

## Contents

Supplementary information.....	1
Model and parameter estimation.....	1
Relationship to spike removal by subtraction of the STA.....	5
Hyperparameter estimation .....	7
Empirical estimate of $g$ .....	9
Choice of basis.....	10
Chunking .....	10
Relationship to LFP-spike correlation removal .....	12
Appendix - Properties of circulant matrices .....	14
List of supplementary figures .....	16
Supplementary Figure 1: Summary of results for all spike-amplitude simulations.....	16
Supplementary Figure 2: Summary of results for all firing-rate simulations.....	16
Supplementary Figure 3: Effect of spike removal in an example recording.....	16
Supplementary Figure 4: Necessity of smoothness assumption in a simulation .....	16
Supplementary Figure 5: Choice of $\mathbf{g}$ .....	16
Supplementary Figure 6: Chunking procedure .....	16
Supplementary References.....	17

## Supplementary information

This section provides the mathematical details of our spike removal algorithm, its software implementation, and a comparison with other algorithms. All Matlab code is available as online supplementary material and from our website (<http://apps.mni.mcgill.ca/research/cpack/lfpcode.zip>).

### Model and parameter estimation

The model is as specified in the main Methods section; we repeat it here for easy reference, with slight changes in notation. Our goal is to estimate the local field potential (LFP) based on a measured wideband voltage trace of length  $n$ . We assume that this wideband signal  $\mathbf{y}$  is the superposition of a low-frequency local field potential  $\mathbf{w}$ , high-frequency spike components  $\boldsymbol{\eta}^k$ , an offset  $\mu$  and white noise  $\epsilon$  (Figure 1B):

$$\mathbf{y} = \mathbf{w} + \sum_{k=1}^m \boldsymbol{\eta}^k + \boldsymbol{\mu} + \boldsymbol{\epsilon} \quad (1)$$

Here  $m$  is the number of sorted neurons emitting spikes. The high-frequency component of the  $k^{\text{th}}$  neuron,  $\boldsymbol{\eta}^k$ , is created by the convolution of the neuron's spike train  $\mathbf{s}^k$ , assumed known (see Figure 1B), and the neuron's spike waveform  $\mathbf{B}\boldsymbol{\phi}^k$ :

$$\boldsymbol{\eta}^k = C'(\mathbf{s}^k)\mathbf{B}\boldsymbol{\phi}^k \quad (2)$$

Here the  $C'(\mathbf{a})$  returns a circulant matrix whose first row is  $\mathbf{a}$ . The product  $C'(\mathbf{a})\mathbf{b}$  returns the circular convolution of  $\mathbf{a}$  and  $\mathbf{b}$ :

$$C'(\mathbf{a})\mathbf{b} = (a \circledast b)_i = \sum_j a_{[i-j]} b_j \quad (3)$$

Here  $a_{[k]} = a_{(k \bmod n)+1}$ . Circulant matrices have a number of properties that are crucial for the tractability of the model parameters; their properties are covered in detail in the last section. We call  $\boldsymbol{\eta}^k$  the *waveform train* of the  $k^{\text{th}}$  neuron.

$\mathbf{B}$  is a matrix of basis functions which map the spike parameters  $\boldsymbol{\phi}^k$  onto a spike waveform. Typically, the number of parameters that describe the spike waveforms is much smaller than the length of the signal, which implies that  $\mathbf{B}$  is much taller than it is wide.

Assumptions are as follows:

$$\begin{aligned} p(\mathbf{w}) &= N(0, \gamma^2 \boldsymbol{\Gamma}) \\ p(\boldsymbol{\epsilon}) &= N(0, \sigma^2 \mathbf{I}) \end{aligned} \quad (4)$$

Here  $N(\mathbf{a}, \boldsymbol{\Sigma})$  represents a multivariate Gaussian with mean  $\mathbf{a}$  and covariance  $\boldsymbol{\Sigma}$ .  $\boldsymbol{\Gamma} = C'(F^{-1}(\mathbf{g}))$  is a matrix that embodies an assumption of smoothness, and  $\boldsymbol{\Gamma}\mathbf{x}$  produces a low-pass filtered version of  $\mathbf{x}$ .

By Bayes' theorem, we have that the posterior probability of the parameters is, up to an additive constant:

$$\begin{aligned} -\log p(\mathbf{w}, \boldsymbol{\phi}^k, \boldsymbol{\mu} | \mathbf{y}) &= \frac{1}{2\sigma^2} \left( \mathbf{y} - \mathbf{w} - \sum_k C'(\mathbf{s}^k)\mathbf{B}\boldsymbol{\phi}^k - \boldsymbol{\mu} \right)' \left( \mathbf{y} - \mathbf{w} - \sum_k C'(\mathbf{s}^k)\mathbf{B}\boldsymbol{\phi}^k - \boldsymbol{\mu} \right) \\ &\quad + \frac{1}{2\gamma^2} \mathbf{w}' \boldsymbol{\Gamma}^{-1} \mathbf{w} \end{aligned} \quad (5)$$

We can solve for the MAP estimate of the parameters by taking partial derivatives of the posterior and setting derivatives to zero. The MAP estimates are given by:

$$\begin{aligned}
\bar{\mathbf{w}} &= (\gamma^2 \mathbf{\Gamma} + \sigma^2 \mathbf{I})^{-1} \gamma^2 \mathbf{\Gamma} \left( \mathbf{y} - \sum_k C'(\mathbf{s}^k) \mathbf{B} \bar{\boldsymbol{\varphi}}^k - \bar{\mu} \right) \\
\bar{\boldsymbol{\varphi}}^k &= (C'(\mathbf{s}^k) \mathbf{B})^+ \left( \mathbf{y} - \bar{\mathbf{w}} - \sum_{j \neq k} C'(\mathbf{s}^j) \mathbf{B} \bar{\boldsymbol{\varphi}}^j - \bar{\mu} \right) \\
\bar{\mu} &= \frac{1}{n} \hat{\mathbf{1}}' \left( \mathbf{y} - \bar{\mathbf{w}} - \sum_k C'(\mathbf{s}^k) \mathbf{B} \bar{\boldsymbol{\varphi}}^k \right)
\end{aligned} \tag{6}$$

Here  $\mathbf{A}^+$  is the pseudoinverse  $\mathbf{A}^+ = (\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'$ , and  $\hat{\mathbf{1}}$  is a vector of ones. It is possible to solve this system by taking initial guesses of the model parameters, recomputing the parameters using these equations in sequence and iterating. Convergence of this iterative solution is however too slow for this to be a practical solution because of the strong couplings<sup>1</sup> between the LFP, the waveforms and the offset.

Instead we isolate each parameter set individually, starting with the waveforms  $\bar{\boldsymbol{\varphi}}^k$ . To simplify the equations, we isolate  $\bar{\boldsymbol{\varphi}}^k$  for a fixed value of  $k$  on the assumption that the other waveforms  $\bar{\boldsymbol{\varphi}}^{j \neq k}$  are known in advance. We will explicitly compensate for the errors induced by this simplification later. We thus define  $\mathbf{v} = \mathbf{y} - \sum_{j \neq k} C'(\mathbf{s}^j) \mathbf{B} \bar{\boldsymbol{\varphi}}^j$  and drop the indices in the equations to lighten the notation:

$$\begin{aligned}
\bar{\mathbf{w}} &= \mathbf{M}(\mathbf{v} - \mathbf{D} \bar{\boldsymbol{\varphi}} - \bar{\mu}) \\
\bar{\boldsymbol{\varphi}} &= \mathbf{D}^+(\mathbf{v} - \bar{\mathbf{w}} - \bar{\mu}) \\
\bar{\mu} &= \frac{1}{n} \hat{\mathbf{1}}'(\mathbf{v} - \bar{\mathbf{w}} - \mathbf{D} \bar{\boldsymbol{\varphi}})
\end{aligned} \tag{7}$$

Here  $\mathbf{D} \equiv C'(\mathbf{s})\mathbf{B}$  and  $\mathbf{M} = (\gamma^2 \mathbf{\Gamma} + \sigma^2 \mathbf{I})^{-1} \gamma^2 \mathbf{\Gamma}$ , a low-pass filter.  $\mathbf{D}'\mathbf{x}$  computes the spike-triggered sum of the signal  $\mathbf{x}$  while  $\mathbf{D}\mathbf{x}$  computes the waveform train associated with the spike waveform  $\mathbf{x}$ . By substituting  $\bar{\mu}$  into the first two equations, then isolating  $\bar{\mathbf{w}}$  in the first equation and finally substituting it in the second equation, we obtain the normal equation:

$$\mathbf{D}'\mathbf{J}(\mathbf{I} - \mathbf{M})\mathbf{D}\bar{\boldsymbol{\varphi}} = \mathbf{D}'\mathbf{J}(\mathbf{I} - \mathbf{M})\mathbf{v} \tag{8}$$

Here  $\mathbf{J} = \mathbf{I} - \frac{1}{n} \hat{\mathbf{1}} \hat{\mathbf{1}}'$  is a centering matrix which when applied to a vector yields the same vector but with its mean set to 0. This equation can be interpreted as follows: the STA ( $\mathbf{D}'$ ) of the centered ( $\mathbf{J}$ ) high-pass

---

<sup>1</sup> By strong coupling between parameters we mean that the terms of the Hessian of the log-posterior involving parameters of different types (LFP/waveforms/offset) are not negligible compared to terms of the Hessian involving parameters of similar type.

filtered  $(\mathbf{I} - \mathbf{M})$  waveform train  $(\mathbf{D}\bar{\boldsymbol{\varphi}})$  is equal to the STA of the centered high-pass filtered wideband signal.

While this equation is satisfyingly compact and intuitive, multiplication by  $\mathbf{M}$  must be done through multiplication in the Fourier domain, and hence a Fourier and an inverse Fourier transform need to be performed for every column of  $\mathbf{D}$ ; this operation is thus a bottleneck. Since spikes are assumed to be of finite duration, however, it follows that we can write  $\mathbf{B} = [\mathbf{B}_s; \mathbf{0}]$ , where  $\mathbf{0}$  is a matrix of zeros of size  $(n - q)$  by  $p$ ,  $q \ll n$  is the length of a spike and  $p \leq q$  is the number of basis functions. Note that this requires time shifting the spike train  $\mathbf{s}$  so that an entry equal to 1 indicates the beginning of a spike rather than its peak. Substituting and simplifying (8), we find:

$$\mathbf{D}'\mathbf{J}(\mathbf{I} - \mathbf{M})\mathbf{D} = \mathbf{B}_s'(\mathbf{k} * \mathbf{B}_s) - \frac{1}{n}(\mathbf{s}'\hat{\mathbf{1}})^2 \left( F^{-1} \left( \frac{\sigma^2}{\sigma^2 + \gamma^2 \mathbf{g}} \right)' \hat{\mathbf{1}} \right) (\mathbf{B}_s' \hat{\mathbf{1}})(\mathbf{B}_s' \hat{\mathbf{1}})' \quad (9)$$

Here  $*$  denotes linear convolution with zero-padding, applied column-wise, and  $\mathbf{k}$ , the convolution kernel, is derived from  $\mathbf{a} = F^{-1} \left( \frac{\sigma^2}{\sigma^2 + \gamma^2 \mathbf{g}} |F(\mathbf{s})|^2 \right)$  by circular time shifting, with  $k_i = a_{[i-q]}$ . Since  $q$  is small, the convolution in (9) is inexpensive and the equation can be solved by preconditioned conjugate gradients at negligible cost. After solving equation (9) sequentially for every neuron, we plug these estimates in the system of equations (6) and solve to find:

$$\bar{\mu} = \frac{1}{n} \hat{\mathbf{1}}' \left( \mathbf{y} - \sum_k C'(\mathbf{s}^k) \mathbf{B} \bar{\boldsymbol{\varphi}}^k \right) \quad (10)$$

Thus  $\bar{\mu}$  is equal to the mean of the wideband signal minus the mean of the spike contribution. Finally, the LFP  $\bar{\mathbf{w}}$  is given by the first equation in the system (6); it is the low-pass filtered wideband signal minus the spike contribution. In practice, the experimenter will probably want to use his or her own filter or filterbank on the despiked wideband signal to obtain the LFP. Hence  $\bar{\mathbf{w}}$  is never actually computed by the despiking algorithm; instead the algorithm works with and returns the despiked wideband signal, defined as:

$$\bar{\mathbf{z}} = \mathbf{y} - \sum_k C'(\mathbf{s}^k) \mathbf{B} \bar{\boldsymbol{\varphi}}^k - \bar{\mu} \quad (11)$$

Because we ignored cross terms when solving for the spike waveforms  $\bar{\boldsymbol{\varphi}}^k$ , and also because of possible numerical instability, solutions must be checked for convergence. It can be verified using equation (6) that for every  $k$ , the following auxiliary equation holds:

$$\mathbf{B}_s[\mathbf{B}_s; \mathbf{0}]' C(\mathbf{s}^k) \mathbf{J}(\mathbf{I} - \mathbf{M}) \left( \mathbf{y} - \sum_j C'(\mathbf{s}^j) \mathbf{B} \bar{\boldsymbol{\varphi}}^j - \mu \right) = \mathbf{0} \quad (12)$$

This states that the STA of a high-pass filtered, centered despiked wideband signal projected onto the basis in which we express spike waveforms is 0. Convergence is attained when the largest deviation from 0 observed is smaller than some fraction of the standard deviation of  $\mathbf{y}$ . When convergence is not attained, the process is repeated. In practice, it rarely takes more than 2 iterations to reach convergence, even when there are several neurons emitting spikes.

Supplementary Figure S3A shows an example of a wideband signal before and after spike removal. The two signals have been offset vertically to facilitate comparison. By construction, the proposed method removes only the mean spike waveform around the time of every spike, and hence does not remove all traces of spikes when the spike waveform is variable. It also relies on the information given by the experimenter about the timing of spikes, and does not remove spikes which have not been detected. These limitations can be overcome in large part with good spike detection, alignment, and sorting. In these cases the method performs admirably, as seen in this figure, and results in an appreciable change in the wideband signal. The changes are much less conspicuous when looking at the LFP, shown in Figure S3B. The difference between the two signals consists of a barely visible artifact around the time of every spike. Although the artifact is very small for each spike, it is highly stereotyped: it always occurs at the same time relative to spikes, and always has the same shape, sign and relative phase. Thus, any technique that looks at temporal relationships between spikes and LFPs will amplify the artifact, masking legitimate spike-LFP relationships, as we demonstrate in the main text.

### Relationship to spike removal by subtraction of the STA

The algorithm is conceptually similar to previously proposed methods of spike removal, in particular methods that subtract a mean spike determined by spike-triggered averaging the signal at the time of each spike (Pesaran et al. 2002; Zanos et al. 2006). Here we make this connection more explicit and highlight the differences between our proposed method and spike removal by STA subtraction.

Consider the case where the smallest interspike interval observed is equal to or longer than the duration of a spike, and the basis in which the waveform is expressed is given by  $\mathbf{B} = [\mathbf{I}_q; \mathbf{0}]$ ; here  $\mathbf{I}_q$  is the identity matrix of size  $q$  by  $q$  where  $q$  is the length of a spike. Furthermore, assume that no information is known about the LFP, such that  $\gamma^2 \rightarrow 0$ . Finally, assume that there is only one neuron spiking. Under these simplifying assumptions, the left hand side of equation (9) becomes:

$$\begin{aligned} \mathbf{D}'\mathbf{J}(\mathbf{I} - \mathbf{M})\mathbf{D}\bar{\boldsymbol{\varphi}} &= \mathbf{B}_s'(\mathbf{k} * \mathbf{B}_s\bar{\boldsymbol{\varphi}}) - \frac{1}{n}(\mathbf{s}'\hat{\mathbf{1}})^2 \left( F^{-1} \left( \frac{\sigma^2}{\sigma^2 + \gamma^2 \mathbf{g}} \right)' \hat{\mathbf{1}} \right) (\mathbf{B}_s'\hat{\mathbf{1}})(\mathbf{B}_s'\hat{\mathbf{1}})' \bar{\boldsymbol{\varphi}} \\ &= r^2 \left( \mathbf{I} - \frac{1}{n} \hat{\mathbf{1}}_q' \right) \bar{\boldsymbol{\varphi}} \end{aligned} \quad (13)$$

Here  $r$  is the total number of spikes. The right-hand side becomes:

$$\begin{aligned} \mathbf{D}'\mathbf{J}(\mathbf{I} - \mathbf{M})\mathbf{y} &= r[\mathbf{I}_q, \mathbf{0}] \left( \mathbf{I} - \frac{1}{n} \hat{\mathbf{1}}' \right) \mathbf{y} \\ &= r^2 \text{STA}(\mathbf{y} - \text{mean}(\mathbf{y})) \end{aligned} \quad (14)$$

Solving for  $\bar{\boldsymbol{\varphi}}$  by inverting  $\left(\mathbf{I} - \frac{1}{n} \hat{\mathbf{1}}_q'\right)$  using the Woodbury matrix identity, we find:

$$\begin{aligned}\bar{\boldsymbol{\varphi}} &= \text{STA}(\mathbf{y} - \text{mean}(\mathbf{y})) + \frac{rq}{n - rq} \text{mean}\left(\text{STA}(\mathbf{y} - \text{mean}(\mathbf{y}))\right) \\ &= \text{STA}(\mathbf{y} - \text{mean}(\mathbf{y})) + c\end{aligned}\quad (15)$$

Under these assumptions, the spike waveform  $\bar{\boldsymbol{\varphi}}$  is equal to the spike-triggered average of the centered wideband signal plus an offset  $c$ . Hence our proposed algorithm reduces to earlier proposals of removing spikes by subtracting the mean spike around the time of each spike in the limit of non-overlapping spikes and no assumptions on the smoothness of the LFP, save for the appearance of an offset  $c$ .

The appearance of this offset is puzzling, especially its sign which is the opposite of what intuition might dictate. This can be explained by the fact that the mean of the wideband signal will rise after spike removal, assuming spikes are negative. But the STA around the time of the spike will be zero after spike removal, which is inconsistent with this positive mean, and this will cause an artifactual downward dip in the STA around the time of a spike. The offset corrects this artifact.

Now that the relationship between our method and straightforward spike removal is clear, we can ask to what extent the unique features of our method are useful. The most important unique feature of our method is the assumption that the LFP is smooth. This assumption is crucial to obtain a correct estimate of the STA of the despiked signal when there is a real relationship between spikes and the LFP. This is illustrated in Figure S4 where we have created a surrogate signal composed of spikes and an LFP composed of low-frequency noise and a Gabor of smaller magnitude and longer time scale than a spike at every spike. Thus, there exists a true relationship between the spikes and the LFP. The STA of the despiked wideband signal when no assumptions are made has a prominent artifact around the time of a spike: it is flat and equal to 0. When an assumption of smoothness is made, however, the method correctly interpolates the signal around the time of the spike (Figure S4A). Figure S4B shows that the STA of the LFP obtained by smoothing the despiked signal without assumptions has an artifactual upward deflection around the time of the spike, while the STA under a smoothness assumption is correctly estimated. The proposed method accomplishes this better behavior around spikes by adjusting the spike waveform, as shown in Figure S4C. The difference between the two waveforms, illustrated in Figure S4D, appears to consist of a DC, linear, and quadratic trend. This shows that our method adjusts spikes waveforms by attenuating low-frequency trends which it attributes to the LFP.

It is important to understand that while the proposed method does noticeably better than previously proposed methods, it has its limitations. Our method works best when spikes are assumed to be short. When longer spike times are assumed (say, larger than 4 ms), the method works less well because in essence it must interpolate over a gap of several dozens of samples to determine what is part of the LFP and what is part of the spike. Paradoxically, therefore, the spike removal algorithm will be less aggressive in its spike removal when spikes are assumed to be too large. We therefore recommend

assuming spike lengths that are 3 ms or less; see the Choice of Basis section for another discussion of the choice of spike length.

## Hyperparameter estimation

Recall that our model assumptions are that:

$$\begin{aligned} p(\mathbf{w}) &= N(0, \gamma^2 \mathbf{\Gamma}); \mathbf{\Gamma} = C'(F^{-1}(\mathbf{g})) \\ p(\boldsymbol{\epsilon}) &= N(0, \sigma^2 \mathbf{I}) \end{aligned} \quad (16)$$

Up to now, we have assumed that  $\sigma$  and  $\gamma$  are known. The strength of the prior relative to the noise is important as it determines what the model considers as signal and what it discounts as noise. These hyperparameters, which control the regularization of the model, cannot be determined by MAP, unlike regular parameters (Wu et al. 2006). Here we determine these parameters by optimizing the marginal likelihood of the model, a metric which takes into account both the quality of the model fit to the data and the number of degrees of freedom in the model to determine the optimal degree of regularization. This method is also known as evidence optimization (Bishop 2007; see chapter 3 for a detailed introduction to this subject). We begin by ignoring the uncertainty in the model due to the parameters of the spike.  $\mathbf{z}$  is defined as before as the despiked wideband signal  $\mathbf{z} = \mathbf{y} - \sum_k C'(\mathbf{s}^k) \mathbf{B} \bar{\boldsymbol{\varphi}}^k - \bar{\boldsymbol{\mu}}$ .  $\bar{\boldsymbol{\varphi}}^k$  and  $\bar{\boldsymbol{\mu}}$  are assumed to have both been estimated according to the methods of the Model and Parameter Estimation section. The marginal likelihood of the model is defined as:

$$p(\mathbf{z}|\sigma, \gamma, \mathbf{g}) = \int p(\mathbf{z}|\mathbf{w}, \sigma) p(\mathbf{w}|\gamma, \mathbf{g}) d\mathbf{w} \quad (17)$$

The marginal likelihood is thus the likelihood of the data (and therefore the model) with the uncertainty in the model parameters  $\mathbf{w}$  marginalized out by integration. Unlike in MAP estimation, normalization constants that ensure that probabilities integrate to 1 are of crucial importance and thus are not ignored. The marginal likelihood is then:

$$p(\mathbf{z}|\sigma, \gamma, \mathbf{g}) = \frac{1}{\sqrt{2\pi|\sigma^2|}} \frac{1}{\sqrt{2\pi|\gamma^2 \mathbf{\Gamma}|}} \int \exp\left(-\frac{1}{2\sigma^2} (\mathbf{z} - \mathbf{w})'(\mathbf{z} - \mathbf{w}) - \frac{1}{2\gamma^2} \mathbf{w}' \mathbf{\Gamma}^{-1} \mathbf{w}\right) d\mathbf{w} \quad (18)$$

Here  $|\mathbf{M}|$  is the determinant of the matrix  $\mathbf{M}$ . The integral is performed by completing the square inside the exponential. Taking the negative log of the integral and grouping terms which are independent of the hyperparameters into a constant term  $k$  (compare eq. 3.86 in Bishop 2007), we find:

$$\begin{aligned} -\log p(\mathbf{z}|\sigma, \gamma, \mathbf{g}) &= k + n \log \sigma + n \log \gamma + \frac{1}{2} \log |\mathbf{I}| + \frac{1}{2} \log |\mathbf{\Gamma}| + \frac{1}{2} \log |\sigma^{-2} \mathbf{I}^{-1} + \gamma^{-2} \mathbf{\Gamma}^{-1}| \\ &+ \frac{1}{2\sigma^2} (\mathbf{z} - \bar{\mathbf{w}})'(\mathbf{z} - \bar{\mathbf{w}}) + \frac{1}{2\gamma^2} \bar{\mathbf{w}}' \mathbf{\Gamma}^{-1} \bar{\mathbf{w}} \\ &= k + \frac{1}{2} \log |\sigma^2 \mathbf{I} + \gamma^2 \mathbf{\Gamma}| + \frac{1}{2\sigma^2} (\mathbf{z} - \bar{\mathbf{w}})'(\mathbf{z} - \bar{\mathbf{w}}) + \frac{1}{2\gamma^2} \bar{\mathbf{w}}' \mathbf{\Gamma}^{-1} \bar{\mathbf{w}} \end{aligned} \quad (19)$$

The leading term measures the model complexity, while the trailing terms measure the misfit of the model to the data; the optimal hyperparameters strike the best balance between model fit and complexity by minimizing their sum. To find optimal hyperparameters, the negative log marginal likelihood is minimized numerically.

Here circulant matrices are particularly useful in two ways. First, the normally problematic log determinant appearing in the marginal likelihood has the special form  $\log |\sigma^2 \mathbf{I} + \gamma^2 \mathbf{\Gamma}| = \sum_i \log(\sigma^2 + \gamma^2 g_i)$  (see Circulant Matrices section for derivation), and is thus inexpensive to compute. Secondly, in the usual approach to evidence optimization (Bishop 2007), we find a MAP solution based on fixed hyperparameters and then determine optimal hyperparameters based on a fixed MAP solution, iterating until convergence. This iterative approach is taken because computing a MAP solution is usually expensive.

In contrast, the error term  $\frac{1}{2\sigma^2} (\mathbf{z} - \bar{\mathbf{w}})' (\mathbf{z} - \bar{\mathbf{w}}) + \frac{1}{2\gamma^2} \bar{\mathbf{w}}' \mathbf{\Gamma}^{-1} \bar{\mathbf{w}}$  can be computed entirely in the Fourier domain as the discrete Fourier transform is an orthogonal transform, and thus preserves inner products up to a constant:  $\mathbf{a}' \mathbf{b} = \frac{1}{n} F(\mathbf{a})' F(\mathbf{b})$ . Remarkably, during evidence optimization, we do not need to perform any forward or inverse Fourier transforms. We found that this non-iterative approach, enabled by the choice of circulant matrices, was more computationally efficient than the usual iterative solutions by almost an order of magnitude.

We do, however, neglect the derivatives of  $\mathbf{z}$  with respect to the hyperparameters (recall that  $\bar{\boldsymbol{\varphi}}^k$  is a function of the hyperparameters), and we need to compensate for this fact. The complete algorithm is as follows:

```

Find optimal hyperparameters based on  $\mathbf{z} = \mathbf{y}$ 
While convergence of hyperparameters and evidence is not reached
    While convergence of auxiliary equation is not reached
        Estimate each  $\bar{\boldsymbol{\varphi}}^k, \bar{\mu}$ 
    End while
    Set  $\mathbf{z} = \mathbf{y} - \sum_k C(\mathbf{s}^k) \mathbf{B} \bar{\boldsymbol{\varphi}}^k - \bar{\mu}$ 
    Recompute hyperparameters based on  $\mathbf{z}$ 
End while
Return  $\mathbf{z}$ 

```

Convergence of hyperparameters is then usually reached in less than 5 iterations. This algorithm is implemented in the Matlab function `despikeLFP`. Our implementation is highly optimized and fast enough to be of practical use in day-to-day research. For instance, a wideband signal lasting about 3 minutes can be despiked in about 20 seconds on a medium-powered computer running 32-bit Matlab or in about 7 seconds on a high-powered computer running 64-bit Matlab on the Intel Core i7 platform. Computational times rise as  $n \log n$ , where  $n$  is the length of the wideband signal, because of the use of FFTs by the algorithm. The method can scale to arbitrarily long recordings by performing the despiking on short segments of data, an approach we detail in the Chunking section.

## Empirical estimate of $\mathbf{g}$

We have now shown how to optimize  $\sigma$  and  $\gamma$  using evidence optimization. There remains a vector of free hyperparameters  $\mathbf{g}$  which controls our assumptions on the frequency content of the LFP. Choosing this vector properly is crucial, since in essence the assumed frequency content of the LFP is the only means through which the model can discriminate which portion of the STA around the time of a spike is artifactual and which portion is due to legitimate spike-LFP correlations (see Relationship to spike removal by STA subtraction section and Figure S4 for more on this).

If  $p(\mathbf{w}) = N(\mathbf{w}|0, \gamma^2 \mathbf{\Gamma})$ , then the covariance of  $\mathbf{w}$  is  $\gamma^2 \mathbf{\Gamma}_{ij} = E_{\mathbf{w}}(w_i w_j)$ , where  $E_{\mathbf{w}}$  denotes the expected value over all  $\mathbf{w}$ . But because we constrain the prior matrix to be circulant, the covariance of  $\mathbf{w}$  is completely described by its autocovariance,  $\gamma^2 \mathbf{\Gamma}_{ij} = E_{\mathbf{w}}(E_k(w_k w_{[k+i-j]}))$ . The Fourier transform of this autocovariance is the expected power spectral density (PSD) of  $\mathbf{w}$ , and thus we have:

$$E_{\mathbf{w}}(|F(\mathbf{w})|^2) \propto \mathbf{g} \quad (20)$$

Thus,  $\mathbf{g}$  should be matched to the expected PSD of the LFP. Here we have two difficulties. First, we never actually observe the LFP, only the wideband signal. Second, we typically observe only a handful of such wideband signals, thus even if  $E_{\mathbf{w}}(|F(\mathbf{w})|^2) = E_{\mathbf{y}}(|F(\mathbf{y})|^2)$  because there are no spikes or noise in our recording, the mean empirical PSD of a handful of wideband signals is very noisy.

We resolved these issues by using our knowledge of the properties of the LFP and the wideband signal. We know that in a certain range of frequencies where the LFP has most of its power, say 1-150 Hz, it account for most of the power in the PSD of the wideband signal and therefore  $E_{\mathbf{w}}(|F(\mathbf{w})|^2) \approx E_{\mathbf{y}}(|F(\mathbf{y})|^2)$  in this frequency range. We therefore fit a function to the PSD of the wideband signal in the range of 1 to 150 Hz and extrapolated this function at lower and higher frequencies to obtain  $\mathbf{g}$ . Extrapolation with highly nonlinear functions is unadvisable, so we used functions which were constant at the lowest frequencies and linear in log-log space at higher frequencies, consistent with previous reports (Bedard and Destexhe 2009).

We found that the function  $-\exp(1 + \log x)$ , which is constant for small  $x$  and decreases linearly for large  $x$  to provide an excellent fit to the PSD of the wideband signal within the range of 1 to 150 Hz. A procedure for fitting this function to a PSD is implemented in the Matlab function `fitLFPpowerSpectrum`. An example of such a fit is shown in Figure S5. Note that we purposefully set  $\mathbf{g}$  to be lower than the empirical PSD of the wideband signal at the highest frequencies, as we know that most of the power at these frequencies is actually due to spikes. Some recordings may require a different function to be fit to the PSD of the wideband signal, for example when there is a peak in the PSD in a range of frequencies. Such a peak could happen for a variety of reasons, for example because of a low-pass filter in the recording system whose cutoff overlaps the PSD of the LFP or because of intrinsic properties of the recorded brain region. In that case, a sum of the function  $-\exp(1 + \log x)$  and a logistic function could be used to fit the PSD of the LFP.

We obtained excellent results with this method of choosing  $\mathbf{g}$  (see Figure S4 for an example). Other functional forms that closely followed the envelope of the PSD of the wideband signal also performed well. Choosing a  $\mathbf{g}$  that was not adjusted to the statistics of the signal yielded less satisfactory results. For example, a choice of  $\mathbf{g} = 1$  for frequencies smaller than a cutoff of 200 Hz and a vanishingly small value elsewhere performed poorly. We therefore highly recommend that  $\mathbf{g}$  be selected on the basis of empirical PSD of the wideband signal.

### Choice of basis

A final implicit set of hyperparameters is the basis  $\mathbf{B}$ . Our algorithm assumes that this basis has the form  $[\mathbf{B}_s; \mathbf{0}]$ , which as we showed earlier is an appropriate form when we assume that spikes are finite. The height of  $\mathbf{B}_s$  corresponds to the duration of spikes measured in samples. As spike-sorting algorithms traditionally use snippets ranging in duration from about 1.5 to 3 ms, it is safe to assume that 3 ms is an upper bound for the duration of spike waveforms. Note that this duration does not correspond to the physical duration of a spike, which is shorter, but rather to the duration of the measured spike waveform, which is affected by the filters of the recording system. We chose the spike length to be equal to 3 ms (30 samples), and aligned spikes so that their peak was located at the 11<sup>th</sup> sample.

Thus, the basis  $\mathbf{B}_s$  was taken to be the identity matrix of size 30x30. As we recorded in areas where neurons fire at high rates, and our sampling rate was relatively low, the spike waveforms  $\boldsymbol{\varphi}^k$  were well constrained in this basis. When recording at higher sampling rates, or in areas with low firing rates, however, spike waveforms may be poorly constrained. In this case,  $\mathbf{B}_s$  can be chosen to be undercomplete, thus parametrizing spike waveforms in a low-dimensional subspace. For example, we could express the waveforms in a spline basis with a higher density of knots around the time of the peak of spikes than elsewhere.

We must note, however, that this method has its limitations. Our algorithm is not well adapted to short recordings that contain a handful of spikes (say, less than 100), as it needs a sufficient amount of data to constrain the spike waveforms. When despiking trial data, therefore, one should perform the despiking on a continuous wideband signal, splitting the data into smaller chunks for trial analysis afterwards, if necessary.

Our implementation of the despiking algorithm automatically multiplies the chosen basis by a whitening matrix  $\mathbf{W}$  obtained through a singular value decomposition, so that the basis internally by the algorithm is orthogonal,  $\mathbf{B}_s' \mathbf{W}' \mathbf{W} \mathbf{B}_s = \mathbf{I}$ . This tends to improve numerical conditioning appreciably. The implementation then expresses the spike waveforms  $\boldsymbol{\varphi}^k$  in the original basis, so this is completely transparent to the end-user.

### Chunking

When the signal is too long, it becomes inconvenient to perform the matrix operations required to estimate the local field potential. In addition, in long recordings electrode drift can cause spike waveforms to shift. We addressed these issues by splitting the signal into overlapping chunks, estimating the local field potential for each chunk, then stitching the results back to obtain the complete despiked signal.

The chunking scheme is illustrated graphically in Figure S6. Here we show a signal which is much shorter than one would use in reality for ease of visualization. The signal is split into overlapping chunks. Within each chunk, the signal is multiplied by an analysis window and added to a time reversed version of itself multiplied by 1 minus this window, thus creating a composite signal. The analysis window has a trapezoid shape:

$$f(i) = \begin{cases} \frac{1}{2} + \frac{1}{2} \frac{i}{N \cdot \text{overlap}} & i < N \cdot \text{overlap} \\ 1 & N \cdot \text{overlap} < i < N - N \cdot \text{overlap} \\ 1 - \frac{1}{2} \frac{i - (N - N \cdot \text{overlap})}{N \cdot \text{overlap}} & i > N - N \cdot \text{overlap} \end{cases} \quad (21)$$

Here  $N$  is the length of a segment and  $\text{overlap}$  is a variable that controls the degree of overlap between chunks. The edges are thus blended together to avoid discontinuities. The composite signal within each chunk is then despiked as in the previous sections. The despiked signals are put back together by multiplying each signal by a synthesis window and summing the windowed signals together. The synthesis window also has a trapezoid shape:

$$h(i) = \begin{cases} 0 & i < 2N \cdot \text{overlap} \\ \frac{(i - 2N \cdot \text{overlap})}{N \cdot \text{overlap}} & 2N \cdot \text{overlap} < i < 3N \cdot \text{overlap} \\ 1 & 3N \cdot \text{overlap} < i < N - 3N \cdot \text{overlap} \\ 1 - \frac{i - (N - 3N \cdot \text{overlap})}{N \cdot \text{overlap}} & N - 3N \cdot \text{overlap} < i < N - 2N \cdot \text{overlap} \\ 0 & i > N - 2N \cdot \text{overlap} \end{cases} \quad (22)$$

The support of the synthesis window is smaller than the size of the analysis window, thus discarding the edges within each chunk. For the initial segment the analysis and synthesis windows are of a different shape to avoid artifacts at the beginning of the recording signal. The analysis window is given by:

$$f_{\text{first}}(i) = \begin{cases} 1 & i < N - 4N \cdot \text{overlap} \\ 1 - \frac{1}{2} \frac{i - (N - 4N \cdot \text{overlap})}{N \cdot \text{overlap}} & N - 4N \cdot \text{overlap} < i < N - 2N \cdot \text{overlap} \\ 0 & i > N - 2N \cdot \text{overlap} \end{cases} \quad (23)$$

And the synthesis window is:

$$h_{\text{first}}(i) = \begin{cases} 1 & i < N - 6N \cdot \text{overlap} \\ \frac{i - (N - 6N \cdot \text{overlap})}{N \cdot \text{overlap}} & N - 6N \cdot \text{overlap} < i < N - 5N \cdot \text{overlap} \\ 0 & i > N - 5N \cdot \text{overlap} \end{cases} \quad (24)$$

The analysis and synthesis windows for the last chunk are mirror inverses of those of the first chunk.

The number of chunks is determined by  $N$ , which is given by the user implicitly through the  $\mathbf{g}$  parameter, and the variable `overlap`, which is given explicitly. Because the number of chunks must be an integer, however, the method automatically adjusts the overlap upwards so that chunks are equispaced and that recovery windows add up to 1 everywhere. The method is implemented in Matlab as the function `despikelFPbyChunks`.

In simulations where ground truth was available, despiking by chunks gave essentially the same results as despiking an entire signal provided that chunks were large enough to obtain reliable estimates of spike waveforms. The chunk size should be a power of 2 for the fastest speeds as FFT routines are typically optimized for such cases (Frigo and Johnson 2005). We recommend using chunks that are a few minutes long (say 2-5 minutes) so that several hundred spikes will be present in each chunk.

### Relationship to LFP-spike correlation removal

Our algorithm is closely related to the algorithm of David et al. 2010 which removes correlations between LFPs and spikes. To make this relationship clearer, we rewrite equation (7) under the assumption of a single spiking neuron as:

$$\begin{aligned}\bar{\mathbf{w}} &= \mathbf{M}(\mathbf{y} - \mathbf{D}\bar{\boldsymbol{\varphi}} - \bar{\boldsymbol{\mu}}) \\ \bar{\mathbf{w}} &= F^{-1}(\mathbf{m}F(\mathbf{y})) - F^{-1}(\mathbf{m}F(\mathbf{D}\bar{\boldsymbol{\varphi}})) - a_1 \\ \text{despiked LFP} &= \text{LFP} - \text{smoothed contribution from spikes} - \text{offset}\end{aligned}\tag{25}$$

Here  $\mathbf{m} = \frac{\gamma^2 \mathbf{g}}{\sigma^2 + \gamma^2 \mathbf{g}}$  is a low pass filter. This can be compared to equation 7 of David et al. 2010:

$$\begin{aligned}L(t) &= L_0(t) - L_{\text{pred}}(t) \\ \text{despiked LFP} &= \text{LFP} - \text{portion of the LFP predictable by spikes}\end{aligned}\tag{26}$$

This suggests that the portion of the LFP predictable by spikes  $L_{\text{pred}}(t)$  defined in David et al. 2010 may be similar to our smoothed contribution from spikes  $F^{-1}(\mathbf{m}F(\mathbf{D}\bar{\boldsymbol{\varphi}}))$ . This connection is made clearer by joining equations (8), (9) and (25) together, and setting the basis  $\mathbf{B} = [\mathbf{I}_q; \mathbf{0}]$ :

$$\begin{aligned}\left( \mathbf{k} * \mathbf{I}_q - \frac{q^2 (\mathbf{s}' \hat{\mathbf{1}})^2}{n} (F^{-1}(1 - \mathbf{m})' \hat{\mathbf{1}}) \hat{\mathbf{1}}_q' \right) \bar{\boldsymbol{\varphi}} &= r [\mathbf{I}_q, \mathbf{0}] F^{-1}((1 - \mathbf{m})F(\mathbf{y})\overline{F(\mathbf{s})}) - a_1 \\ \mathbf{K}\bar{\boldsymbol{\varphi}} &= r [\mathbf{I}_q, \mathbf{0}] F^{-1}((1 - \mathbf{m})F(\mathbf{y})\overline{F(\mathbf{s})}) - a_2 \\ \bar{\boldsymbol{\varphi}} &= r \mathbf{K}^{-1} [\mathbf{I}_q, \mathbf{0}] F^{-1}((1 - \mathbf{m})F(\mathbf{y})\overline{F(\mathbf{s})}) - a_3\end{aligned}\tag{27}$$

$a_i$  are offsets and  $\mathbf{K}$  is a convolution matrix derived from  $\mathbf{a} = F^{-1}((1 - \mathbf{m})|F(\mathbf{s})|^2)$  by time shifting,  $K_{ij} = a_{[i-j+1]}$ . Hence our smoothed contribution from spikes is:

$$\begin{aligned}
F^{-1}(\mathbf{m}F(D\bar{\boldsymbol{\varphi}})) &= pF^{-1}\left(\mathbf{m}F(\mathbf{s})F\left(\begin{bmatrix} \mathbf{I}_q & \mathbf{0} \end{bmatrix}' \mathbf{K}^{-1} \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \end{bmatrix} F^{-1}((1 - \mathbf{m})F(\mathbf{y})\overline{F(\mathbf{s})})\right)\right) - a_4 \\
&= pF^{-1}\left(\mathbf{m}F(\mathbf{s})F\left(\begin{bmatrix} \mathbf{K}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} F^{-1}((1 - \mathbf{m})F(\mathbf{y})\overline{F(\mathbf{s})})\right)\right) - a_4
\end{aligned} \tag{28}$$

Meanwhile, the portion of the LFP predictable by spikes as defined in David et al. 2010 can be written in our notation as:

$$L_{\text{pred}}(t) = \kappa F^{-1}\left(F(\mathbf{s})F\left(\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} F^{-1}\left(\mathbf{m} \frac{F(\mathbf{y})\overline{F(\mathbf{s})}}{|F(\mathbf{s})|^2}\right)\right)\right) \tag{29}$$

Notice the similarity between these two equations. This is especially striking when taking into account that  $\mathbf{K}^{-1}$  is related, but not equal to  $\boldsymbol{\Lambda} = F^{-1}\left(1/((1 - \mathbf{m})|F(\mathbf{s})|^2)\right)$ . Now suppose that  $\mathbf{K}^{-1}[\mathbf{I}_q, \mathbf{0}] \approx [\mathbf{I}_q, \mathbf{0}]C'(\boldsymbol{\Lambda})$ . This would hold if the support of  $\boldsymbol{\Lambda}$  is smaller than the length of spikes. Hence, taking  $q$  to be large, our smoothed spike contributions becomes:

$$\approx pF^{-1}\left(\mathbf{m}F(\mathbf{s})F\left(\begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} F^{-1}\left(\frac{F(\mathbf{y})\overline{F(\mathbf{s})}}{|F(\mathbf{s})|^2}\right)\right)\right) - a_4 \tag{30}$$

If we take the further step of assuming that the support of  $F^{-1}(\mathbf{m})$  is smaller than the length of spikes, the two equations become equal, modulo a different handling of offsets. Hence the methods are essentially equivalent in the limit of long spikes. With short spikes, however, the order of the filters and the projection matrix in each method is different, and hence they handle edges around the time of spikes differently. Specifically, in our method the smoothed spike contribution is obtained by first taking the STA of a high-passed filtered wideband signal, keeping the first  $q$  elements of this STA, then undoing the high-pass filtering and compensating for the spike-train autocorrelation to obtain a spike waveform, and finally convolving this waveform with the spike train and smoothing it. In contrast, the David et al. method takes a STA of the low-pass filtered wideband signal, compensates for the autocorrelation of the signal, then keeps the first  $q$  elements of this STA to obtain the smoothed waveform which is convolved with the spike train to obtain a smoothed spike contribution.

For small  $q$  (short spikes), keeping the  $q$  first elements of the STA of a low-passed filtered signal (the smoothed mean spike) will lead to artifacts at the edges of spikes. The exact way that the edges are handled in the David et al. method is further complicated in this method by the use of Hanning windows to perform the required Fourier transforms. The edge handling of the method of David et al. 2010 should not be considered a design flaw, as the authors were interested in the scenario where  $q$  is large,

in which case the edge handling is sufficient; it does, however, render their method inappropriate for removing spikes in the scenario where  $q$  is small (short spikes).

## Appendix - Properties of circulant matrices

A circulant matrix is defined as:

$$C(\mathbf{x}) = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & & x_{n-1} \\ x_{n-1} & x_n & x_1 & & x_{n-2} \\ \vdots & & & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix} \quad (31)$$

By this definition, circulant matrices are closed under addition and  $C(\mathbf{a}) + C(\mathbf{b}) = C(\mathbf{a} + \mathbf{b})$ . The transpose of a circulant matrix is circulant. The product of two circulant matrices is circulant. Note also that the identity matrix  $\mathbf{I}$  is circulant. Multiplication of the transpose of a circulant matrix by a vector corresponds to a circular convolution:

$$C'(\mathbf{a})\mathbf{b} = (a \circledast b)_i = \sum_j a_{[i-j]} b_j \quad (32)$$

Here  $a_{[k]} = a_{(k \bmod n)+1}$  with  $n$  being the length of  $\mathbf{a}$ . By the circular convolution theorem we have that:

$$C'(\mathbf{a})\mathbf{b} = F^{-1}(F(\mathbf{a})F(\mathbf{b})) \quad (33)$$

Here  $F(\mathbf{a})$  is the discrete Fourier transform of  $\mathbf{a}$ . This implies the following properties:

$$\begin{aligned} C'(\mathbf{a})\mathbf{b} &= C'(\mathbf{b})\mathbf{a} \\ C(\mathbf{a})\mathbf{b} &= F^{-1}\left(\overline{F(\mathbf{a})}F(\mathbf{b})\right) \\ (C'(\mathbf{a}))^{-1} &= C'\left(F^{-1}\left(\frac{1}{F(\mathbf{a})}\right)\right) \end{aligned} \quad (34)$$

The determinant of a circulant matrix can be found by noting that:

$$\begin{aligned} C'(\mathbf{a})\mathbf{b} &= F^{-1}(F(\mathbf{a})F(\mathbf{b})) \\ &= \hat{\mathbf{F}}^{-1}\text{diag}(\hat{\mathbf{F}}\mathbf{a})\hat{\mathbf{F}}\mathbf{b} \end{aligned} \quad (35)$$

Here  $\hat{\mathbf{F}}$  is the discrete Fourier transform matrix (DFT matrix) which maps a vector onto its discrete Fourier transform. Hence:

$$\begin{aligned}
|C'(\mathbf{a})| &= |\hat{\mathbf{F}}^{-1} \text{diag}(\hat{\mathbf{F}}\mathbf{a}) \hat{\mathbf{F}}| \\
&= |\hat{\mathbf{F}}^{-1}| |\text{diag}(\hat{\mathbf{F}}\mathbf{a})| |\hat{\mathbf{F}}| \\
&= |\text{diag}(\hat{\mathbf{F}}\mathbf{a})| |\hat{\mathbf{F}}\hat{\mathbf{F}}^{-1}| \\
&= \prod_i \text{abs}(\hat{\mathbf{F}}\mathbf{a})_i
\end{aligned} \tag{36}$$

This expression is valid for  $\mathbf{a} \in \mathbb{R}^N$ . Therefore the log-determinant of a circulant matrix is:

$$\log|C'(\mathbf{a})| = \sum_i \log |F(\mathbf{a})_i| \tag{37}$$

A symmetric circulant matrix corresponds to circular convolution by a symmetric kernel, and its corresponding Fourier coefficients are real. A symmetric circulant matrix whose corresponding Fourier coefficients are positive is positive definite and therefore is a valid covariance matrix.

## List of supplementary figures

### Supplementary Figure 1: Summary of results for all spike-amplitude simulations

A) SNRs and B) Mean Spike Field Coherence (SFC) for various normalized spike amplitude simulation cases for the wide band signal and three different bands of the LFP, before (upper panels) and after (lower panels) spike removal.

### Supplementary Figure 2: Summary of results for all firing-rate simulations

A) SNRs and B) Mean Spike Field Coherence (SFC) for various mean firing rate simulation cases for the wide band signal and three different bands of the LFP, before (upper panels) and after (lower panels) spike removal.

### Supplementary Figure 3: Effect of spike removal in an example recording

A) wideband signal before and after despiking B) low-pass filtered wideband signal before and after despiking. Note that although the spike artifact is barely visible after low-pass filtering, it is highly stereotyped and always in sync with spike times. Therefore it can seriously bias analyses which look at spike-LFP temporal relationships, as we show in the main text.

### Supplementary Figure 4: Necessity of smoothness assumption in a simulation

A) STA of despiked wideband signal under an assumption of smoothness and without this assumption. Note the prominent artifact around 0 without the assumption of smoothness. The STA of the signal uncorrupted by spikes is shown for comparison B) STA of smoothed despiked wideband signal under an assumption of smoothness and without this assumption. The artifact around 0 can masquerade as a legitimate feature of the spike-LFP relationship. C) spike waveform recovered by the model under an assumption of smoothness and without the assumption. D) difference of the two spike waveforms. The two recovered spike waveforms differ mostly by DC, linear and quadratic trends, confirming the intuition that with the smoothness assumption the low-frequency components in the spike waveform are now correctly attributed to the LFP.

### Supplementary Figure 5: Choice of $g$

$g$  is constrained to be a low-complexity parametric function. It should match the PSD of the wideband signal in the range of 1 to 150 Hz. At higher frequencies,  $g$  can undershoot the PSD of the wideband signal; much of the power at these frequencies is attributable to noise and spikes.

### Supplementary Figure 6: Chunking procedure

In the chunking procedure, the signal is split into overlapping chunks (top). For each chunk, a composite signal is formed by the addition of a windowed signal and time-reversed version of this signal windowed with a complementary window. The composite signal is fed into the despiking algorithm. The recovered signal chunk is then windowed through a synthesis window. These windowed despiked chunks are added together to recover the complete signal. The recovery windows add up to 1 at every point in time.

## Supplementary References

**Bedard C, and Destexhe A.** Macroscopic models of local field potentials and the apparent 1/f noise in brain activity. *Biophys J* 96: 2589-2603, 2009.

**Bishop C.** *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2007.

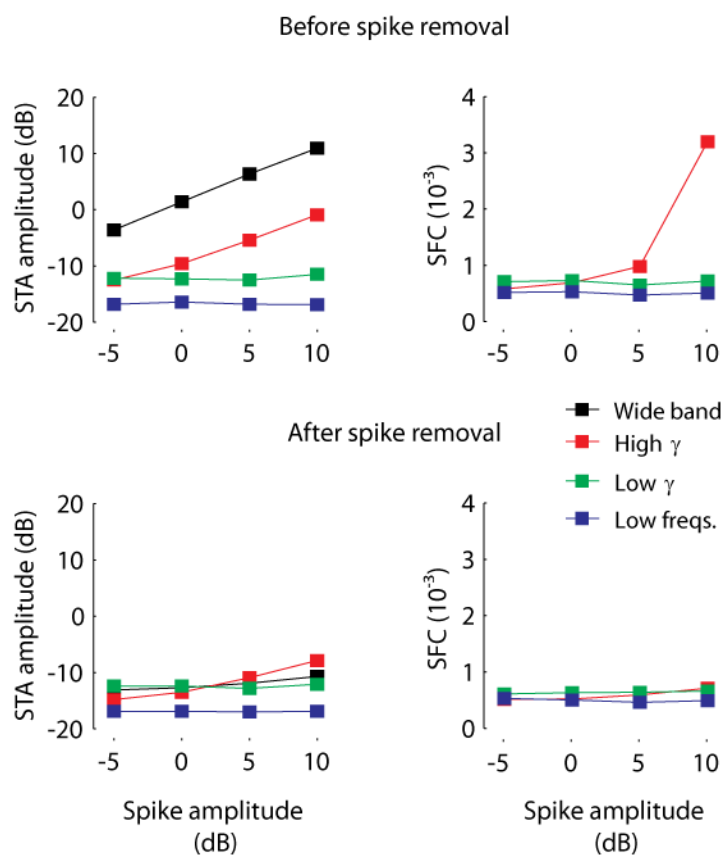
**Frigo M, and Johnson SG.** The design and implementation of FFTW3. *Proc IEEE* 93: 216-231, 2005.

**Pesaran B, Pezaris J, Sahani M, Mitra P, and Andersen R.** Temporal structure in neuronal activity during working memory in macaque parietal cortex. *nature neuroscience* 5: 805-811, 2002.

**Wu MC, David SV, and Gallant JL.** Complete functional characterization of sensory neurons by system identification. *Annu Rev Neurosci* 29: 477-505, 2006.

**Zanos TP, Zanos SP, Courellis SH, Marmarelis VZ, and Ojemann GA.** Nonlinear dynamic modeling of the relationship between local field potentials and neural discharge in human temporal cortex. In: *Society for Neuroscience Conference*. Atlanta, GA: 2006.

Figure S1



## Figure S2

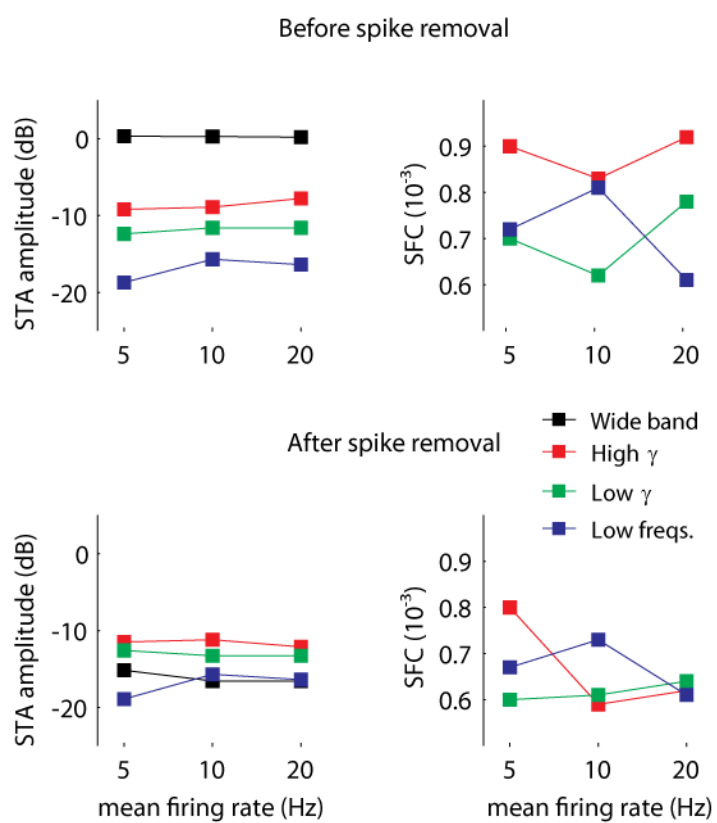


Figure S3

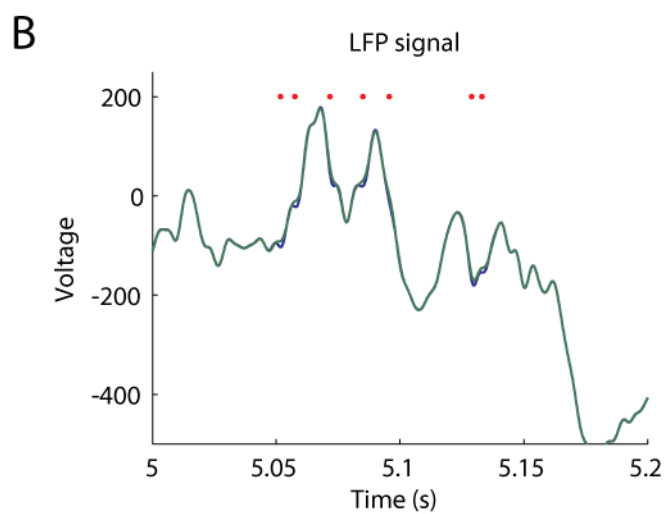
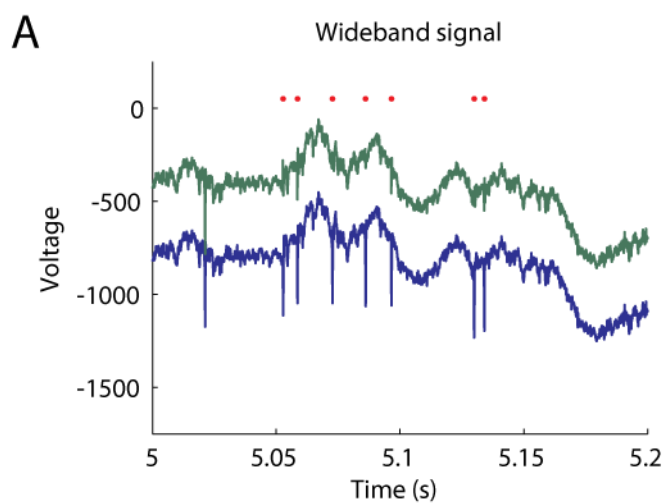


Figure S4

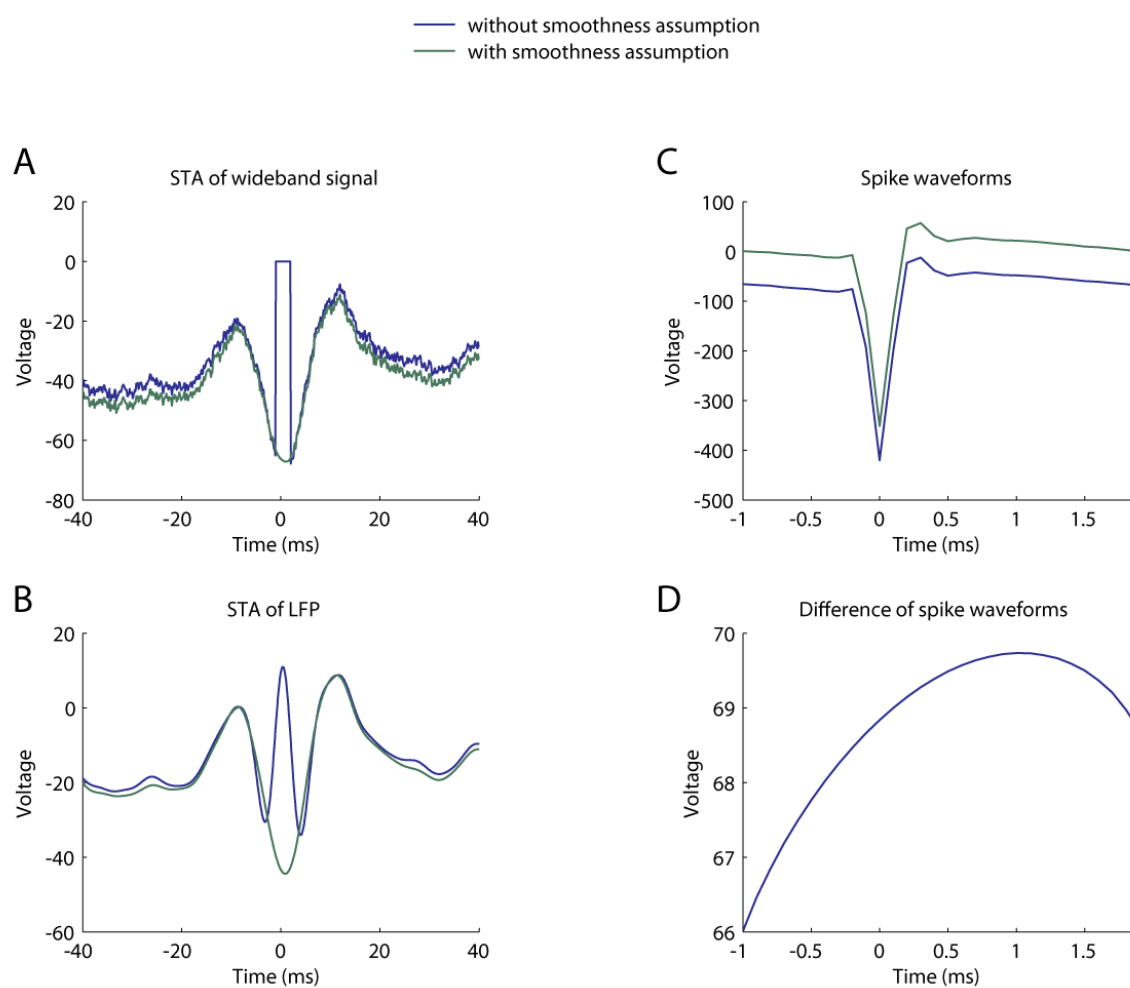


Figure S5

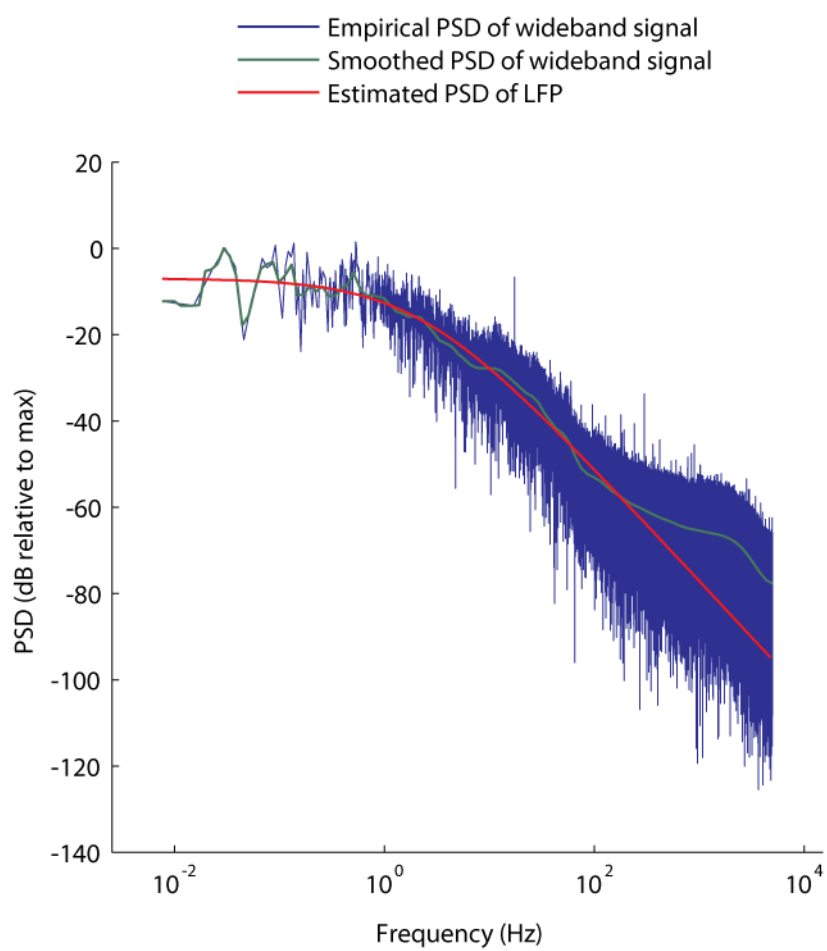


Figure S6

