tips of the dendrites (Fig. 8). More generally, a heuristic approximation to the greedy algorithm which uses only the first iteration of the variance reductions performs as well as the full greedy approach, due to the local effects of each observation. This heuristic produces an order of magnitude speed increase, making the proposed methods potentially tractable in experimental settings in which dendritic reconstructions may be obtained in living preparations (Losavio et al. 2008; Cuntz et al. 2010).

Because we rely on the steady-state covariance C_0, our procedure is so far limited to the case of time-invariant dynamics. In addition, we have assumed that both dynamics and observation noise in the dendrite is Gaussian. Generalizations of all of these assumptions seem feasible. For example, extended Kalman filter or hybrid particle filter methods (Doucet et al. 2001) may be applicable to the case of active dendritic membranes (Huys and Paninski 2009), where the dynamics depend strongly on the recent voltage history. Non-Gaussian observation noise could in some cases be handled using the Laplace approximation methods discussed in Paninski et al. (2010). A final promising direction is to consider banded approximations of C_t (Pnevmatikakis et al. 2011), instead of the low-rank approximation we have exploited here; such a banded approximation would be appropriate whenever both the dynamics and observation matrices are sparse and local, and may be applied even in the context of high-SNR observations of a time-varying dynamics model.

Acknowledgements JHH is supported by the Columbia College Rabi Scholars Program. LP is supported by a McKnight Scholar award and an NSF CAREER award. Computation performed on the Columbia Hotfoot HPC Research Cluster. We thank K. Rahnama Rad and C. Guestrin for helpful conversations.

Appendix A: Derivation of the fast backward Kalman smoother

We will derive a procedure for efficiently calculating the smoothed covariance $C_t^s = Cov(V_t|Y_{1:T})$ and mean $\mu_t^s = E(V_t|Y_{1:T})$ in $O(N)$ time, where N is the number of dendritic compartments. We will follow the same general strategy as Paninski (2010). As with the forward covariance, our goal is to express the smoothed covariance as a low-rank perturbation to C_0 of the form

$$C_t^s \approx C_0 + P_t G_t P_t^T.$$

In general, the smoothed covariance is described by the backwards recursion (Shumway and Stoffer 2006)

$$C_t^s = C_t^f + J_t[C_{t+1}^s - C(V_{t+1}|Y_{1:t})]J_t^T, \tag{12}$$

where

$$\begin{aligned} J_t &= C_t^f A^T [C(V_{t+1}|Y_{1:t})]^{-1} \\ &= C_t^f A^T (AC_t^f A^T + \sigma^2 dtI)^{-1} \\ &= C_t^f A^T (A(C_0 + U_t D_t U_t^T)A^T + \sigma^2 dtI)^{-1} \\ &= C_t^f A^T (C_0 + AU_t D_t U_t^T A^T)^{-1} \\ &= C_t^f A^T (C_0^{-1} - C_0^{-1} AU_t([D_t]^{-1} \\ &\qquad + U_t^T A^T C_0^{-1} AU_t^T)^{-1} U_t^T A^T C_0^{-1}) \\ &= C_t^f A^T (C_0^{-1} - R_t Q_t R_t^T), \end{aligned} \tag{13}$$

and where we have used the fact that $C_0 = AC_0A^T + \sigma^2 dtI$ to obtain the fourth line and the Woodbury lemma to obtain the fifth line, then abbreviated

$$R_t = C_0^{-1}AU_t$$

and

$$Q_t = (D_t^{f-1} + U_t^T A^T C_0^{-1} AU_t^T)^{-1}.$$

As in the $U_tD_tU_t^T$ case, Q_t is small and $R_tQ_tR_t^T$ is low-rank, so both matrices can be manipulated efficiently.

First we must re-express $C_{t+1}^s - C(V_{t+1}|Y_{1:t})$ from Eq. (12) as a low-rank matrix of the form $W_tH_tW_t^T$:

$$\begin{aligned} &C_{t+1}^s - C(V_{t+1}|Y_{1:t}) \\ &\quad = C_{t+1}^s - (AC_t^fA^T + \sigma^2 dtI) \\ &\quad = C_{t+1}^s - (A(C_0 + U_tD_tU_t^T)A^T + \sigma^2 dtI) \\ &\quad = C_0 + P_{t+1}G_{t+1}P_{t+1}^T - C_0 - AU_tD_tU_t^TA^T \\ &\quad = P_{t+1}G_{t+1}P_{t+1}^T - AU_tD_tU_t^TA^T. \end{aligned}$$

In order to express $P_{t+1}G_{t+1}P_{t+1}^T - AU_tD_tU_t^TA^T$ as a single low rank matrix, we choose an orthogonal basis for the two matrices

$$W_t = orth([P_{t+1} \quad AU_t])$$

and write

$$\begin{aligned} C_{t+1}^s - C(V_{t+1}|Y_{1:t}) &= P_{t+1}G_{t+1}P_{t+1}^T - AU_tD_tU_t^TA^T \\ &= W_tH_tW_t^T, \end{aligned} \tag{14}$$

where

$$H_t = W_t^TP_{t+1}G_{t+1}P_{t+1}^TW_t - W_t^TAU_tD_tU_t^TA^TW_t.$$

Our next task is to expand the second term in Eq. (12) and use orthogonalization to condense the resulting sum of low rank matrices. Generally suppressing time variation for notational clarity, substituting Eqs. (5), (13), and (14) into Eq. (12) gives

$$\begin{aligned} C_t^s &= C_t^f + J_t[C_{t+1}^s - C(V_{t+1}|Y_{1:t})]J_t^T \\ &= C_t^f + (C_0 + UDU^T)A^T(C_0^{-1} - RQR^T)WHW^T \\ &\quad \times (C_0^{-1} - RQR^T)A(C_0 + UDU^T). \end{aligned} \tag{15}$$

Expanding the inner part of the second term on the right hand side gives

$$\begin{aligned} &(C_0^{-1} - RQR)WHW^T(C_0^{-1} - RQR) \\ &\quad = (C_0^{-1} + L)X(C_0^{-1} + L) \\ &\quad = C_0^{-1}XC_0^{-1} + LXL + C_0^{-1}XL + LXC_0^{-1}, \end{aligned}$$

where we abbreviate

$$L = -RQR^T$$

and

$$X = WHW^T.$$

As above we can choose an orthogonal basis for the four low rank matrices $(C_0^{-1}XC_0^{-1} + \cdots + LXC_0^{-1})$, making sure to orthogonalize "thin" matrices so the computation is efficient,

$$O_1 = orth([C_0^{-1}W \quad R]),$$

and write

$$C_0^{-1}XC_0^{-1} + LXL + C_0^{-1}XL + LXC_0^{-1} = O_1M_1O_1^T.$$

Now we can substitute the condensed inner term

$$(C_0^{-1} - RQR)WHW^T(C_0^{-1} - RQR) = O_1M_1O_1^T$$

back into Eq. (15) and expand to get

$$\begin{aligned} C_t^s &= C_t^f + (C_0 + UDU^T)A^T(O_1M_1O_1^T)A \\ &\quad \times (C_0 + UDU^T) \\ &= C_t^f + (C_0 + \Omega)A^T\Theta A(C_0 + \Omega) \\ &= C_0 + \Omega + \Omega A^T\Theta AC_0 + C_0A^T\Theta A\Omega \\ &\quad + \Omega A^T\Theta A\Omega + C_0A^T\Theta AC_0, \end{aligned}$$

abbreviating

$$\Omega = UDU^T$$

and

$$\Theta = O_1M_1O_1^T.$$

Again, we can find an orthogonal basis O_2 for the sum of low rank matrices $\Omega + \cdots + C_0A^T\Theta AC_0$ and express C_t^s in the form $C_0 + O_2M_2O_2^T$, where

$$O_2 = orth([U \quad C_0A^TO_1])$$

and

$$\begin{aligned} M_2 = O_2^T(&\Omega + \Omega A^T\Theta AC_0 + C_0A^T\Theta A\Omega \\ &+ \Omega A^T\Theta A\Omega + C_0A^T\Theta AC_0)O_2. \end{aligned}$$

We obtain P and G by truncating $O_2 M_2 O_2^T$ in order to control its rank. In Matlab we would do

$$[P', G'] = svd(O_2 M_2^{1/2}, \text{'econ'}),$$

then choose P as the first n columns of P' and G as the square of the first n diagonal elements of G'. We determine n by capturing some large proportion c of the variance in $O_2 M_2 O_2^T$. That is, n is the least solution of the inequality

$$\sum_{i \le n} G_{ii}^2 \ge c \sum_i G_{ii}^2.$$

The proportion c was typically 0.99 or greater in the experiments described here. The exact value does not measurably impact the experimental results. To see why the SVD of $O_2 M_2^{1/2}$ allows us to reconstruct $O_2 M_2 O_2^T$, consider that the SVD produces:

$$O_2 M_2^{1/2} = P' G' X',$$

where G' is diagonal and P' and X' are orthogonal. Thus,

$$\begin{aligned} O_2 M_2 O_2^T &= (O_2 M_2^{1/2})(O_2 M_2^{1/2})^T \\ &= (P' G' X)(P' G' X')^T \\ &= P' G' X' X'^T G' P'^T \\ &= P' (G')^2 P'^T \end{aligned}$$

since X' is orthogonal.

At this point computing the backwards recursion for the smoothed mean is straightforward (Shumway and Stoffer 2006):

$$\begin{aligned} \mu_t^s &= \mu_t^f + J_t(\mu_{t+1}^s - A\mu_t^f) \\ &= \mu_t^f + (C_t^f A^T (C_0^{-1} + L))(\mu_{t+1}^s - A\mu_t^f) \\ &= \mu_t^f + ((C_0 + \Omega) A^T (C_0^{-1} + L))(\mu_{t+1}^s - A\mu_t^f) \\ &= \mu_t^f + (A^T + \Omega A^T C_0^{-1} + C_0 A^T L + \Omega A^T L) \\ &\quad \times (\mu_{t+1}^s - A\mu_t^f) \end{aligned}$$

Note that the base case of the recursion is

$$\mu_T^s = \mu_T^f \text{ and } C_T^s = C_T^f$$

so

$$P_T = U_T \text{ and } G_T = D_T.$$

Appendix B: Efficient computation of the mutual information

Here we briefly sketch the computation of the mutual information $I(V_{1:T}; Y_{1:T})$, which can be written as usual in terms of a difference between prior and conditional entropies,

$$I(V; Y) = H(V) - H(V|Y).$$

While we have not explored this objective function in depth in this work, as noted in the main text the information comes equipped with an attractive submodularity property in the important special case that all of the observations are conditionally independent of V. Thus it may be useful to explore this and alternate objective functions in the future, and so we provide a brief discussion of this function here, for completeness.

Because we are only interested in relative changes in $I(V; Y)$, we can ignore the prior entropy $H(V)$ term (which does not depend on the observations Y, or on the chosen observation sequence $\mathcal{O}$). To calculate the conditional entropy $H(V|Y)$ recall that that $p(V|Y)$ is Gaussian (since $p(V)$ is Gaussian and $p(Y|V)$ is linear-Gaussian), and therefore $H(V|Y)$ reduces to the computation of the determinant of the conditional covariance matrix $Cov(V_{1:T}|Y_{1:T})$. Note that this is not the same as the smoothed covariance matrix C_t^s, which is of dimension $N \times N$; $Cov(V_{1:T}|Y_{1:T})$ has dimension $NT \times NT$, and contains the smoothed covariance matrices C_t^s along its block-diagonal. While efficient algorithms are available for the computation of $|Cov(V_{1:T}|Y_{1:T})|$ (exploiting the fact that the inverse of this matrix has a convenient block-tridiagonal form (Paninski et al. 2010)), it is slightly easier to sidestep this issue by using the identity

$$\begin{aligned} \log p(Y) &= \log \int p(V, Y) dV \\ &= \log p(\hat{V}) + \log p(Y|\hat{V}) \\ &\quad + \frac{1}{2} \log |Cov(V|Y)| + const. \end{aligned}$$

for the marginal likelihood $p(Y)$ of any observed data sequence Y in this Gaussian model; here $\hat{V}$ abbreviates the conditional expectation $\hat{V} = E(V_{1:T}|Y_{1:T})$, which may be computed efficiently using the recursion described in the preceding section. Once we have $\hat{V}$, we can simply plug in to compute $\log p(\hat{V})$ and $\log p(Y|\hat{V})$. Finally, $p(Y) = \int p(V_T, Y_{1:T}) dV_T$ may be computed

using the standard forward recursion for the Kalman filter, interpreted as a Gaussian hidden Markov model (Rabiner 1989), which in turn may be implemented here in a straightforward and efficient manner exploiting the low-rank nature of the forward covariances C_t^f. Once all of these pieces are computed, we may use the above equation to obtain the determinant term $|Cov(V_{1:T}|Y_{1:T})|$, and therefore compute the objective function $I(V; Y)$.

References

Ascoli, G. (2007). Mobilizing the base of neuroscience data: The case of neuronal morphologies. *Nature Reviews Cancer, 7*(4), 318–324.

Bell, J., & Craciun, G. (2005). A distributed parameter identification problem in neuronal cable theory models. *Mathematical Biosciences, 194*(1), 1–19.

Bloomfield, S. A., & Miller, R. F. (1986). A functional organization of on and off pathways in the rabbit retina. *Journal of Neuroscience, 6*(1), 1–13.

Brockwell, P., & Davis, R. (1991). *Time series: Theory and methods*. Springer.

Canepari, M., Popovic, M., Vogt, K., Holthoff, K., Konnerth, A., Salzberg., B. M., et al. (2011). Imaging submillisecond membrane potential changes from individual regions of single axons, dendrites and spines. In M. Canepari, & D. Zecevic (Eds.), *Membrane potential imaging in the nervous system* (pp. 25–41). New York: Springer. ISBN 978-1-4419-6558-5.

Canepari, M., Willadt, S., Zecevic, D., & Vogt, K. E. (2010). Imaging inhibitory synaptic potentials using voltage sensitive dyes. *Biophysical Journal, 98*(9), 2032–2040.

Chaloner, K., & Verdinelli, I. (1995). Bayesian experimental design: A review. *Statistical Science, 10*, 273–304.

Cover, T., & Thomas, J. (1991). *Elements of information theory*. New York: Wiley.

Cox, S., & Griffith, B. (2001) . Recovering quasi-active properties of dendrites from dual potential recordings. *Journal of Computational Neuroscience, 11*, 95–110.

Cox, S. J., & Raol, J. H. (2004). Recovering the passive properties of tapered dendrites from single and dual potential recordings. *Mathematical Biosciences, 190*(1), 9–37.

Cuntz, H., Forstner, F., Borst, A., & Häusser, M. (2010). One rule to grow them all: A general theory of neuronal branching and its practical application. *PLoS Computers in Biology , 6*(8), e1000877, 08. doi:10.1371/journal.pcbi.1000877.

Das, A., & Kempe, D. (2008). Algorithms for subset selection in linear regression. In *Proceedings of the 40th annual ACM symposium on theory of computing, STOC '08* (pp. 45–54).

Dayan, P., & Abbott, L. (2001). *Theoretical neuroscience*. MIT Press.

Djurisic, M., Popovic, M., Carnevale, N., & Zecevic, D. (2008). Functional structure of the mitral cell dendritic tuft in the rat olfactory bulb. *Journal of Neuroscience, 28*(15), 4057–4068.

Doucet, A., de Freitas, N., & Gordon, N. (Eds.) (2001). *Sequential Monte Carlo in practice*. Springer.

Durbin, J., & Koopman, S. (2001). *Time series analysis by state space methods*. Oxford: Oxford University Press.

Federov, V. V. (1972). *Theory of optimal experiments*. Orlando, FL: Academic.

Grewe, B. F., & Helmchen, F. (2009). Optical probing of neuronal ensemble activity. *Current Opinion in Neurobiology, 19*(5), 520–529. ISSN 0959-4388.

Grewe, B. F., Langer, D., Kasper, H., Kampa, B. M., & Helmchen, F. (2010). High-speed in vivo calcium imaging reveals neuronal network activity with near-millisecond precision. *Nature Methods, 7*(5), 399–405.

Hines, M. (1984). Efficient computation of branched nerve equations. *International Journal of Bio-Medical Computing, 15*(1), 69–76.

Homma, R., Baker, B. J., Jin, L., Garaschuk, O., Konnerth, A., Cohen, L. B., et al. (2009). Wide-field and two-photon imaging of brain activity with voltage- and calcium-sensitive dyes. In J. M. Walker, & F. Hyder (Eds.), *Dynamic brain imaging. Methods in molecular biology* (Vol. 489, pp 43–79). Humana Press. ISBN 978-1-59745-543-5.

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

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

Ishizuka, N., Cowan, W. M., & Amaral, D. G. (1995). A quantitative analysis of the dendritic organization of pyramidal cells in the rat hippocampus. *Journal of Comparative Neurology, 362*(1), 17–45.

Kellems, A., Roos, D., Xiao, N., & Cox, S. (2009). Low-dimensional, morphologically accurate models of subthreshold membrane potential. *Journal of Computational Neuroscience, 27*, 161–176.

Koch, C. (1999). *Biophysics of computation*. Oxford: Oxford University Press.

Krause, A. (2010). Sfo: A toolbox for submodular function optimization. *Journal of Machine Learning Research, 11*, 1141–1144.

Krause, A., & Guestrin, C. (2005). Near-optimal nonmyopic value of information in graphical models. In *Conference on uncertainty in artificial intelligence (UAI)*.

Krause, A., McMahan, B., Guestrin, C., & Gupta, A. (2007). *Selecting observations against adversarial objectives*. Technical report, In NIPS.

Krause, A., McMahan, B., Guestrin, C., & Gupta, A. (2008a). Robust submodular observation selection. *Journal of Machine Learning Research, 9*, 2761–2801.

Krause, A., Singh, A., & Guestrin, C. (2008b). Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research, 9*, 235–284.

Lewi, J., Butera, R., & Paninski, L. (2009). Sequential optimal design of neurophysiology experiments. *Neural Computation, 21*, 619–687.

Losavio, B. E., Liang, Y., Pang, A. S., Kakadiaris, I. A., Colbert, C. M., & Saggau, P. (2008). Live neuron morphology automatically reconstructed from multiphoton and confocal imaging data. *Journal of Neurophysiology, 100*(4), 2422–2429. doi:10.1152/jn.90627.2008. URL:http://jn.physiology.org/content/100/4/2422.abstract.

Morse, T., Davison, A., & Hines, M. (2001). *Parameter space reduction in neuron model optimization through minimization of residual voltage clamp current*. Society for Neuroscience Abstracts.

Nemhauser, G., Wolsey, L., & Fisher, J. (1978). An analysis of the approximations for maximizing submodular set functions. *Mathematical Programming, 14*, 265–294.

Nevian, T., Larkum, M., Polsky, A., & Schiller, J. (2007). Properties of basal dendrites of layer 5 pyramidal neurons:

A direct patch-clamp recording study. *Nature Neuroscience, 10*, 206–214.

Paninski, L. (2010). Fast Kalman filtering on quasilinear dendritic trees. *Journal of Computational Neuroscience, 28*, 211–228.

Paninski, L., Ahmadian, Y., Ferreira, D., Koyama, S., Rahnama, K., Vidne, M., et al. (2010). A new look at state-space models for neural data. *Journal of Computational Neuroscience, 29*, 107–126.

Paninski, L., Rad, K. R., & Huggins, J. (2011). *Fast low-snr Kalman filtering, with applications to high-dimensional smoothing* (under review).

Petrusca, D., Grivich, M. I., Sher, A., Field, G. D., Gauthier, J. L., Greschner, M., et al. (2007). Identification and characterization of a Y-like primate retinal ganglion cell type. *Journal of Neuroscience, 27*(41), 11019–11027.

Pnevmatikakis, E. A., Kelleher, K., Chen, R., Josic, K., Saggau, P., & Paninski, L. (2011). Fast nonnegative spatiotemporal calcium smoothing in dendritic trees. In *COSYNE*.

Press, W., Teukolsky, S., Vetterling, W., & Flannery, B. (1992). *Numerical recipes in C*. Cambridge: Cambridge University Press.

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE, 77*, 257–286.

Reddy, G. D., Kelleher, K., Fink, R., & Saggau, P. (2008). Three-dimensional random access multiphoton microscopy for functional imaging of neuronal activity. *Nature Neuroscience, 11*(6), 713–720.

Seeger, M. (2009). *On the submodularity of linear experimental design*. Unpublished Note.

Shumway, R., & Stoffer, D. (2006). *Time series analysis and its applications*. Springer.

Sjostrom, P. J., Rancz, E. A., Roth, A., & Hausser, M. (2008). Dendritic excitability and synaptic plasticity. *Physiological Reviews, 88*(2), 769–840.

Spruston, N. (2008). Pyramidal neurons: Dendritic structure and synaptic integration. *Nature Reviews Neuroscience, 9*, 206–221.

Stuart, G., & Sakmann, B. (1994). Active propagation of somatic action potential into neocortical pyramidal cell dendrites. *Nature, 367*, 69–72.

Stuart, G., Spruston, N., & Häusser, M. (Eds.) (1999). *Dendrites*. Oxford: Oxford University Press.

Vucinic, D., & Sejnowski, T. J. (2007). A compact multiphoton 3d imaging system for recording fast neuronal activity. *PLoS ONE, 2*(8), e699.

Wood, R., Gurney, K., & Wilson, C. (2004). A novel parameter optimisation technique for compartmental models applied to a model of a striatal medium spiny neuron. *Neurocomputing, 58–60*, 1109–1116.

Zador, A., & Pearlmutter, B. (1993). *Efficient computation of sparse elements of the inverse of a sparse near-tridiagonal matrix with application to the nerve equation*. Technical Report, Oregon Graduate Institute of Science and Technology, Department of Computer Science and Engineering.