KU LEUVEN

FACULTEIT INGENIEURSWETENSCHAPPEN

Real-time detection of voltage patterns in the brain

Tomas Fiers

Abstract – 30 Nov

Contents

Abbreviations

BPF   Band-pass filter. See chapter 2.

CA1   "Cornu Ammonis", subregion 1. Region in the hippocampus where voltages are recorded from (see **????**).

CA3   "Cornu Ammonis", subregion 3. Region in the hippocampus (see **????**). CA3 sends many axons (called "Schafer collaterals") to CA1.

GEVal   Generalized eigenvalue. See section 3.2.

GEVec   Generalized eigenvector. See section 3.2.

IQR   Interquartile range. A measure of the spread of a set of one-dimensional values, that is robust to outliers. Difference between the 75th and the 25th data percentile.

KDE   Kernel density estimate.

LFP   Local field potential. The extracellular electric potential (see **??**).

RMS   Root-mean-square. $\sqrt{\langle x_t^2 \rangle}$ for a signal x_t.

RNN   Recurrent neural network. See chapter 4.

SNR   Signal-to-noise ratio. See section 3.2.

SOTA   State of the art. The algorithm currently used for SWR detection, namely an online single channel band-pass filter.

SWR   Sharp wave-ripple. The pattern in the LFP that we want to detect in real-time. See **??**.

Symbols

Notation

y Scalars are denoted in lowercase italic.

$\mathbf{z}$ Vectors are denoted in lowercase boldface.

$\mathbf{A}$ Matrices are denoted in uppercase boldface.

$\langle \cdot \rangle$ Time-average of a signal.

$\odot$ Elementwise multiplication. ("Hadamard product").

$\sigma(\cdot)$ Sigmoid 'squashing' function, $\mathbb{R} \to (0, 1)$. $\sigma(x) = \frac{1}{1+\exp(-x)}$.

$\tanh(\cdot)$ Hyperbolic tangent, $\mathbb{R} \to (-1, 1)$. $\tanh(x) = 2\,\sigma(x) - 1$.

Signals

$\mathbf{z}_t$ Digitized LFP sample at discrete time step t. $\mathbf{z}_t \in \mathbb{R}^C$, with C the number of channels (i.e. the number of electrodes simultaneously recorded from). Input to an SWR detection algorithm.

o_t Output signal of an SWR detection algorithm, $\in \mathbb{R}$.

n_t 'Envelope'. Transformation of o_t, so that it is constrained to $\mathbb{R}^+$. Should be high when the corresponding input sample $\mathbf{z}_t$ is part of an SWR segment, and low when it is not. $n_t = |o_t|$ for online linear filters; $n_t = \sigma(o_t)$ for the RNN's of chapter 4.

y_t Binary target signal, used when training data-driven SWR detection algorithms. We define $y_t = 1$ when the corresponding input sample $\mathbf{z}_t$ is part of an SWR segment, and $y_t = 0$ when it is not.

7

Measures & parameters

T Detection threshold applied to the envelope n_t. $T \in [\min n_t, \ \max n_t]$. Each threshold T yields a different P-value, R-value, F_1-value, etc.

P Precision. Also known as positive predictive value. The fraction of correct detections, out of all detections.

R Recall. Also known as sensitivity, hit rate, or true positive rate. The fraction of detected reference SWR segments, out of all reference SWR segments.

F_β F-score: weighted harmonic mean of recall and precision. $F_\beta = \frac{(1+\beta^2)PR}{\beta^2 P + R}$. Measures detection performance "for a user who attaches β times as much importance to recall as to precision." [1]

F_1 F-score where recall and precision are weighted equally.

Introduction − 16 Nov [9p]

1.1 Closed-loop brain-computer interfaces [1p]

1.2 Sharp wave-ripples [6p]

Description [2p]

Scientific importance [2p]

Biophysics [1p]

Closed-loop technology [1p]

1.3 Problem statement [1p]

1.4 Thesis overview [1p]

Single-channel linear filtering − 9 Nov [10p]

2.1 Offline labelling of SWR segments [3p]

To quantify the performance of a sharp wave-ripple (SWR) detection algorithm, we need an evaluation or 'test' recording, annotated with the segments of time when actual SWR events were present. This section describes how such annotations can be made, and how our data specifically was annotated.

SWR's are an empirical phenomenon of hippocampal area CA1, 'defined' by what their voltage traces look like. In other words, there is no ground truth available to know when SWR's occur. Scientists looking to annotate their LFP recordings with SWR segments therefore have to rely either on judgement calls by human labellers, or on an automated, offline SWR detection algorithm. The former could be considered more subjective, and is definitely more labour intensive than the latter – especially if multiple scientists are consulted to obtain an consensus labelling. Most studies use an automated, offline algorithm to detect SWR segments (see appendix A for some examples).

'Offline' here means that SWR detection happens after the recording has been completed, and that there are thus no real-time constraints on the detection algorithm. This means that 1) there are no hard bounds on algorithm execution time, and 2) that the algorithm can use information 'from the future': when deciding whether a recording sample $\mathbf{x}_t$ belongs to an SWR segment, it can use samples $\mathbf{x}_{t_f}$ in its decision that occured after $\mathbf{x}_t$ (i.e. $t_f > t$), instead of using only past samples $\mathbf{x}_{t_p}$ (where $t_p \leq t$).

The main steps of the offline SWR detection algorithm that most studies use (such as the ones cited in appendix A) can be summarized as follows:

1. Use a single channel of input data; namely from an electrode in the pyramidal cell layer of CA1, where the ripple part of SWR's is most strongly present. (We will denote this voltage signal with z_t);

2. Band-pass filter the recording to retain only 'ripple' frequencies. (We will denote the filter output as o_t);

3. Obtain the envelope of the band-pass filtered signal. (We will denote this envelope with n_t);

4. Calculate a 'high' and a 'low' threshold to apply to n_t, based on the average envelope magnitude and two custom multipliers;

5. Define ripple events as times when the envelope crossed the high threshold;

6. Define the start and end times of each such ripple event as the closest low threshold crossings of the envelope.

Note that this procedure only detects ripples, and not sharp waves.[1]

2.2   State-of-the-art online SWR detectors [3p]

2.3   Quantifying detector performance [4p]

We often want to summarize the accuracy of a detector in a single number. We choose the maximum F_1-score of the detector as this number.

---

[1]Although interestingly, the sharp wave part of sharp wave-ripples was discovered before the ripple part [2, p. 1].
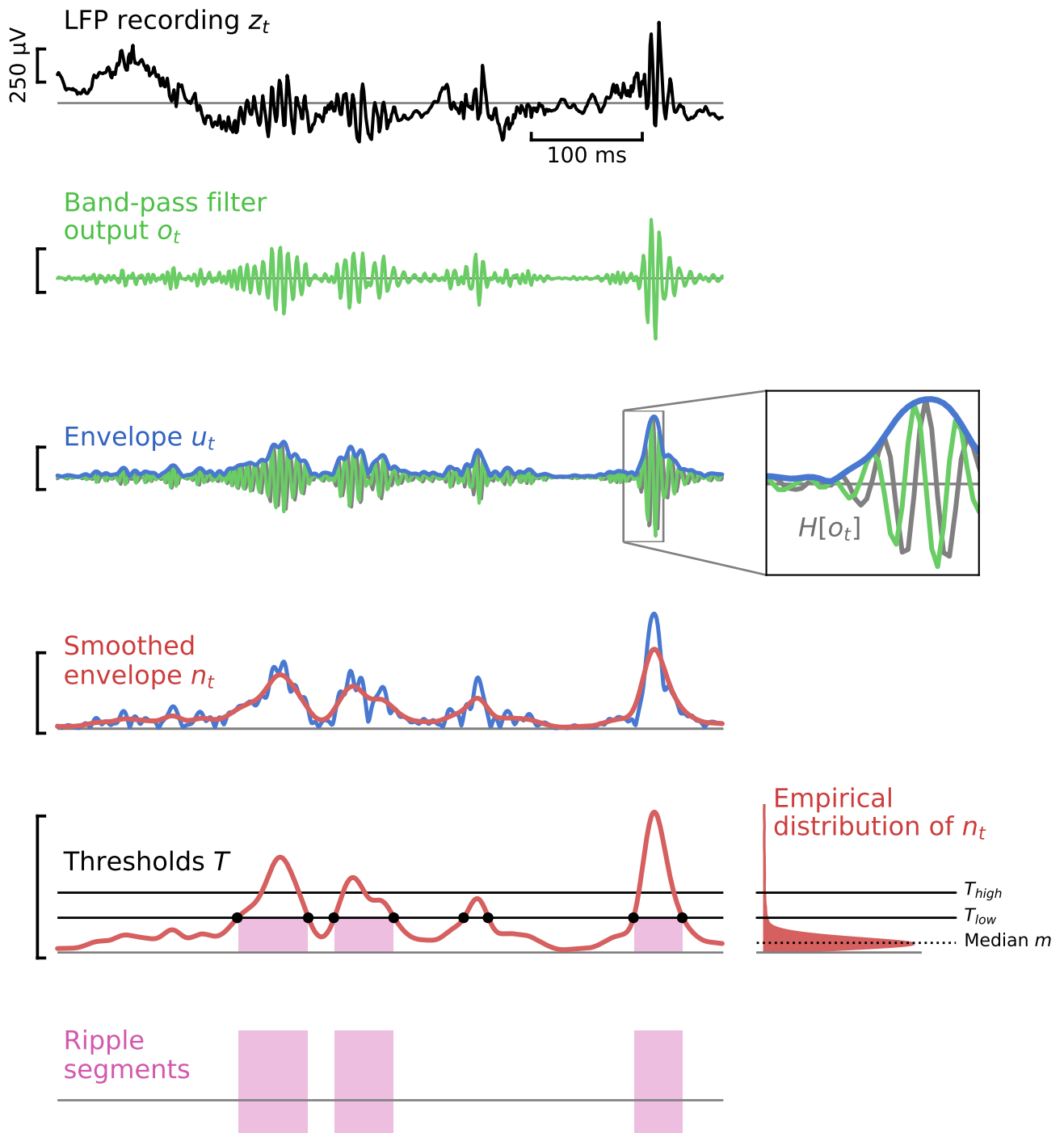
Figure 2.1: **Steps for automated, offline SWR labelling**. See text for details. Each vertical scalebar indicates the same voltage range. $H[o_t]$ denotes the Hilbert transform of o_t. Note its phase lag of $90°$ with respect to o_t. In the second panel from the bottom, note the two threshold crossings of T_{low} (marked with black dots) that did not result in a ripple segment, because the second threshold T_{high} was not crossed. The distribution of the envelope n_t was estimated using the entire dataset (and not just the displayed time slice).

Multi-channel linear filtering − 26 Oct [10p]

3.1   Data-driven algorithms [1p]

In this and the following chapter, we describe *supervised*, or data-driven SWR detection algorithms: they require training data $\mathbf{z}_t^{\text{train}}$, and an associated labelling y_t^{train} which marks the presence of an SWR event in $\mathbf{z}_t^{\text{train}}$, for every discrete time sample t. We arbitrarily define $y_t \in \{0, 1\}$, with $y_t = 1$ when the corresponding input sample $\mathbf{z}_t$ is part of an SWR segment, and $y_t = 0$ when it is not.

The problem of obtaining such a labelling y_t for some recording data $\mathbf{z}_t$ is the topic of section 2.1. Training labels can be obtained either by human expert labellers, or by using an automated *offline* SWR detection algorithm, where we assume that the automated labelling corresponds well to a supposed human expert labelling. In this thesis, we use the automated labelling method of section 2.1 to generate target labellings y_t^{train}.

Before a supervised algorithm can be used for real-time detection, its parameters have to be 'tuned'. This is done using a training dataset $(\mathbf{z}_t^{\text{train}}, y_t^{\text{train}})$, during the so called *training phase*. Parameters are changed such that the algorithm's output o_t for an input $\mathbf{z}_t^{\text{train}}$ matches the target labelling y_t^{train} well. Sections 3.2 and 4.2 describe how this tuning can be done for two concrete detection algorithms.

The hope is that the trained algorithm also performs well on input data $\mathbf{z}_t^{\text{test}}$ not part of the training set. That is, that the algorithm has good *generalization performance*. When this is not the case and the algorithm is tuned so that it only performs well on the training data, we say that the algorithm has been *overfit*. Often, so called *regularization* methods exist to discourage overfitting on the training data. Some example regularization methods are discussed in sections 3.5 and 4.3.

3.2   Linear signal-to-noise maximisation [2p]

In this chapter, we search for a linear combination of channels that yields an output signal o_t useful for sharp wave-ripple detection. More precisely, we search for a vector
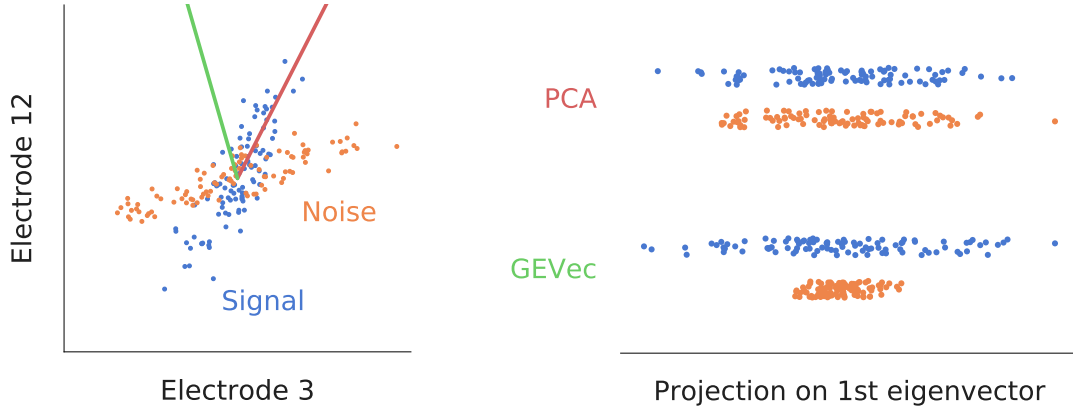
Figure 3.1: **Linear signal-to-noise maximisation**. Toy example to illustrate the generalized eigenvector approach to signal detection. *Left*: multi-channel time-series data plotted in 'phase space' (meaning without time axis), with blue dots representing samples where the signal was present, and orange dots representing samples where it was not. Actually toy data drawn from two 2-dimensional Gaussian distributions with different covariance matrices. Red vector: first eigenvector of the signal covariance matrix (also known as the first principal component). Green vector: first generalized eigenvector of the signal and noise covariance matrices. *Right*: Projection of both data sets on both the ordinary eigenvector ("PCA") and the generalized eigenvector ("GEVec"). The ratio of the projected signal data variance versus the projected noise data variance is maximised for the GEVec case.

$\mathbf{w} \in \mathbb{R}^C$ in channel (or electrode) space to project the samples $\mathbf{z}_t \in \mathbb{R}^C$ on, so that the output signal

$$o_t = \mathbf{w}^T \mathbf{z}_t \tag{3.1}$$

has high variance (or power) during SWR events, and low variance outside them.[1] This principle is illustrated with a two-dimensional toy dataset in fig. 3.1. We can then detect SWR events using threshold crossings of the envelope of o_t, as discussed in section 2.2.

The next two sections describe how this vector $\mathbf{w}$ can be found.

The optimisation problem

Suppose all training samples $\mathbf{z}_t^{\text{train}}$ are gathered and divided over two data matrices $\mathbf{S} \in \mathbb{R}^{C \times N_S}$ and $\mathbf{N} \in \mathbb{R}^{C \times N_N}$, where $\mathbf{S}$ (for 'signal') contains all N_S samples of $\mathbf{z}_t^{\text{train}}$ where an SWR is present, and $\mathbf{N}$ (for 'noise') contains all N_N other samples. (These matrices can be easily constructed by concatenating segments from $\mathbf{z}_t^{\text{train}}$).

---

[1]We assume that the input signals are zero-mean, such that the power P of the output signal equals its variance: $P_o = \langle o_t^2 \rangle = \langle (o_t - \mu_o)^2 \rangle = \text{Var}(o_t)$ when $\mu_o = 0$, which is the case for zero-mean input channels: $\mu_o = \langle o_t \rangle = \langle \mathbf{w}^T \mathbf{z}_t \rangle = \sum_i w_i \langle z_{t,i} \rangle = 0$ when $\langle z_{t,i} \rangle = 0$ for all channels i.
This zero-mean assumption is reasonably well fulfilled for the analysed recording: the sample values of a 10-second moving average of the recording are near zero (median -0.02 μV, IQR 0.13 μV. In comparison, the RMS-value of the recording is 210 μV).
Input data that is not zero-mean can be readily transformed to be so (even in an online setting), by subtracting a (moving) average from the input signal.

Equation (3.1) then becomes, in vector notation:

$$\mathbf{o}_S = \mathbf{w}^T\mathbf{S}$$
$$\mathbf{o}_N = \mathbf{w}^T\mathbf{N},$$

where each element of the row vectors $\mathbf{o}_S$ and $\mathbf{o}_N$ is a filtered sample of $\mathbf{S}$ and $\mathbf{N}$, respectively. Figure 3.1 (right) shows the distribution of the values in two example data vectors $\mathbf{o}_S$ and $\mathbf{o}_N$.

We want to find the weight vector $\widehat{\mathbf{w}}$ that maximises the variance of $\mathbf{o}_S$ versus the variance of $\mathbf{o}_N$, i.e.

$$\widehat{\mathbf{w}} = \arg\max_{\mathbf{w}} \frac{\mathrm{Var}(\mathbf{o}_S)}{\mathrm{Var}(\mathbf{o}_N)} \tag{3.2}$$

$$= \arg\max_{\mathbf{w}} \frac{\frac{1}{N_S}\mathbf{o}_S\mathbf{o}_S^T}{\frac{1}{N_N}\mathbf{o}_N\mathbf{o}_N^T}$$

$$= \arg\max_{\mathbf{w}} \frac{\frac{1}{N_S}\mathbf{w}^T\mathbf{S}\mathbf{S}^T\mathbf{w}}{\frac{1}{N_N}\mathbf{w}^T\mathbf{N}\mathbf{N}^T\mathbf{w}}$$

In this last equation, we recognize the empirical covariance matrices $\mathbf{R}_{SS}$ and $\mathbf{R}_{NN}$, which are defined as:

$$\mathbf{R}_{SS} = \frac{1}{N_S}\mathbf{S}\mathbf{S}^T \tag{3.3}$$

$$\mathbf{R}_{NN} = \frac{1}{N_N}\mathbf{N}\mathbf{N}^T \tag{3.4}$$

$\mathbf{R}_{SS} \in \mathbb{R}^{C \times C}$ and $\mathbf{R}_{NN} \in \mathbb{R}^{C \times C}$ are symmetric matrices, where each diagonal element yields the variance of a channel, and each off-diagonal element yields the covariance between a pair of channels.

The condition for the optimal weight vector, eq. (3.2), is thus equivalent to:

$$\widehat{\mathbf{w}} = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T\mathbf{R}_{SS}\mathbf{w}}{\mathbf{w}^T\mathbf{R}_{NN}\mathbf{w}} \tag{3.5}$$

In appendix C, we show that the solution $\widehat{\mathbf{w}}$ to this optimisation problem is the first so called "generalized eigenvector" of $(\mathbf{R}_{SS}, \mathbf{R}_{NN})$.

The generalized eigenproblem

An arbitrarily scaled vector $\mathbf{w}_i$ is a so called *generalized eigenvector* (GEVec) for the ordered matrix pair $(\mathbf{R}_{SS}, \mathbf{R}_{NN})$ when the following holds:

$$\mathbf{R}_{SS}\mathbf{w}_i = \lambda_i\mathbf{R}_{NN}\mathbf{w}_i, \tag{3.6}$$

for some scalar λ_i, which is called the *generalized eigenvalue* (GEVal) corresponding to $\mathbf{w}_i$. The largest scalar λ_1 for which eq. (3.6) holds is the 'first' GEVal, and as mentioned before, the corresponding GEVec $\mathbf{w}_1$ is the solution $\widehat{\mathbf{w}}$ to eq. (3.5).

Since the 1960's, numerically stable algorithms exist that solve the generalized eigen-problem eq. (3.6) [3]. A specialized algorithm is applicable when the input matrices are symmetric – as is the case for $\mathbf{R}_{SS}$ and $\mathbf{R}_{NN}$. This algorithm (based on a Cholesky factorization and the classical QR-algorithm for ordinary eigenproblems) is implemented in the LAPACK software package (as `ssygv` and `dsygv`), and can be easily applied using e.g. the `eig` function from MATLAB, or the `eigh` function from SciPy's `linalg` module.

3.3   Result for SWR detection

We divided the 34-minute long LFP recording into two datasets. The first 60% was used as training data, to calculate the covariance matrices $\mathbf{R}_{SS}$ and $\mathbf{R}_{NN}$, and to calculate from these the optimal linear combination of channels $\widehat{\mathbf{w}}$, as described in the preceding sections. The remaining 40% was used to evaluate this filter $\widehat{\mathbf{w}}$, and to compare it to the state-of-the-art method (the single-channel online band-pass filter).[2]

Figure 3.2A shows an excerpt of the test input signal, and the corresponding filter output envelopes (blue for state-of-the art method, orange for GEVec-based multichannel method. Filter outputs o_t are rectified to obtain envelopes $n_t = |o_t|$). Additional excerpts are shown in fig. D.1. The elements of $\widehat{\mathbf{w}}$ (i.e. the filter weights) are visualized in fig. 3.2B.

It is clear that the GEVec filter output indeed has high power during SWR events, as promised by the theoretical derivation. The filter weights and signal excerpts reveal that the GEVec output is composed mainly of a few channels in the stratum radiatum (channels 4-5-6 here), where they pick up the sharp waves. However, the filter output envelope is also high for sharp wave-like activity on these channels, without or with only very weak ripple activity in the higher channels: see figs. D.1a, D.1b and D.1d. This results in false positive detections.

We also notice some premature detections (figs. D.1a and D.1c), where the sharp wave – and thus also the GEVec filter output – already has high power before the corresponding ripple has started. The GEVec detection then happens before the start of the reference SWR-segment, which is based on ripple power. These early detections thus count (arguably unfairly so) as false positives, under the evaluation scheme that we use.

This effect, where the sharp wave is discernible before the ripple, results in faster detections when the detection *does* fall within the reference segment. When all 468 SWR events from the test set are analysed, we find a large improvement in detection latency: at thresholds where both methods detect 80% of these reference SWR segments, the median absolute detection latency drops from 24 ms for the state-of-the-art online band-pass filter to 12 ms for the GEVec-based filter. The relative detection latency drops by 32.5 percentage points, from 58.1% to 25.6%. Similar latency improvements are found for other recall and precision values: see fig. 3.3.

This strong improvement in detection latency trades off with an increase in false positives, as was already observed qualitatively. At the aforementioned sensitivity of 80%, the

---

[2]The 60-40 division was chosen arbitrarily – under the constraint that there are sufficient amounts of both training and test data.
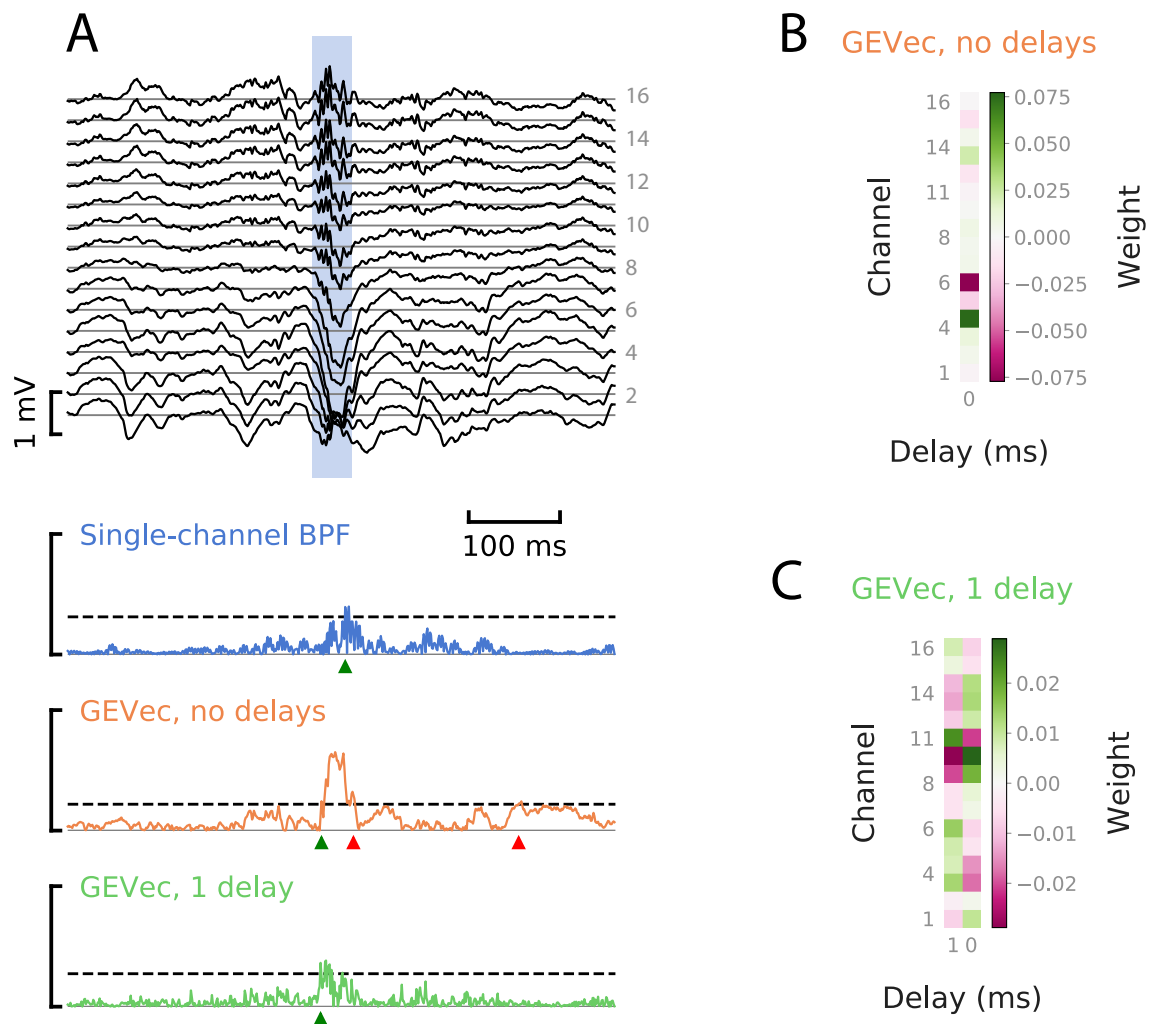
Figure 3.2: **Linear, SNR-maximising combinations of electrodes**.
A. Example input and output signals. *Top*: multi-channel LFP, $\mathbf{z}_t$. Light-blue vertical band: a reference SWR segment. *Bottom*: output envelopes n_t, for different filtering algorithms. Dashed horizontal lines: detection thresholds, chosen so that each algorithm reaches a recall value of 80%. Green triangles: correct detections. Red triangles: incorrect detections. Brackets indicate envelope range (min, max) over the entire test set.
B. Generalized eigenvector $\widehat{\mathbf{w}}$ (i.e. the weights of the multichannel filter), for a purely spatial filter.
C. Generalized eigenvector $\widehat{\mathbf{w}}$ for a spatiotemporal filter with a one-sample delay.
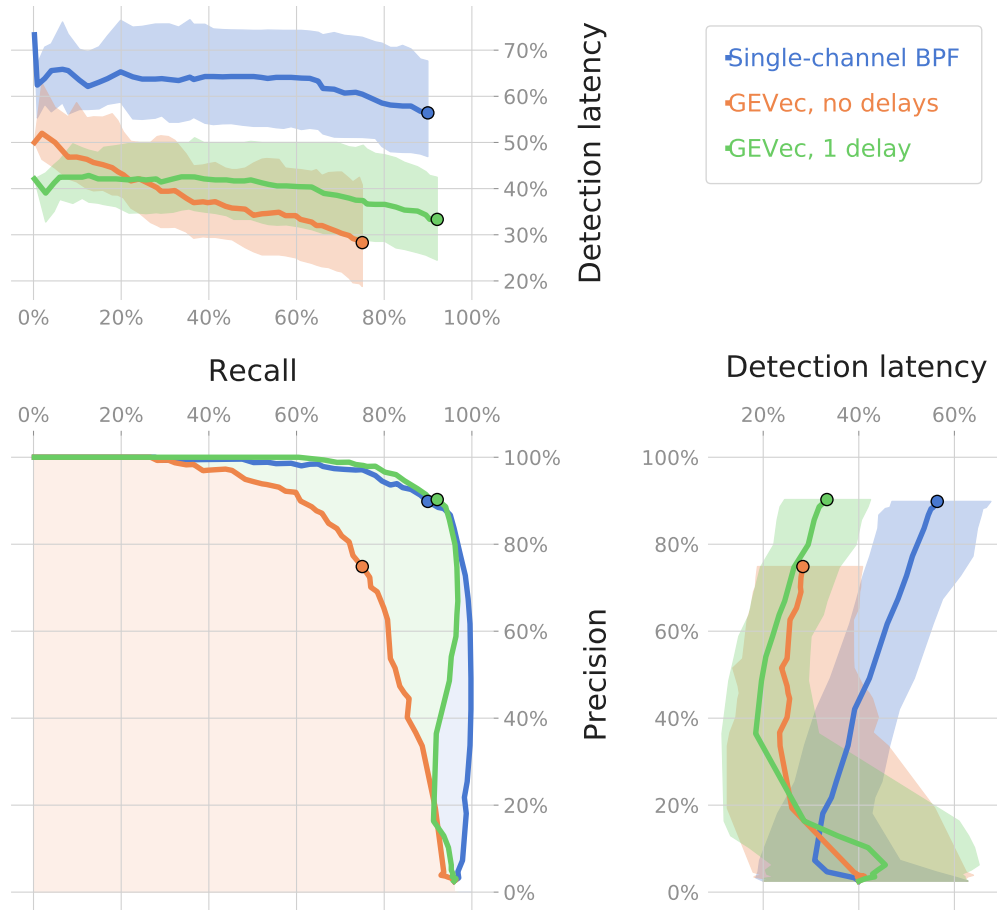
Figure 3.3: **Sensitivity, precision & latency tradeoffs,** for different linear filters & for a range of thresholds.

Each threshold setting for an algorithm corresponds to a point on the precision-recall curve in the bottom-left panel, and to a distribution of relative detection latencies. The median and interquartile range of this distribution are plotted in the top-left or bottom-right panel (as a point of the bold curve, and a slice of the shaded band, respectively).

These latency distribution plots are divided over two panels so that their entire range can be clearly visualized, both for the low recall – high precision regime as for the high recall – low precision regime. The cutoff is made at the point where recall equals precision (AKA the $\max F_1$ point), marked with shaded black circles.

state-of-the-art method has a precision of 94%, whereas the GEVec-based method has a precision of only 63% (i.e. more than a third of detected events are classified as false positives). This strong decrease in precision is true over the entire PR-curve: see fig. 3.3.

3.4 Combining space and time [1p]

It is hardly surprising that the GEVec-based algorithm as described above cannot discern ripple activity (which is by definition a temporal pattern), as the algorithm is a purely *spatial* filter: at each timestep t, only current information from the different channels is used in calculating the output o_t, without incorporating temporal information from previous timesteps $t_p < t$.

The GEVec method can be easily adapted to also incorporate temporal information however, by defining a vector $\mathbf{z}_t^{\text{stack}} \in \mathbb{R}^{CP}$ which consists of stacked sample vectors (each consisting of C channels) from P different timesteps $t_p \leq t$. The linear weights $\mathbf{w}^{\text{stack}} \in \mathbb{R}^{CP}$ used to obtain the output signal $o_t = (\mathbf{w}^{\text{stack}})^T \mathbf{z}_t^{\text{stack}}$ are then calculated analogously to the purely spatial filter, i.e. as the first generalized eigenvector of the ordered pair $(\mathbf{R}_{SS}^{\text{stack}}, \mathbf{R}_{NN}^{\text{stack}})$, with both covariance matrices $\in \mathbb{R}^{CP \times CP}$.

Adding just one such delayed time step (i.e. $P = 2$) yields a major performance improvement (see fig. 3.3): the precision-recall curve shoots up to (and even slightly exceeds) the PR-curve of the state-of-the-art algorithm, while the latency improvements of the 'no delay' GEVec algorithm are mostly retained: at a sensitivity of 80%, the median absolute latency for the one delay GEVec filter is 15 ms, which is 9 ms faster than the state-of-the-art method (and 3 ms slower than the no delay GEVec filter). The relative latency is 36.6%: 21.5 percentage-points lower than the state-of-the-art (and 11 pp. higher than the no delay GEVec filter).

At this 80% recall mark, the one delay GEVec filter attains a precision of 97%, a 3% increase over the state-of-the-art. The full precision-recall-latency tradeoff and algorithm comparison is shown in fig. 3.3. Note that for very low thresholds, the PR-curve of the one-delay GEVec method is no longer concave: decreasing the threshold further yields more (not less) missed reference SWR segments. Figure 3.2C shows the GEVec $\mathbf{w}^{\text{stack}}$. Figure 3.2A and fig. D.1 show example output envelopes (green traces).

These results, particularly the visualized weights in fig. 3.2C, indicate that this one-delay GEVec filter utilizes spatiotemporal information about both the sharp wave and the ripple.

Choosing the number of delays

Adding more delays (fig. 3.4) improves detection accuracy even further – up to a peak *max F_1* of 93% at about eleven delays. (This corresponds to eleven milliseconds, or approximately half a ripple phase). Detection latency also increases with increasing number of delays, but only slightly, always staying well below the state-of-the-art latency. Like the *max F_1* score, latency stagnates after about eleven delays. Further, we note that the latency distributions of the GEVec-based detectors have a lower spread than those of the state-of-the-art detector.
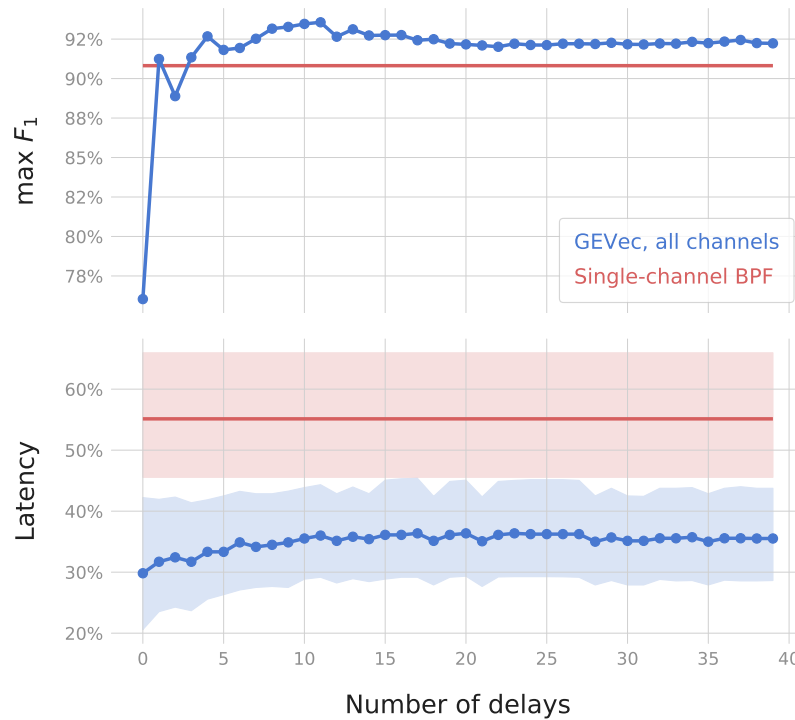
Figure 3.4: **Performance of the GEVec-based SWR detector, for different delay line lengths**. At the chosen 1000 Hz sampling rate, each delay corresponds to 1 ms. The red baseline is the state-of-the-art SWR detector. Detection latency is specified as a fraction of the duration of the corresponding SWR event, and is evaluated at the threshold where each detector reaches its maximum F_1-score. In the latency panel, bold lines and shaded areas indicate the median and the interquartile range of the latency distributions, respectively.

There is thus a slight tradeoff to be made when choosing the number of delays for a GEVec-based detector: using more delays yields detectors that are more accurate, but also slightly slower. Given that the decrease in speed is minor (about five percentage-points), it is reasonable to choose the amount of delays that maximizes detection accuracy. In this analysis, this optimal point is reached at an eleven milliseconds-long delay line.

Selecting channels

Many CA1 LFP recordings are not made with multichannel probes, but rather with one or more tetrodes. It is therefore relevant to ask how GEVec-based detectors perform on these types of recordings. We can approximate this setting with our current multichannel probe recording, by only using one or a few channels.

Each such selection of input channels, in combination with a certain number of delays, yields a different GEVec-based SWR detector. We evaluate and compare these detectors using the test data set, analyzing both accuracy (fig. 3.5) and latency (fig. 3.6).

We note some general trends. Using no extra delays results in low accuracy detectors, no matter which channels are included (top row of fig. 3.5). For most input channel
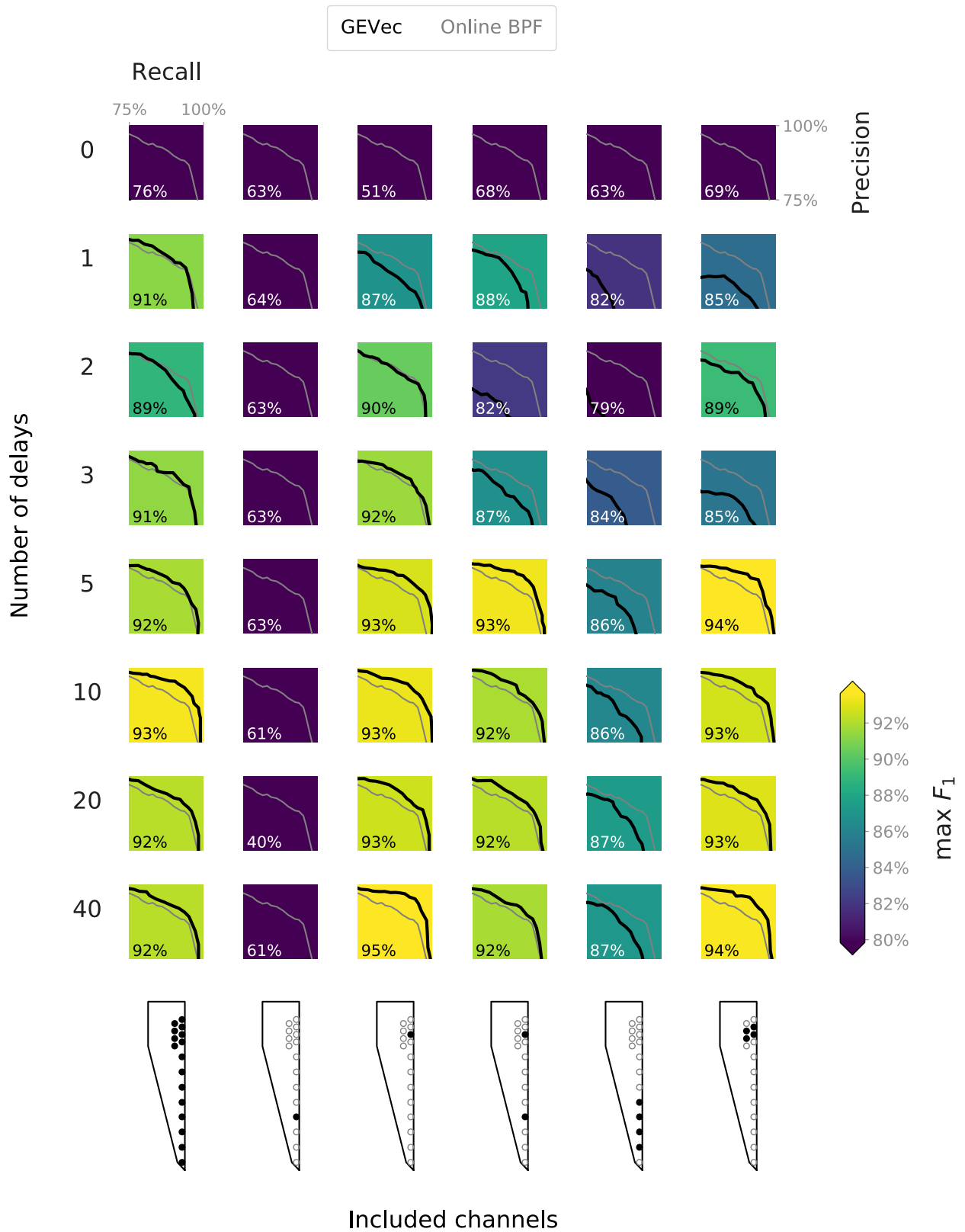
Figure 3.5: **Accuracy of GEVec-based SWR detectors,** for different combinations of active input channels and number of delays. Active input channels are marked in solid black on a schematic of the probe tip. (Height of schematic: 687 μm). The percentage and background color of each panel indicate the maximum F_1 value obtained for that GEVec-based SWR detector. Each panel includes the same baseline from the state-of-the-art SWR detector, in gray.

Figure 3.6: **Latency of GEVec-based SWR detectors**. See fig. 3.5 for legend.

combinations, the tradeoff in number of delays observed in the previous section is preserved: using more delays increases accuracy (up to a certain point), while also increasing detection latency.

The weight visualizations of fig. 3.2 show a relatively high weight for the stratum radiatum / sharp wave channels. Using only one such channel as input results in low accuracy detectors however, no matter how many delays are used (second column of figs. 3.5 and 3.6). Using four such channels (second to last column) results in detectors with an accuracy approaching, but not reaching, that of the state-of-the art ripple-based detector when a sufficient number of delays is used. Interestingly, this decent accuracy is reached despite none of these four channels (numbers 2 to 5) displaying any noticeable ripple activity (fig. 3.2A).

Conversely, using one channel from the pyramidal cell layer (where ripples are observed) yields GEVec-based detectors on par with those that use all channels as input, both regarding accuracy and latency (third and first column of figs. 3.5 and 3.6).

This one channel GEVec-based detector thus also outperforms the "state-of-the-art" detector based on a band-pass filter. The state-of-the-art detector also uses only one channel as input, and is also a linear filter. This suggests that either 1) finding linear filter weights through 'machine learning' (the GEVec technique) yields better results than manual filter design; or 2) that the band-pass filter used as "state-of-the-art" detector was not designed optimally in the first place.

Adding a stratum radiatum / sharp wave channel to the single pyramidal cell layer channel (fourth column of figs. 3.5 and 3.6), or using a cluster of channels in the pyramidal cell layer (last column) does not improve performance over the one channel case.

3.5 Regularization [1p]

3.6 Multiple eigenvectors [2p]

3.7 Spectral preprocessing [1p]

Nonlinear signal detection − 9 Nov [9p]

Chapter *5*

Discussion − 23 Nov [4p]

5.1 Comparing detectors [2p]

5.2 Further work [2p]

Chapter 6

Conclusions – 30 Nov [1p]

References

[1] C. J. Van Rijsbergen. *Information Retrieval*. 2nd edition. Newton, MA, USA: Butterworth-Heinemann, 1979. URL: http://www.dcs.gla.ac.uk/Keith/Preface.html.

[2] György Buzsáki. "Hippocampal Sharp Wave-ripple: A Cognitive Biomarker for Episodic Memory and Planning". In: *Hippocampus* 25.10 (Sept. 26, 2015), pp. 1073–1188. ISSN: 1050-9631. DOI: 10.1002/hipo.22488. URL: http://onlinelibrary.wiley.com/doi/full/10.1002/hipo.22488 (visited on 04/23/2018).

[3] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Fourth edition. The Johns Hopkins University Press, 2013.

[4] Zoltán Nádasdy, Hajime Hirase, András Czurkó, Jozsef Csicsvari, and György Buzsáki. "Replay and Time Compression of Recurring Spike Sequences in the Hippocampus". In: *Journal of Neuroscience* 19.21 (Nov. 1, 1999), pp. 9497–9507. ISSN: 0270-6474, 1529-2401. DOI: 10.1523/JNEUROSCI.19-21-09497.1999. pmid: 10531452. URL: http://www.jneurosci.org/content/19/21/9497 (visited on 08/02/2018).

[5] Jozsef Csicsvari, Hajime Hirase, András Czurkó, Akira Mamiya, and György Buzsáki. "Oscillatory Coupling of Hippocampal Pyramidal Cells and Interneurons in the Behaving Rat". In: *Journal of Neuroscience* 19.1 (Jan. 1, 1999), pp. 274–287. ISSN: 0270-6474, 1529-2401. DOI: 10.1523/JNEUROSCI.19-01-00274.1999. pmid: 9870957. URL: http://www.jneurosci.org/content/19/1/274 (visited on 11/13/2018).

[6] Jozsef Csicsvari, Hajime Hirase, Akira Mamiya, and György Buzsáki. "Ensemble Patterns of Hippocampal CA3-CA1 Neurons during Sharp Wave–Associated Population Events". In: *Neuron* 28.2 (Nov. 1, 2000), pp. 585–594. ISSN: 0896-6273. DOI: 10.1016/S0896-6273(00)00135-5. URL: http://www.sciencedirect.com/science/article/pii/S0896627300001355 (visited on 08/01/2018).

[7] Jozsef Csicsvari, Hajime Hirase, András Czurkó, Akira Mamiya, and György Buzsáki. "Fast Network Oscillations in the Hippocampal CA1 Region of the Behaving Rat". In: *The Journal of Neuroscience* 19.16 (Aug. 15, 1999), RC20–RC20. ISSN: 0270-6474, 1529-2401. DOI: 10.1523/JNEUROSCI.19-16-j0001.1999. URL: http://www.jneurosci.org/lookup/doi/10.1523/JNEUROSCI.19-16-j0001.1999 (visited on 11/13/2018).

[8] Christoph J. Behrens, Leander P. van den Boom, Livia de Hoz, Alon Friedman, and Uwe Heinemann. "Induction of Sharp Wave–Ripple Complexes *in Vitro* and Reorganization of Hippocampal Networks". In: *Nature Neuroscience* 8.11 (Nov. 2005), pp. 1560–1567. ISSN: 1546-1726. DOI: 10.1038/nn1571. URL: https://www.nature.com/articles/nn1571 (visited on 07/31/2018).

[9] Josef H. L. P. Sadowski, Matthew W. Jones, and Jack R. Mellor. "Sharp-Wave Ripples Orchestrate the Induction of Synaptic Plasticity during Reactivation of Place Cell Firing Patterns in the Hippocampus". In: *Cell Reports* 14.8 (Mar. 1, 2016), pp. 1916–1929. ISSN: 2211-1247. DOI: 10.1016/j.celrep.2016.01.061. URL: http://www.sciencedirect.com/science/article/pii/S2211124716300390 (visited on 06/17/2018).

[10] Shayok Dutta, Etienne Ackermann, and Caleb Kemere. "Analysis of an Open Source, Closed-Loop, Realtime System for Hippocampal Sharp-Wave Ripple Disruption". In: *bioRxiv* (Apr. 11, 2018). DOI: 10.1101/298661. URL: https://www.biorxiv.org/content/early/2018/04/11/298661 (visited on 05/30/2018).

[11] Sen Cheng and Loren M. Frank. "New Experiences Enhance Coordinated Neural Activity in the Hippocampus". In: *Neuron* 57.2 (Jan. 24, 2008), pp. 303–313. ISSN: 0896-6273. DOI: 10.1016/j.neuron.2007.11.035. pmid: 18215626. URL: https://www.cell.com/neuron/abstract/S0896-6273(07)01030-6 (visited on 11/13/2018).

[12] Caleb Kemere, Margaret F. Carr, Mattias P. Karlsson, and Loren M. Frank. "Rapid and Continuous Modulation of Hippocampal Network State during Exploration of New Places". In: *PLOS ONE* 8.9 (2-Sep-2013), e73114. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0073114. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0073114 (visited on 11/13/2018).

[13] Lloyd N. Trefethen and David III Bau. *Numerical Linear Algebra*. Siam, 1997.

Appendices

SWR detection in the literature

We gathered a representative sample of research papers that use automated offline sharp wave-ripple detection. This appendix quotes these papers; specifically their description of the SWR detection procedure. Emphasis is added.

Nádasdy et al. 1999

"For the extraction of sharp-wave (SPW) ripple events during sleep, the wide-band recorded data were bandpass filtered digitally (**150–250 Hz**). The power (**root mean square**) of the filtered signal was calculated, and the beginning, peak, and end of individual ripple episodes were determined. The threshold for ripple detection was set to **7 SDs above the background mean** power (Csicsvari et al., 1999 [5])." [4]

Csicsvari et al. 2000

"*Detection of SPW-Associated Fast Ripples*: The procedures described here were identical to those described earlier (Csicsvari et al., 1999b [7]). The wide-band (1–5 kHz) recorded data was digitally band-pass filtered (**80–250 Hz**), and the power (**root-mean-square**) of the filtered signal was calculated for each electrode. The mean and standard deviation (SD) of the power signal were calculated to determine the detection threshold. Oscillatory epochs with a power of **one** or more **SD above the mean** were detected. The beginning and the end of oscillatory epochs were marked at points where the power fell below **0.5 SD**. Theta periods, detected by using the theta-delta power ratio (Csicsvari et al., 1999a [5]), were excluded from the analysis." [6]

Behrens et al. 2005

"For ripple detection, raw data were filtered with a Spike 2 software band-pass filter of **40–400 Hz** (threshold: **4–6** times the s.d. of **eventless baseline noise**). For sharp wave detection, recordings were low-pass filtered at 20 Hz." [8]

Sadowski et al. 2016

"Ripples were detected offline in the LFP recorded on one CA1 channel. Raw LFP signal was filtered between **120 and 250 Hz**, and deflections in the ripple **power envelope**

greater than **5 SDs from the mean** were classified as ripple events. Ripple start times were defined locally as when ripple power exceeded **2 SDs**. Samples of raw LFP and detected ripple times were compared manually to verify detection fidelity." [9]

Dutta et al. 2018

"Post-recording, ripple events were defined on tetrodes that displayed characteristics of the CA1 area of the hippocampus. Specifically, the recorded LFP in one of the channels of the selected tetrode (same one subject to online detection for our realtime analysis) first had a digital reference subtracted away. This signal was then LFP band filtered with a 400 Hz low-pass infinite impulse response (IIR) filter (from Trodes). Afterwards the signal was decimated and ripple band filtered (**150–250 Hz**) with a **25 tap** finite impulse response (**FIR**) filter. Ripple band filtering was done using a forward and a time-reversed path, resulting in a net **zero group delay** (time shift from filtering). The instantaneous power of the ripple band filtered signal was then calculated via a **Hilbert Transform** and further **smoothened with a Gaussian kernel** with a **4 ms standard deviation**. Ripple events were detected as times when z-score of the smoothened power signal signal exceeded a threshold of **3 z-units** for **at least 15 ms**. The canonical ripple epochs were defined as the time points from which the processed signal **returned down to the mean before and after threshold crossings** [11], [12].

In cases when multiple electrodes (typically channels on different tetrodes) are available for ripple detection, a different canonical definition is required. Ripples were initially defined using as above for each electrode. A canonical multichannel ripple was defined as one which is simultaneously detected on each electrode (two in our analysis). The multichannel ripple epoch is defined as the union of the detected single-channel ripple epochs, i.e., the start of the earliest ripple detected and to end with bound of last ripple detected. As such, we obtain a conservative ripple detection latency estimate while covering the entire span of the time the LFP is in a high ripple band power state. We reanalyzed our data with the canonical ripples being defined on different channels and tetrodes with a 300 tap bandpass FIR filter allowing 1% "ripple" in the passband with 30 dB suppression in the stopband but our results and subsequent conclusions remained consistent." [10]

Appendix B

Data description

Generalized eigenvectors maximize signal-to-noise

In this appendix, we show that the weight vector that maximizes the ratio of signal to noise variance (i.e. the solution $\widehat{\mathbf{w}}$ to eq. (3.5)) is equivalent to the first generalized eigenvector $\mathbf{w}_1$ of the ordered pair of covariance matrices $(\mathbf{R}_{SS}, \mathbf{R}_{NN})$ (as defined in section 3.2).

This ratio of variances in eq. (3.5) is a quotient of quadratic forms, namely the so called "generalized Rayleigh quotient" of $(\mathbf{R}_{SS}, \mathbf{R}_{NN})$.

Formally, the generalized Rayleigh quotient of a non-zero vector $\mathbf{w} \in \mathbb{R}^N$ and the ordered, symmetric matrix pair $(\mathbf{A}, \mathbf{B})$ is the scalar $r(\mathbf{w})$ defined as:

$$r(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{A} \mathbf{w}}{\mathbf{w}^T \mathbf{B} \mathbf{w}} \tag{C.1}$$

We must then prove the following:

Theorem

The generalized eigenvector $\mathbf{w}_1$ corresponding to the largest generalized eigenvalue λ_1 of $(\mathbf{A}, \mathbf{B})$, is also the vector $\widehat{\mathbf{w}}$ that maximises the generalized Rayleigh quotient $r(\mathbf{w})$ of $(\mathbf{A}, \mathbf{B})$.

Proof

As a first step, we will show that if $\widehat{\mathbf{w}}$ is the maximum of $r(\mathbf{w})$, that it is indeed an eigenvector of $(\mathbf{A}, \mathbf{B})$. In the second step, we will show that the largest eigenvalue λ_1 of $(\mathbf{A}, \mathbf{B})$ corresponds to the maximum of $r(\mathbf{w})$.

If $\widehat{\mathbf{w}}$ is a maximum of $r(\mathbf{w})$, then

$$\boldsymbol{\nabla} r(\widehat{\mathbf{w}}) = \mathbf{0}. \tag{C.2}$$

Working out the partial derivatives that comprise the gradient of $r(\mathbf{w})$, we find:

$$\nabla r(\mathbf{w}) = \frac{2\mathbf{A}\mathbf{w}(\mathbf{w}^T\mathbf{B}\mathbf{w}) - 2\mathbf{B}\mathbf{w}(\mathbf{w}^T\mathbf{A}\mathbf{w})}{(\mathbf{w}^T\mathbf{B}\mathbf{w})^2}$$

With eq. (C.2), we then have the following condition for our maximising vector $\widehat{\mathbf{w}}$:

$$2\mathbf{A}\widehat{\mathbf{w}}(\widehat{\mathbf{w}}^T\mathbf{B}\widehat{\mathbf{w}}) = 2\mathbf{B}\widehat{\mathbf{w}}(\widehat{\mathbf{w}}^T\mathbf{A}\widehat{\mathbf{w}})$$

or

$$\mathbf{A}\widehat{\mathbf{w}} = \frac{\widehat{\mathbf{w}}^T\mathbf{A}\widehat{\mathbf{w}}}{\widehat{\mathbf{w}}^T\mathbf{B}\widehat{\mathbf{w}}} \mathbf{B}\widehat{\mathbf{w}}$$

$$\mathbf{A}\widehat{\mathbf{w}} = r(\widehat{\mathbf{w}}) \mathbf{B}\widehat{\mathbf{w}}$$

This is the generalized eigenvalue/eigenvector definition (eq. (3.6)) for $\mathbf{w}_i = \widehat{\mathbf{w}}$ and $\lambda_i = r(\widehat{\mathbf{w}})$.

We have thus shown that if $\widehat{\mathbf{w}}$ is a maximum of $r(\mathbf{w})$, that it is an eigenvector of $(\mathbf{A}, \mathbf{B})$, with $r(\widehat{\mathbf{w}})$ its corresponding eigenvalue.

As the second step, we now show that $r(\widehat{\mathbf{w}})$ is the *largest* eigenvalue of $(\mathbf{A}, \mathbf{B})$. We follow the reasoning of Trefethen and Bau, who prove a related result for the ordinary Rayleigh quotient [13, p. 204].

We will rewrite the generalized Rayleigh quotient $r(\mathbf{w})$ by writing the arbitrary vector $\mathbf{w}$ as a linear combination of the generalized eigenvectors $\mathbf{w}_i$ of $(\mathbf{A}, \mathbf{B})$: $\mathbf{w} = \sum_i c_i \mathbf{w}_i$. Then:

$$r(\mathbf{w}) = \frac{(\sum_i c_i \mathbf{w}_i)^T \mathbf{A} (\sum_i c_i \mathbf{w}_i)}{(\sum_i c_i \mathbf{w}_i)^T \mathbf{B} (\sum_i c_i \mathbf{w}_i)}$$

$$= \frac{\sum_i c_i^2 \mathbf{w}_i^T \mathbf{A} \mathbf{w}_i}{\sum_i c_i^2 \mathbf{w}_i^T \mathbf{B} \mathbf{w}_i}$$

$$= \frac{\sum_i c_i^2 \lambda_i \mathbf{w}_i^T \mathbf{B} \mathbf{w}_i}{\sum_i c_i^2 \mathbf{w}_i^T \mathbf{B} \mathbf{w}_i}.$$

Generalized eigenvectors are defined up to a scaling factor. We may therefore define our $\mathbf{w}_i$ to be scaled such that $\mathbf{w}_i^T \mathbf{B} \mathbf{w}_i = 1$. We then have:

$$r(\mathbf{w}) = \frac{\sum_i c_i^2 \lambda_i}{\sum_i c_i^2}.$$

Each generalized Rayleigh quotient is thus a convex combination of generalized eigenvalues λ_i. The maximum of a convex combination of one-dimensional points is obtained in the largest of these points. If λ_1 is thus the largest generalized eigenvalue of $(\mathbf{A}, \mathbf{B})$, then $\max r(\mathbf{w}) = \lambda_1$.

We have thus shown that $\arg\max r(\mathbf{w}) = \mathbf{w}_1$, where $\mathbf{w}_1$ is an eigenvector of $(\mathbf{A}, \mathbf{B})$, and that its corresponding eigenvalue $\lambda_1 = \max r(\mathbf{w})$ is the largest of the eigenvalues of $(\mathbf{A}, \mathbf{B})$.
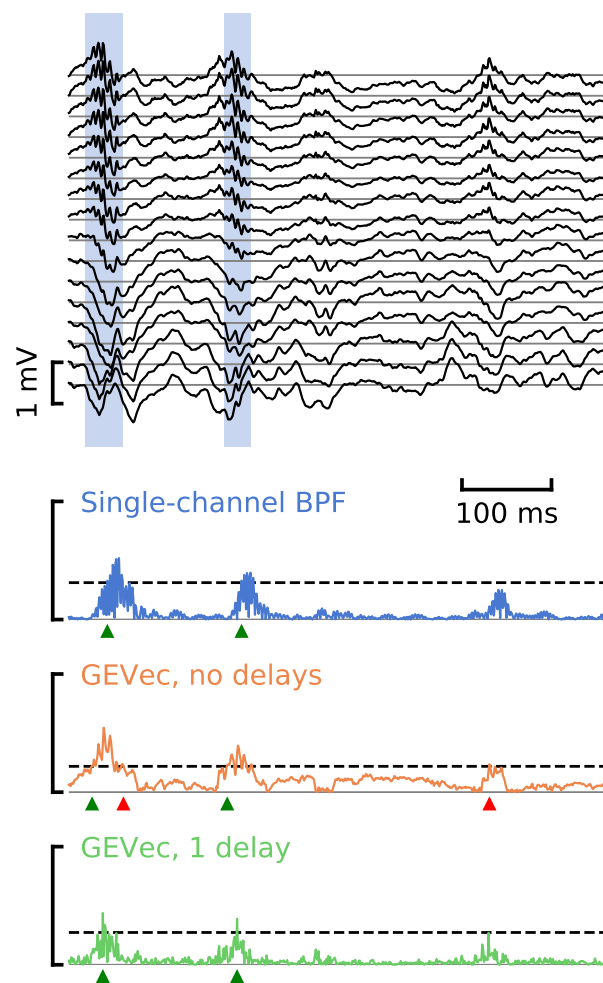
$\square$

Appendix D
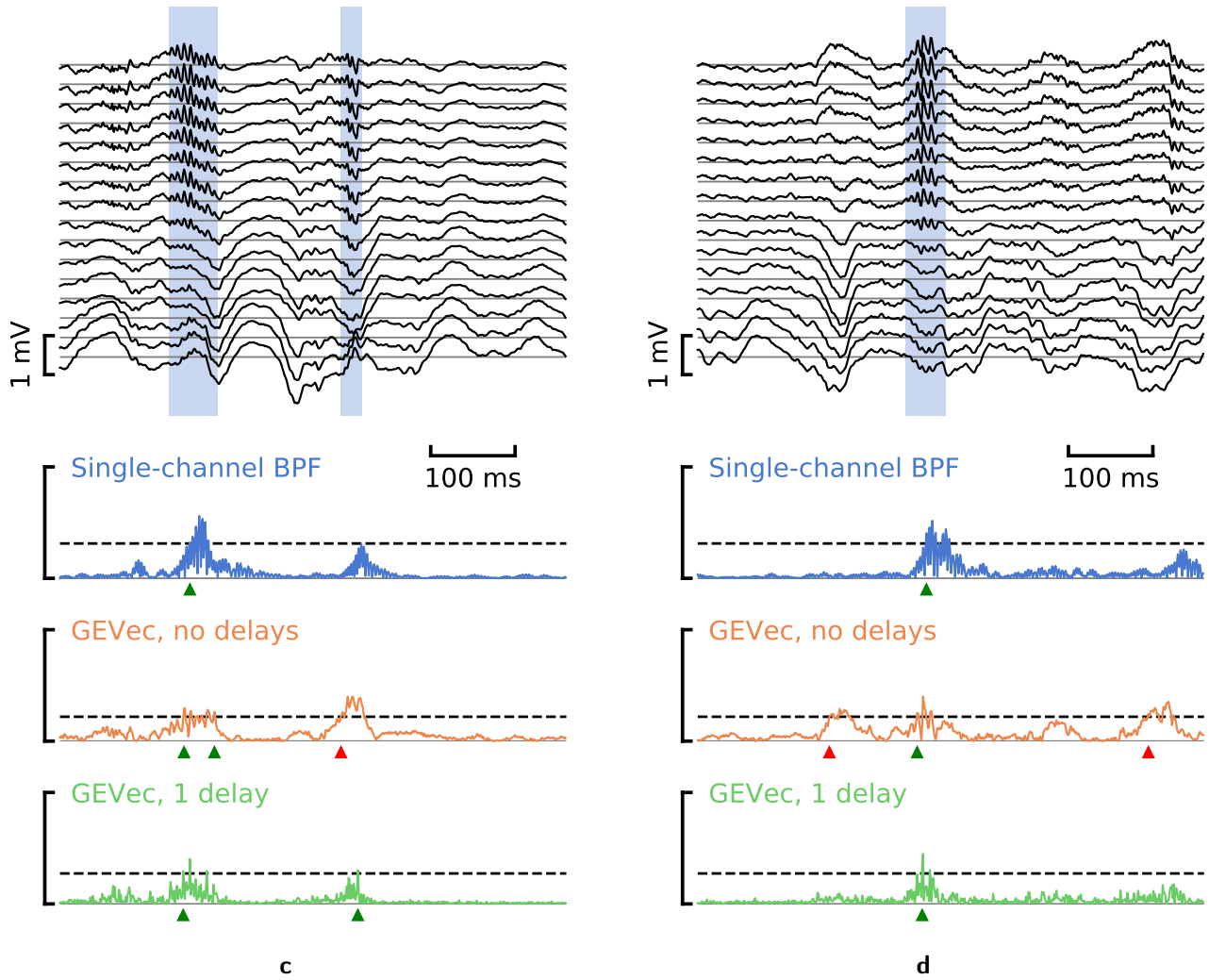
Supplemental figures
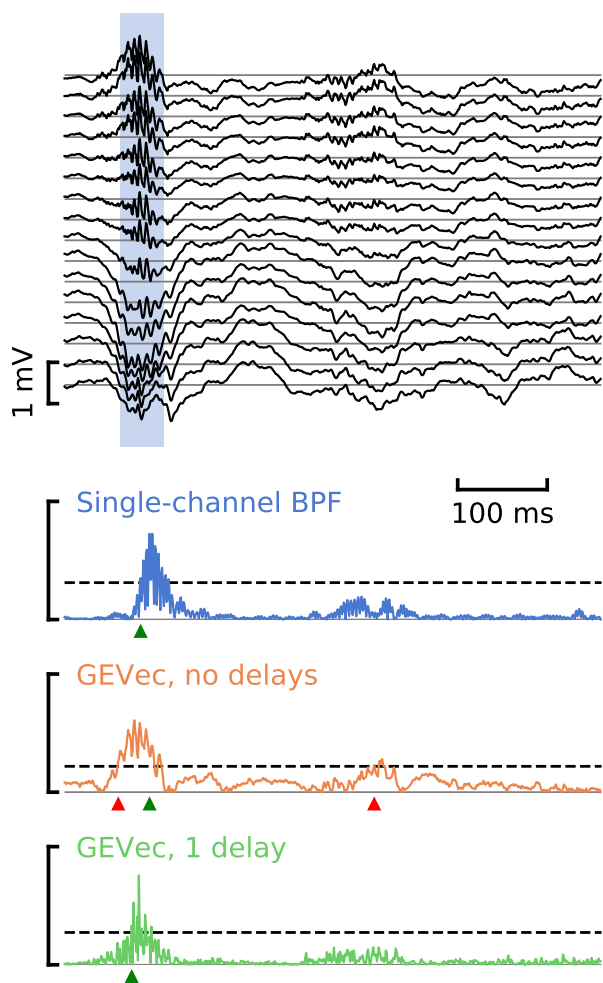
Figure D.1: **Extracts from input data and corresponding linear filter output envelopes**. See fig. 3.2 for legend.