

Fast Active Set Methods for Online Spike Inference from Calcium Imaging

Authored by:

Liam Paninski
Johannes Friedrich

Abstract

Fluorescent calcium indicators are a popular means for observing the spiking activity of large neuronal populations. Unfortunately, extracting the spike train of each neuron from raw fluorescence calcium imaging data is a nontrivial problem. We present a fast online active set method to solve this sparse nonnegative deconvolution problem. Importantly, the algorithm progresses through each time series sequentially from beginning to end, thus enabling real-time online spike inference during the imaging session. Our algorithm is a generalization of the pool adjacent violators algorithm (PAVA) for isotonic regression and inherits its linear-time computational complexity. We gain remarkable increases in processing speed: more than one order of magnitude compared to currently employed state of the art convex solvers relying on interior point methods. Our method can exploit warm starts; therefore optimizing model hyperparameters only requires a handful of passes through the data. The algorithm enables real-time simultaneous deconvolution of $\mathcal{O}(10^5)$ traces of whole-brain zebrafish imaging data on a laptop.

1 Paper Body

Calcium imaging has become one of the most widely used techniques for recording activity from neural populations in vivo [1]. The basic principle of calcium imaging is that neural action potentials (or spikes), the point process signal of interest, each induce an optically measurable transient response in calcium dynamics. The nontrivial problem to extract the spike train of each neuron from a raw fluorescence trace has been addressed with several different approaches, including template matching [2] and linear deconvolution [3, 4], which are outperformed by sparse nonnegative deconvolution [5]. The latter can be interpreted as the MAP estimate under a generative model (linear convolution plus noise; Fig. 1), whereas fully Bayesian methods [6, 7] can provide some further improvements, but are more computationally expensive. Supervised methods trained on simultaneously-recorded electrophysiological and imaging data [8, 9]

have also recently achieved state of the art results, but are more black-box in nature. The methods above are typically applied to imaging data offline, after the experiment is complete; however, there is a need for accurate and fast real-time processing to enable closed-loop experiments, a powerful strategy for causal investigation of neural circuitry [10]. In particular, observing and feeding back the effects of circuit interventions on physiologically relevant timescales will be valuable for directly testing whether inferred models of dynamics, connectivity, and causation are accurate in vivo, and recent experimental advances [11, 12] are now enabling work in this direction. Brain-computer interfaces (BCIs) also rely on real-time estimates of neural activity. Whereas most BCI systems rely on electrical recordings, BCIs have been driven by optical signals too [13], providing new insight into how neurons change their activity during learning on a finer spatial scale than possible with intracortical electrodes. Finally, adaptive experimental design approaches [14, 15, 16] also rely on online estimates of neural activity. 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

Fluorescence

s

c

y

2 1 0 0

150

Time

300

Figure 1: Generative autoregressive model for calcium dynamics. Spike train s gets filtered to produce calcium trace c ; here we used $p = 2$ as order of the AR process. Added noise yields the observed fluorescence y .

Even in cases where we do not require the strict timing/latency constraints of real-time processing, we still need methods that scale to large data sets as for example in whole-brain imaging of larval zebrafish [17, 18]. A further demand for scalability stems from the fact that the deconvolution problem is solved in the inner loop of constrained nonnegative matrix factorization (CNMF) [19], the current state of the art for simultaneous denoising, deconvolution, and demixing of spatiotemporal calcium imaging data. In this paper we address the pressing need for scalable online spike inference methods. We build on the success of framing spike inference as a sparse nonnegative deconvolution problem. Current algorithms employ interior point methods to solve the ensuing optimization problem and are fast enough to process hundreds of neurons in about the same time as the recording [5], but will not scale to currently obtained larger data sets such as whole-brain zebrafish imaging. Furthermore, these interior point methods scale linearly, but they cannot be warm started, i.e. be initialized with the solution from a previous iteration to gain speed-ups, and do not run online. We noted a close connection between the MAP problem and isotonic regression, which fits data by a monotone piecewise constant function. A classic isotonic regression algorithm is the pool adjacent violators algorithm (PAVA) [20, 21], which sweeps through the data looking for violations of the monotonicity con-

straint. When it finds one, it adjusts the estimate to the best possible fit with constraints, which amounts to pooling data points with the same fitted value. During the sweep adjacent pools that violate the constraints are merged. We generalized PAVA to derive an Online Active Set method to Infer Spikes (OASIS) that yields speed-ups in processing time by at least one order of magnitude compared to interior point methods on both simulated and real data. Further, OASIS can be warm-started, which is useful in the inner loop of CNMF, and also when adjusting model hyperparameters, as we show below. Importantly, OASIS is not only much faster, but operates in an online fashion, progressing through the fluorescence time series sequentially from beginning to end. The advances in speed paired with the inherently online fashion of the algorithm enable true real-time online spike inference during the imaging session, with the potential to significantly impact experimental paradigms. We expect our algorithm to be a useful tool for the neuroscience community, to enable new experiments that online access to spike timings affords and to be of interest in other fields, such as physics and quantitative finance, that deal with jump diffusion processes. The rest of this paper is organized as follows: Section 2 introduces the autoregressive model for calcium dynamics. In Section 3 we derive our active set method for the sparse nonnegative deconvolution problem for the simple case of AR(1) dynamics and generalize it to arbitrary AR(p) processes in the Supplementary Material. We further use the problem’s dual formulation to adjust the sparsity level in a principled way (following [19]), and describe methods for fitting model hyperparameters including the coefficients of the AR process. In Section 4 we show some results on simulated as well as real data. Finally, in Section 5 we conclude with possible further extensions.

2

Autoregressive model for calcium dynamics

We assume we observe the fluorescence signal for T timesteps, and denote by s_t the number of spikes that the neuron fired at the t -th timestep, $t = 1, \dots, T$, cf. Figure 1. We approximate the calcium concentration dynamics c using a stable autoregressive process of order p (AR(p)) where p is a small positive integer, usually $p = 1$ or 2 , $p \times c_t = \sum_{i=1}^p c_{t-i} + s_t$. (1)

The observed fluorescence $y \in \mathbb{R}^T$ is related to the calcium concentration as [5, 6, 7]: $y_t = a c_t + \epsilon_t$, $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$ (2)

where a is a nonnegative scalar and the noise is assumed to be i.i.d. zero mean Gaussian with variance σ^2 . For the remainder we assume units such that $a = 1$ without loss of generality. The parameters ϕ_i and σ can be estimated from the autocovariance function and the power spectral density (PSD) of y respectively [19]. The autocovariance approach assumes that the spiking signal s comes from a homogeneous Poisson process and in practice often gives a crude estimate of ϕ_i . We will improve on this below (Fig. 3) by fitting the AR coefficients directly, which leads to better estimates, particularly when the spikes have some significant autocorrelation. The goal of calcium deconvolution is to extract an estimate of the neural activity s from the vector of observations y . As discussed in [5, 19], this leads to the following nonnegative LASSO problem

for estimating the calcium concentration: minimize c

$$\begin{aligned} & \frac{1}{2} \|y - X\beta\|^2 + \lambda \|\beta\|_1 \\ & \text{subject to } s = G\beta \geq 0 \end{aligned} \quad (3)$$

where the ℓ_1 penalty enforces sparsity of the neural activity and the lower triangular matrix G is defined as: $G = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}$

$$\begin{aligned} & G = \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & & \ddots \end{bmatrix} \\ & \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & & \ddots \end{bmatrix} \end{aligned} \quad (4)$$

Following the approach in [5] the spike signal s is relaxed from nonnegative integers to arbitrary nonnegative values.

Derivation of the active set algorithm

The optimization problem (3) could be solved using generic convex program solvers. Here we derive the much faster Online Active Set method to Infer Spikes (OASIS).

Online Active Set method to Infer Spikes (OASIS)

For simplicity we consider first the AR(1) model and defer the cumbersome general case $p > 1$ to the Supplementary Material. We begin by inserting the definition of s (Eq. 3, skipping the index of i for a single AR coefficient). Using that s is constrained to be nonnegative yields for the sparsity penalty $\lambda \|\beta\|_1 = \lambda \sum_{i=1}^n \beta_i$

$$\begin{aligned} & \min_{\beta} \frac{1}{2} \|y - X\beta\|^2 + \lambda \sum_{i=1}^n \beta_i \\ & \text{subject to } \beta \geq 0 \end{aligned} \quad (5)$$

with $\beta_t := \beta(1 - \delta + \delta T)$ (with δ denoting Kronecker's delta) by noting that the sum of the last column of G is 1, whereas all other columns sum to 0.

$$\begin{aligned} & \min_{\beta} \frac{1}{2} \|y - X\beta\|^2 + \lambda \sum_{t=1}^n \beta_t \\ & \text{subject to } \beta_{t+1} \leq \beta_t \end{aligned} \quad (6)$$

shares some similarity to isotonic regression with the constraint $\beta_{t+1} \leq \beta_t$. However, our constraint $\beta_{t+1} \leq \beta_t$ bounds the rate of decay instead of enforcing monotonicity. We generalize PAVA to handle the additional factor δ . The algorithm is based on the following: For an optimal solution, if $y_t > y_{t+1}$, then the constraint becomes active and holds with equality, $\beta_t = \beta_{t+1}$. (Supposing the opposite, i.e. $y_t < y_{t+1}$, we could move β_{t+1} and β_t by some small ϵ to decrease the objective without violating the constraints, yielding a proof by contradiction.) We first present the algorithm in a way that

conveys its core ideas, then improve the algorithm's efficiency by introducing 'pools' of variables (adjacent c_t values) which are updated simultaneously. We introduce temporary values c_0 and initialize them to the unconstrained least squares solution, $c_0 = y$. Initially all constraints are in the 'passive set' and possible violations are fixed by subsequently adding the respective constraints to the 'active set'. Starting at $t = 2$ one moves forward until a violation of the constraint $c_0 - c_{t-1}$ at some time t is detected (Fig. 2A). Now the constraint is added to the active set and enforced by setting $c_0 = c_{t-1}$. Updating the two time steps by minimizing $\frac{1}{2}(y_t - c_0)^2 + \frac{1}{2}(y_{t-1} - c_0)^2 + \frac{\lambda}{2}(c_0 - c_{t-1})^2$ yields an updated value c_0 . However, this updated value can violate the constraint $c_0 - c_{t-2}$ and we need to update $c_0 - c_{t-2}$ as well, etc., until we have backtracked some t steps to time $t' = t - t$.

A move forward 7
 track back 3
 C move forward 3 D move forward 3 E move forward 7
 track back 7
 track back 3
 H move forward 3
 F
 G
 I
 track back 3
 ...

Figure 2: Illustration of OASIS for an AR(1) process (see Supplementary Video). Red lines depict true spike times. The shaded background shows how the time points are gathered in pools. The pool currently under consideration is indicated by the blue crosses. A constraint violation is encountered for the second time step (A) leading to backtracking and merging (B). The algorithm proceeds moving forward (C-E) until the next violation occurs (E) and triggers backtracking and merging (F-G) as long as constraints are violated. When the most recent spike time has been reached (G) the algorithm proceeds forward again (H). The process continues until the end of the series has been reached (I). The solution is obtained and pools span the inter-spike-intervals.

where the constraint $c_0 - c_{t-1}$ is already valid. At most one needs to backtrack to the most recent spike, because $c_0 - c_{t-1}$ at spike times t (Eq. 1). (Because such delays could be too long for some interesting closed loop experiments, we show in the Supplementary Material how well the method performs if backtracking is limited to just few frames.) Solving $\frac{1}{2}(y_t - c_0)^2 + \frac{1}{2}(y_{t-1} - c_0)^2 + \frac{\lambda}{2}(c_0 - c_{t-1})^2$ by setting the derivative to zero yields c_0

$$P_t = \frac{y_t + y_{t-1} + \lambda c_{t-1}}{2 + \lambda} \quad \text{for } t = 1, \dots, T. \quad (8)$$

and the next values are updated according (Along the way it is worth noting that, because a spike induces a calcium response described by kernel h with components h_i

(y_{t+1})

$y_{t+1} = \sum_{i=1}^t c_i h_{t+1-i}$ could be expressed in the more familiar regression form as $h_{t+1}^T v$, where $h_{t+1} = [h_{t+1}, \dots, h_1]^T$ we used the notation $v_{i:j}$ to describe a vector formed by components i to j of v .) Now one moves forward again (Fig. 2C-E) until detection of the next violation (Fig. 2E), backtracks again to the most recent spike (Fig. 2G), etc. Once the end of the time series is reached (Fig. 2I) we have found the optimal solution and set $c = c_0$.

In a worst case situation a constraint violation is encountered at every step of the forward sweep PT through the series. Updating all t values up to time t yields overall $\sum_{t=1}^T (T-t+1)$ updates and an $O(T^2)$ algorithm. In order to obtain a more efficient algorithm we introduce pools which are tuples of the form (v_i, w_i, t_i, l_i) with value v_i , weight w_i , event time t_i and pool length l_i . Initially there is a pool $(y_t, 1, t, 1)$ for each time step t . During backtracking pools get combined and only the first value $v_i = c_0 t_i$ is explicitly considered, while the other values are merely defined implicitly via $c_{t+1} = c_t$. The constraint $c_{t+1} \leq c_t$ translates to $v_{i+1} \leq l_i v_i$ as the criterion determining whether pools need to be combined. The introduced weights allow efficient value updates whenever pools are merged by avoiding recalculating the sums in equation (8). Values are updated according to $w_i v_i + \sum_{j=i+1}^l w_j v_{j+1} v_i \leq (w_i + \sum_{j=i+1}^l w_j) v_{i+1}$ where the denominator is the new weight of the pool and the pool lengths are summed $w_i \leq w_i + \sum_{j=i+1}^l w_j$ (10) $l_i \leq l_i + l_{i+1}$ (11)

4 Whenever pools i and $i+1$ are merged, former pool $i+1$ is removed and the succeeding pool indices decreased by 1. It is easy to prove by induction that the updates according to equations (9-11) guarantee that equation (8) holds for all values (see Supplementary Material) without having to explicitly calculate it. The latter would be expensive for long pools, whereas merging two pools has $O(1)$ complexity independent of the pool lengths. With pooling the considered worst case situation results in a single pool that is updated at every step forward, yielding $O(T)$ complexity. Analogous to PAVA, the updates solve equation (6) not just greedily but optimally. The final algorithm is summarized in Algorithm 1 and illustrated in Figure 2 as well as in the Supplementary Video. Algorithm 1 Fast online deconvolution algorithm for AR(1) processes with positive jumps

Require: data y , decay factor γ , regularization parameter λ

1: initialize pools as $P = \{(v_i, w_i, t_i, l_i)\}_{i=1}^T$ $\{(y_t, \gamma(1-\gamma) + \gamma t, 1)\}_{t=1}^T$

2: while $i \leq T$ do

3: while $i \leq |P|$ and $v_{i+1} \leq l_i v_i$ do

4: $i = i + 1$. move forward

5: if $i == |P|$ then break

6: while $i \leq l_i$ and $v_{i+1} \leq l_i v_i$ do . track back

$w = v + \sum_{j=i+1}^l w_j$

$v = v_i$

7: $i = i + 1$

8: $P_i = (v, w, t_i, l_i + l_{i+1})$

9: $i = i + 1$

10: for (v, w, t, l) in P do

11: for $\tau = 0, \dots, l$ do

12: $c_{t+\tau} = \max(0, v)$

13: return c

. Eqs. (9-11)

Dual formulation with hard noise constraint

subject to $s = Gc \geq 0$

(12)

(13)

2 z IX i ? i ? X 1 ? ? ? li (1 ? ? ?iz) t vi ? ?? ? ? yti +t = ? 2 T (14) w i i = 1
t=0 ? P P ?? + ? 2 ? 4? 2 and solving the quadratic equation yields ?? = with
? = i,t ? it , ? = 2 i,t ? it ? it 2? li P iz) t and = i,t ? 2it ? ? ? 2 T where ? it =
1?? w(1?? ? and ? it = yti +t ? vi ? t . i 5

run Alg. 1 2

?? 0

?? ?

run Alg. 1 RSS

D ? 2T E

?

?

??

?

run Alg. 1

Fluorescence

RSS

B ? 2T

Truth

Estimate

```

Data
Correlation: 0.734
0 2
Correlation: 0.753
0 2
Correlation: 0.767
0 2
Correlation: 0.777
0 2
Correlation: 0.791
0
.. .
Iterate B ? E F
?3 iterations 2 to converge 0
true ?
10
? from autocovariance
20
Correlation: 0.849
Time [s]
30
40

```

Figure 3: Optimizing sparsity parameter λ and AR coefficient α . (A) Running the active set method, with conservatively small estimate of λ , yields an initial denoised estimate (blue) of the data (yellow) roughly capturing the truth (red). We also report the correlation between the deconvolved estimate and true spike train as direct measure for the accuracy of spike train inference. (B) Updating sparsity parameter λ according to Eq. (14) such that $RSS = \lambda^2 T$ (left) shifts the current estimate downward (right, blue). (C) Running the active set method enforces the constraints again and is fast due to warm-starting. (D) Updating λ by minimizing the polynomial function $RSS(\lambda)$ and (E) running the warm-started active set method completes one iteration, which yields already a decent fit. (F) A few more iterations improve the solution further and the obtained estimate is hardly distinguishable from the one obtained with known true λ (turquoise dashed on top of blue solid line). Note that determining λ based on the autocovariance (purple) yields a crude solution that even misses spikes (at 24.6 s and 46.5 s).

The solution \hat{v}_i provides a good approximate proposal step for updating the pool values v_i (using Eq. 13). Since this update proposal is only approximate it can give rise to violated constraints (e.g., negative values of v_i). To satisfy all constraints Algorithm 1 is run to update the pool structure, cf. Figure 3C, but with a warm start: we initialize with the current set of merely z pools P_0 instead of the T pools for a cold start (Alg. 1, line 1). This step returns a set of v_i values that satisfy the constraints and may merge pools (i.e., delete spikes); then the procedure (update λ then rerun the warm-started Algorithm 1) can be iterated until no further pools need to be merged, at which point the

procedure has converged. In practice this leads to an increasing sequence of γ values (corresponding to an increasingly sparse set of spikes), and no pool-split (i.e., add-spike) moves are necessary¹. This warm-starting approach brings major speed benefits: after the residual is updated following a γ update, the computational cost of the algorithm is linear in the number of pools z , hence warm starting drastically reduces computational costs from $k_1 T$ to $k_2 z$ with proportionality constants k_1 and k_2 : if no pool boundary updates are needed then after warm starting the algorithm only needs to pass once through all pools to verify that no constraint is violated, whereas a cold start might involve a couple passes over the data to update pools, so k_2 is typically significantly smaller than k_1 , and z is typically much smaller than T (especially in sparsely-spiking regimes). 3.3

Optimizing the AR coefficient

Thus far the parameter γ has been known or been estimated based on the autocovariance function. We can improve upon this estimate by optimizing γ as well, which is illustrated in Figure 3. After updating γ followed by running Algorithm 1, we perform a coordinate descent step in γ that minimizes the RSS, cf. Figure 3D. The RSS as a function of γ is a high order polynomial, cf. equation (8), and we need to settle for numerical solutions. We used Brent's method [22] with bounds $0 \leq \gamma \leq 1$. One iteration consists now of steps B-E in Figure 3, while for known γ only B-C were necessary. 1

Note that it is possible to cheaply detect any violations of the KKT conditions in a candidate solution; if such a violation is detected, the corresponding pool could be split and the warm-started Algorithm 1 run locally near the detected violations. However, as we noted, due to the increasing γ sequence we did not find this step to be necessary in the examples examined here.

6
Activity
Fluor.
B
OASIS
2
CVXPY
Truth
D
Data
1
Time [s]
1.0 0.1
0 1
0.01
0.5
0 0
25
C
6

E 2
 0 1
 0 0
 Time [s]
 30
 0
 Time [s]
 O. E. M. S. G.
 0 OASIS ECOS MOSEK SCS GUROBI
 50
 Time [s]
 Activity
 Fluor.
 A
 30
 1 0.1
 1
 0.01 O. E. M. S. G.
 0 OASIS ECOS MOSEK SCS GUROBI Solver

Figure 4: OASIS produces the same high quality results as convex solvers at least an order of magnitude faster. (A) Raw and inferred traces for simulated AR(1) data, (B) simulated AR(2) and (C) real data from [29] modeled as AR(2) process. OASIS solves equation (3) exactly for AR(1) and just approximately for AR(2) processes, nevertheless well extracting spikes. (D) Computation time for simulated AR(1) data with given γ (blue circles, Eq. 3) or inference with hard noise constraint (green x, Eq. 12). GUROBI failed on the noise constrained problem. (E) Computation time for simulated AR(2) data.

4 4.1

Results Benchmarking OASIS

We generated datasets of 20 fluorescence traces each for $p = 1$ and 2 with a duration of 100 s at a framerate of 30 Hz, such that $T = 3,000$ frames. The spiking signal came from a homogeneous Poisson process. We used $\gamma = 0.95$, $\gamma = 0.3$ for the AR(1) model and $\gamma_1 = 1.7$, $\gamma_2 = 0.712$, $\gamma = 1$ for the AR(2) model. Figures 4A-C are reassuring that our suggested (dual) active set method yields indeed the same results as other convex solvers for an AR(1) process and that spikes are extracted well. For an AR(2) process OASIS is greedy and yields good results that are similar to the one obtained with convex solvers (lower panels in Fig. 4B and C), with virtually identical denoised fluorescence traces (upper panels). An exact fast (primal) active set method for this case is presented in the extended journal version of this paper [23]. Figures 4D,E report the computation time (γ SEM) averaged over all 20 traces and ten runs per trace on a MacBook Pro with Intel Core i5 2.7 GHz CPU. We compared the run time of our algorithm to a variety of state of the art convex solvers that can all be conveniently called from the convex optimization toolbox CVXPY [24]: embedded conic solver (ECOS, [25]), MOSEK [26], splitting conic solver (SCS, [27]) and GUROBI [28]. With given sparsity parameter γ (Eq. 3)

OASIS is about two magnitudes faster than any other method for an AR(1) process (Fig. 4D, blue disks) and more than one magnitude for an AR(2) process (Fig. 4E). Whereas the other solvers take almost the same time for the noise constrained problem (Eq. 12, Fig. 4D,E, green x), our method takes about three times longer to find the value of the dual variable λ compared to the formulation where the residual is part of the objective; nevertheless it still outperforms the other algorithms by a huge margin. We also ran the algorithms on longer traces of length $T = 30,000$ frames, confirming that OASIS scales linearly with T . Our active set method maintained its lead by 1-2 orders of magnitude in computing time. Further, compared to our active set method the other algorithms required at least an order of magnitude more RAM, confirming that OASIS is not only faster but much more memory efficient. Indeed, because OASIS can run in online mode the memory footprint can be $O(1)$, instead of $O(T)$. We verified these results on real data as well. Running OASIS with the hard noise constraint and $p = 2$ on the GCaMP6s dataset collected at 60 Hz from [29] took 0.101 ± 0.005 s per trace, whereas the fastest other methods required 2.37 ± 0.12 s. Figure 4C shows the real data together with the inferred denoised and deconvolved traces as well as the true spike times, which were obtained by simultaneous electrophysiological recordings [29]. We also extracted each neuron's fluorescence activity using CNMF from an unpublished whole-brain zebrafish imaging dataset from the M. Ahrens lab. Running OASIS with hard noise constraint and

$p = 1$ (chosen because the calcium onset was fast compared to the acquisition rate of 2 Hz) on 10,000 traces out of a total of 91,478 suspected neurons took 81.5 s whereas ECOS, the fastest competitor, needed 2,818.1 s. For all neurons we would hence expect 745 s for OASIS, which is below the 1,500 s recording duration, and over 25,780 s for ECOS and other candidates. 4.2

Hyperparameter optimization

We have shown that we can solve equation (3) and equation (12) faster than interior point methods. The AR coefficient λ was either known or estimated based on the autocorrelation in the above analyses. The latter approach assumes that the spiking signal comes from a homogeneous Poisson process, which does not generally hold for realistic data. Therefore we were interested in optimizing not only the sparsity parameter λ , but also the AR(1) coefficient λ . To illustrate the optimization of both, we generated a fluorescence trace with spiking signal from an inhomogeneous Poisson process with sinusoidal instantaneous firing rate (Fig. 3), thus mimicking realistic data. We conservatively initialized λ to a small value of 0.9. The value obtained based on the autocorrelation was 0.9792 and larger than the true value of 0.95. The left panels in Figures 3B and D illustrate the update of λ from the previous value λ_{old} to λ_{new} by solving a quadratic equation analytically (Eq. 14) and the update of λ by numerical minimization of a high order polynomial respectively. Note that after merely one iteration (Fig. 3E) a good solution is obtained and after three iterations the solution is virtually identical to the one obtained when the true value of λ has been provided (Fig. 3F). This holds not only visually, but also when judged by the correlation between deconvolved activity and ground truth spike

train, which was 0.869 compared to merely 0.773 if γ was obtained based on the autocorrelation. The optimization was robust to the initial value of γ , as long as it was positive and not, or only marginally, greater than the true value. The value obtained based on the autocorrelation was considerably greater and partitioned the time series into pools in a way that missed entire spikes. A quantification of the computing time for hyperparameter optimization as well as means to reduce it are presented in the extended journal version [23].

5

Conclusions

We presented an online active set method for spike inference from calcium imaging data. We assumed that the forward model to generate a fluorescence trace from a spike train is linear-Gaussian. Future work will extend the method to nonlinear models [30] incorporating saturation effects and a noise variance that increases with the mean fluorescence to better resemble the Poissonian statistics of photon counts. In the Supplementary Material we already extend our mathematical formulation to include weights for each time point as a first step in this direction. Further development, contained in the extended journal version [23], includes and optimizes an explicit fluorescence baseline. It also provides means to speed up the optimization of model hyperparameters, including the added baseline. It presents an exact and fast (primal) active set method for AR($p \leq 1$) processes and more general calcium response kernels. A further extension is to add the constraint that positive spikes need to be larger than some minimal value, which renders the problem non-convex. A minor modification to our algorithm enables it to find an (approximate) solution of this non-convex problem, which can be marginally better than the solution obtained with ℓ_1 regularization. Acknowledgments We would like to thank Misha Ahrens and Yu Mu for providing whole-brain imaging data of larval zebrafish. We thank John Cunningham for fruitful discussions and Scott Linderman as well as Daniel Soudry for valuable comments on the manuscript. Funding for this research was provided by Swiss National Science Foundation Research Award P300P2_158428, Simons Foundation Global Brain Research Awards 325171 and 365002, ARO MURI W911NF-12-1-0594, NIH BRAIN Initiative R01 EB22913 and R21 EY027592, DARPA N66001-15-C-4032 (SIMPLEX), and a Google Faculty Research award; in addition, this work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/ Interior Business Center (DoI/IBC) contract number D16PC00003. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/IBC, or the U.S. Government. 8

2 References

- [1] C Grienberger and C Konnerth. Imaging calcium in neurons. *Neuron*, 73(5):862–885, 2012. [2] B F Grewe, D Langer, H Kasper, B M Kampa, and F Helmchen. High-speed in vivo calcium imaging reveals neuronal network activity with near-millisecond precision. *Nat Methods*, 7(5):399–405, 2010. [3] E Yaksi and R W Friedrich. Reconstruction of firing rate changes across neuronal populations by temporally deconvolved Ca^{2+} imaging. *Nat Methods*, 3(5):377–383, 2006. [4] T F Holekamp, D Turaga, and T E Holy. Fast three-dimensional fluorescence imaging of activity in neural populations by objective-coupled planar illumination microscopy. *Neuron*, 57(5):661–672, 2008. [5] J T Vogelstein et al. Fast nonnegative deconvolution for spike train inference from population calcium imaging. *J Neurophysiol*, 104(6):3691–3704, 2010. [6] J T Vogelstein, B O Watson, A M Packer, R Yuste, B Jedynek, and L Paninski. Spike inference from calcium imaging using sequential monte carlo methods. *Biophys J*, 97(2):636–655, 2009. [7] E A Pnevmatikakis, J Merel, A Pakman, and L Paninski. Bayesian spike inference from calcium imaging data. *Asilomar Conference on Signals, Systems and Computers*, 2013. [8] T Sasaki, N Takahashi, N Matsuki, and Y Ikegaya. Fast and accurate detection of action potentials from somatic calcium fluctuations. *J Neurophysiol*, 100(3):1668–1676, 2008. [9] L Theis et al. Benchmarking spike rate inference in population calcium imaging. *Neuron*, 90(3):471–482, 2016. [10] L Groseknick, J H Marshel, and K Deisseroth. Closed-loop and activity-guided optogenetic control. *Neuron*, 86(1):106–139, 2015. [11] J P Rickgauer, K Deisseroth, and D W Tank. Simultaneous cellular-resolution optical perturbation and imaging of place cell firing fields. *Nat Neurosci*, 17(12):1816–1824, 2014. [12] A M Packer, L E Russell, H WP Dagleish, and M H?usser. Simultaneous all-optical manipulation and recording of neural circuit activity with cellular resolution in vivo. *Nat Methods*, 12(2):140–146, 2015. [13] K B Clancy, A C Koralek, R M Costa, D E Feldman, and J M Carmenta. Volitional modulation of optically recorded calcium signals during neuroprosthetic learning. *Nat Neurosci*, 17(6):807–809, 2014. [14] J Lewi, R Butera, and L Paninski. Sequential optimal design of neurophysiology experiments. *Neural Comput*, 21(3):619–687, 2009. [15] M Park and J W Pillow. Bayesian active learning with localized priors for fast receptive field characterization. In *Adv Neural Inf Process Syst*, pages 2348–2356, 2012. [16] B Shababo, B Paige, A Pakman, and L Paninski. Bayesian inference and online experimental design for mapping neural microcircuits. In *Adv Neural Inf Process Syst*, pages 1304–1312, 2013. [17] M B Ahrens, M B Orger, D N Robson, J M Li, and P J Keller. Whole-brain functional imaging at cellular resolution using light-sheet microscopy. *Nat Methods*, 10(5):413–420, 2013. [18] N Vladimirov et al. Light-sheet functional imaging in fictively behaving zebrafish. *Nat Methods*, 2014. [19] E A Pnevmatikakis et al. Simultaneous denoising, deconvolution, and demixing of calcium imaging data. *Neuron*, 89(2):285–299, 2016. [20] M Ayer et al. An empirical distribution function for sampling with incomplete information. *Ann Math Stat*, 26(4):641–647, 1955. [21] R E Barlow, D J Bartholomew, JM Bremner, and H D Brunk. Statistical inference under

order restrictions: The theory and application of isotonic regression. Wiley New York, 1972. [22] R P Brent. Algorithms for Minimization Without Derivatives. Courier Corporation, 1973. [23] J Friedrich, P Zhou, and L Paninski. Fast active set methods for online deconvolution of calcium imaging data. arXiv, 1609.00639, 2016. [24] S Diamond and S Boyd. CVXPY: A Python-embedded modeling language for convex optimization. J Mach Learn Res, 17(83):1?5, 2016. [25] A Domahidi, E Chu, and S Boyd. ECOS: An SOCP solver for embedded systems. In European Control Conference (ECC), pages 3071?3076, 2013. [26] E D Andersen and K D Andersen. The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In High performance optimization, pages 197?232. Springer, 2000. [27] B O?Donoghue, E Chu, N Parikh, and S Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. J Optim Theory Appl, pages 1?27, 2016. [28] Gurobi Optimization Inc. Gurobi optimizer reference manual, 2015. [29] T-W Chen et al. Ultrasensitive fluorescent proteins for imaging neuronal activity. Nature, 499(7458):295? 300, 2013. [30] T A Pologruto, R Yasuda, and K Svoboda. Monitoring neural activity and $[Ca^{2+}]$ with genetically encoded Ca^{2+} indicators. J Neurosci, 24(43):9572?9579, 2004.