

# Real-time detection of voltage patterns in the brain

Tomas Fiers

Thesis submitted for the degree of  
Master of Science in  
Biomedical Engineering

**Thesis supervisors:**

Prof. dr. ir. A. Bertrand  
Prof. dr. F. Kloosterman

**Assessors:**

Prof. dr. ir. R. Puers  
Dr. eng. J. Couto

**Mentors:**

Ir. J. Wouters  
Dott. D. Ciliberti

© Copyright KU Leuven

Without written permission of the thesis supervisors and the author it is forbidden to reproduce or adapt in any form or by any means any part of this publication. Requests for obtaining the right to reproduce or utilize parts of this publication should be addressed to Faculteit Ingenieurswetenschappen, Kasteelpark Arenberg 1 bus 2200, B-3001 Heverlee, +32-16-321350.

A written permission of the thesis supervisors is also required to use the methods, products, schematics and programmes described in this work for industrial or commercial use, and for submitting this publication in scientific contests.

---

## Abstract – 30 Nov

---

---

# Contents

---

<b>Symbols</b>	<b>6</b>
<b>Abbreviations</b>	<b>7</b>
<b>Glossary</b>	<b>8</b>
<b>1 Introduction – 9 Nov [9p]</b>	<b>9</b>
1.1 Closed-loop brain-computer interfaces [1p] . . . . .	9
1.2 Sharp wave-ripples [6p] . . . . .	9
1.3 Problem statement [1p] . . . . .	9
1.4 Thesis overview [1p] . . . . .	9
<b>2 Single-channel linear filtering – 19 Oct [10p]</b>	<b>10</b>
2.1 Offline labelling of SWR segments [3p] . . . . .	10
2.2 State of the art online SWR detectors [3p] . . . . .	10
2.3 Quantifying & comparing detectors [4p] . . . . .	10
<b>3 Multi-channel linear filtering – 12 Oct [10p]</b>	<b>11</b>
3.1 Data-driven algorithms [1p] . . . . .	11
3.2 Linear signal-to-noise maximisation [2p] . . . . .	11
3.3 Combining space and time [1p] . . . . .	14
3.4 Regularization [1p] . . . . .	14
3.5 Channels [1p] . . . . .	14
3.6 Taps [1p] . . . . .	14
3.7 Multiple eigenvectors [2p] . . . . .	14
3.8 Spectral preprocessing [1p] . . . . .	14
<b>4 Nonlinear signal detection – 2 Nov [9p]</b>	<b>15</b>
4.1 Recurrent neural networks [2p] . . . . .	15
4.2 Optimization [3p] . . . . .	15
4.3 Regularization [2p] . . . . .	15
4.4 Channels [1p] . . . . .	15
4.5 Network size [1p] . . . . .	15
<b>5 Discussion – 16 Nov [4p]</b>	<b>16</b>
5.1 Comparing detectors [2p] . . . . .	16
5.2 Further work [2p] . . . . .	16

<i>CONTENTS</i>	5
<b>6 Conclusions – 30 Nov [1p]</b>	<b>17</b>
<b>References</b>	<b>18</b>
<b>Appendices</b>	<b>19</b>

---

# 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.
- $\odot$  Elementwise multiplication. (“Hadamard product”).
- $\sigma(\cdot)$  Sigmoid ‘squashing’ function.  $\sigma(x) = \frac{1}{1+\exp(-x)}$ ,  $\in (0, 1)$ .
- $\tanh(\cdot)$  Hyperbolic tangent.  $\tanh(x) = 2 \sigma(x) - 1$ ,  $\in (-1, 1)$ .

## 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,  $\in \mathbb{R}$ .
- $p_t$  Transformation of  $o_t$ , 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.  $p_t = |o_t|$  for online linear filters;  $p_t = \sigma(o_t)$  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.

---

# Abbreviations

---

<b>CA1</b>	Cornu ammonis, region 1. Region in the hippocampus where voltages are recorded from (see ???).
<b>CA3</b>	Cornu ammonis, region 3. Region in the hippocampus (see ???). CA3 sends many axons (called “Schafer collaterals”) to CA1.
<b>GEVec</b>	generalized eigenvector. See <a href="#">section 3.2</a> .
<b>GEVal</b>	generalized eigenvalue. See <a href="#">section 3.2</a> .
<b>LFP</b>	Local field potential. The extracellular electric potential (see ??).
<b>RNN</b>	Recurrent neural network. See <a href="#">chapter 4</a> .
<b>SWR</b>	Sharp wave-ripple. The pattern in the LFP that we want to detect in real-time. See ??.

---

# Glossary

---

<b>Apical</b>	The 'top' side of a pyramidal neuron. In the neocortex: the side near the skull. In CA1: the 'stratum radiatum' side (see ??).
<b>Basal</b>	The 'bottom' side of a pyramidal neuron. In the neocortex: the side towards the center of the brain. In CA1: the 'stratum oriens' side (see ??).
<b>Tetrode</b>	Four thin electrode wires ( $< 30\mu\text{m}$ in diameter), bundled together.
<b>Neurite</b>	Either an axon or a dendrite.
<b>Schaffer collateral</b>	Axon originating in CA3 and terminating in CA1.
<b>Soma</b>	(plural: somata). Cell body of a neuron.



---

## Introduction – 9 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]**

## *Chapter 2*

---

# Single-channel linear filtering – 19 Oct [10p]

---

- 2.1 Offline labelling of SWR segments [3p]
- 2.2 State of the art online SWR detectors [3p]
- 2.3 Quantifying & comparing detectors [4p]

---

## Multi-channel linear filtering – 12 Oct [10p]

---

### 3.1 Data-driven algorithms [1p]

In this and the following chapter, we describe *supervised*, or *data-driven* detection algorithms: they require training data  $\mathbf{z}_t^{\text{train}}$ , and an associated labelling  $y_t^{\text{train}}$  which marks when there is an event present in  $\mathbf{z}_t^{\text{train}}$ . 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 a reference target labelling  $y_t$  for some recording data  $\mathbf{z}_t$  is the same as marking SWR segments for offline analysis or for offline evaluation of an online detection algorithm. As described in [section 2.1](#), this can be done either by human expert labellers, or by using an automated *offline* SWR detection algorithm, where we assume that the automated labelling would correspond well to a human expert labelling. In this thesis, we use the automated labelling method of [section 2.1](#) to generate target labellings  $y_t^{\text{train}}$ .

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  $y_t^{\text{train}}$  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 3.4](#) 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

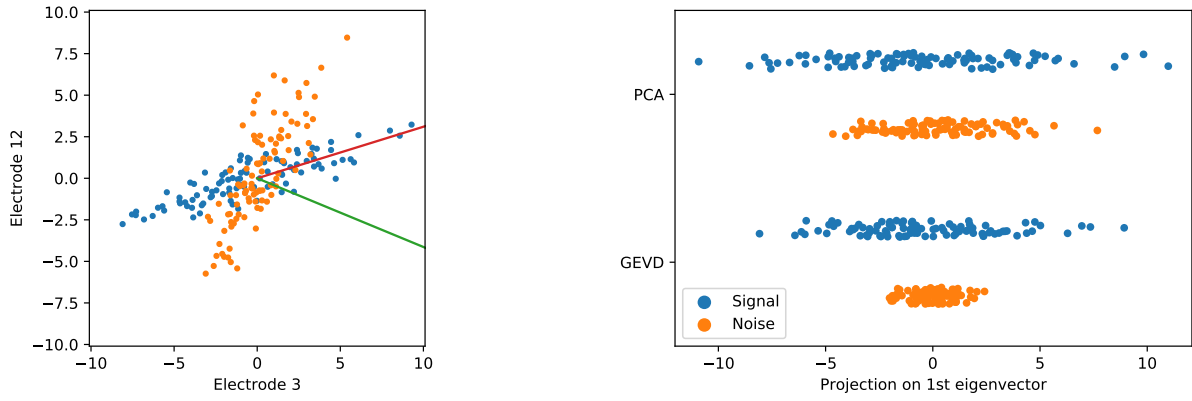


Figure 3.1: **Linear signal-to-noise maximisation.** Toy example to illustrate the generalized eigenvector approach to signal detection. *Left*: multi-channel time-series data plotted in ‘phase space’ (meaning without time axis), with blue dots representing samples where the signal was present, and orange 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 (“GEVD”). The ratio of the projected signal data variance versus the projected noise data variance is maximised for the GEVD 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

$$o_t = \mathbf{w}^T \mathbf{z}_t \quad (3.1)$$

has high variance (or power) during SWR events, and low variance outside them. 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 remainder of this section describes how the vector  $\mathbf{w}$  can be found.

### The optimization 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 all segments in  $\mathbf{z}_t^{\text{train}}$  where an SWR is present, and all segments where there is no SWR present).

[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  $\mathbf{o}_N$  corresponds to one (multichannel) sample of  $\mathbf{S}$  or  $\mathbf{N}$ . [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  $\hat{\mathbf{w}}$  that maximises the variance of  $\mathbf{o}_S$  versus the variance of  $\mathbf{o}_N$ , i.e.

$$\begin{aligned}\hat{\mathbf{w}} &= \arg \max_{\mathbf{w}} \frac{\text{Var}(\mathbf{o}_S)}{\text{Var}(\mathbf{o}_N)} \\ &= \arg \max_{\mathbf{w}} \frac{\frac{1}{n_S} \mathbf{o}_S \mathbf{o}_S^T}{\frac{1}{n_N} \mathbf{o}_N \mathbf{o}_N^T} \\ &= \arg \max_{\mathbf{w}} \frac{\frac{1}{n_S} \mathbf{w}^T \mathbf{S} \mathbf{S}^T \mathbf{w}}{\frac{1}{n_N} \mathbf{w}^T \mathbf{N} \mathbf{N}^T \mathbf{w}}\end{aligned}\tag{3.2}$$

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:

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

In [appendix A](#), we show that the solution  $\hat{\mathbf{w}}$  to this optimisation problem is the first so called “generalized eigenvector” of  $(\mathbf{R}_{SS}, \mathbf{R}_{NN})$ .

### The generalized eigenvalue problem

The generalized eigenvalue problem for the ordered matrix pair  $(\mathbf{R}_{SS}, \mathbf{R}_{NN})$  consists of finding the scalars  $\lambda_i$  and corresponding vectors  $\mathbf{w}_i$  for which the following holds:

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

where  $\lambda_i$  is a so called *generalized eigenvalue* (GEVal) and  $\mathbf{w}_i$  is the corresponding arbitrarily scaled *generalized eigenvector* (GEVec). The largest scalar  $\lambda_1$  for which [eq. \(3.6\)](#) holds is the ‘first’ GEVal, and the corresponding GEVec  $\mathbf{w}_1$  is the solution  $\hat{\mathbf{w}}$  to [eq. \(3.5\)](#).

If  $\mathbf{R}_{NN}$  is of full rank and thus invertible, [eq. \(3.6\)](#) is equivalent to

$$\mathbf{R}_{NN}^{-1} \mathbf{R}_{SS} \mathbf{w}_i = \lambda_i \mathbf{w}_i$$

The generalised eigenvectors and eigenvalues of  $(\mathbf{R}_{SS}, \mathbf{R}_{NN})$  are thus the ordinary eigenvectors and eigenvalues of  $\mathbf{R}_{NN}^{-1} \mathbf{R}_{SS}$ . While mathematically exact, this is a poor method to calculate GEVecs in practice: when  $\mathbf{R}_{NN}$  is ill-conditioned, small changes in  $\mathbf{R}_{NN}$  (resulting from quantization and other sources of ‘noise’) will then lead to disproportionately large errors in  $\mathbf{R}_{NN}^{-1}$ , and thus also in the the calculated GEVecs of  $(\mathbf{R}_{SS}, \mathbf{R}_{NN})$  [1], [2]. In practice,  $\mathbf{R}_{NN}$  is indeed likely to be ill-conditioned:

### 3.3 Combining space and time [1p]

### 3.4 Regularization [1p]

### 3.5 Channels [1p]

### 3.6 Taps [1p]

### 3.7 Multiple eigenvectors [2p]

### 3.8 Spectral preprocessing [1p]

## Nonlinear signal detection – 2

### Nov [9p]

---

- 4.1 Recurrent neural networks [2p]
- 4.2 Optimization [3p]
- 4.3 Regularization [2p]
- 4.4 Channels [1p]
- 4.5 Network size [1p]

---

## Discussion – 16 Nov [4p]

---

5.1 Comparing detectors [2p]

5.2 Further work [2p]



---

## Conclusions – 30 Nov [1p]

---

---

# References

---

- [1] Lloyd N. Trefethen and David III Bau. *Numerical Linear Algebra*. Siam, 1997.
- [2] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Fourth. The Johns Hopkins University Press, 2013.

---

# Appendices

---

A	Generalized eigenvectors maximise signal-to-noise
---	---

20
----

*Appendix A*

---

# Generalized eigenvectors maximise signal-to-noise

---