# Neural Spikes Classification in Multichannel Recordings

Martina Camarda

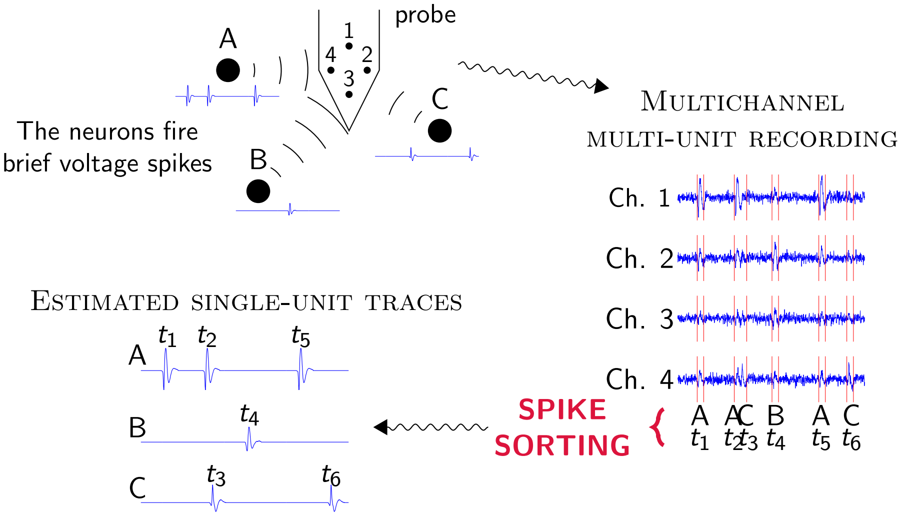
Department of Information Engineering
University of Padova

October 20, 2009

# Outline

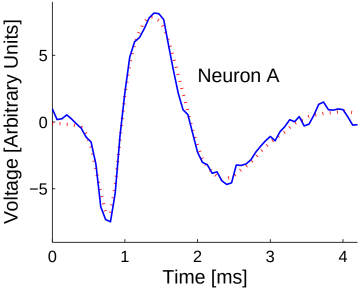
- The spike sorting problem in Neurophysiology
  Purposes, signal features, and difficulties.
- A new algorithm: MCI4SC
  Assumption, main ideas and implementation.
- Analysis of experimental data
  Results and comparisons.
- Conclusions

# The Spike Sorting Problem



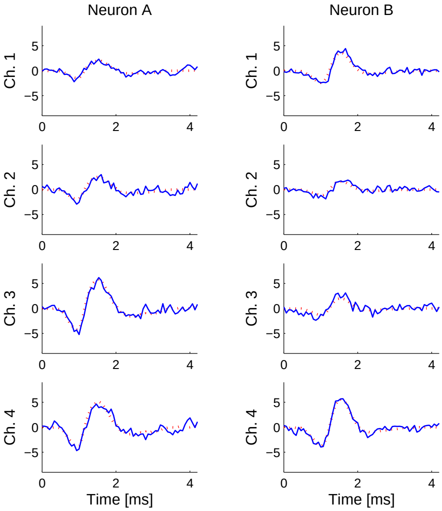
# Signal Features: The Spike Waveform

In general, spikes from the **same neuron** have similar waveforms and spikes from **different neurons** have different waveforms.



# Signal Features: The Neuron Position



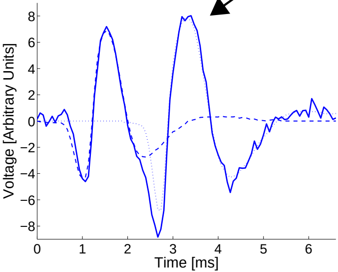


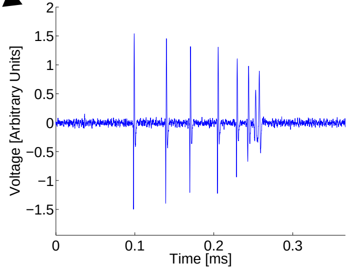

The *neuron position* (with respect to electrodes) is related to the *Amplitude Ratios* among channels.

In general, spikes from the **same neuron** have similar amplitude ratios and spikes from **different neurons** have different amplitude ratios.

# Main Difficulties

- Unknown number of **involved neurons**
- **Noise** in the recorded data
- **Electrode drifts** during the recording
- **Bursting neurons** generate spikes varying in amplitude and shape
- Presence of **Overlapping Spikes**



# Spike sorting methods in literature

Many spike sorting algorithms have been developed,
but there is still no consensus on which is the best method.

- Amplitude and Window Discriminator
- Template Matching
- Principal Components Analysis
- Methods based on Wavelet Transformations
- Multichannel methods
- Methods for overlapping spikes resolution

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

**MCI4SC: Multi-Channel Inversion For Spike Classification**

$\underbrace{\text{For Spike Classification}}$

assigns each spike
to its own neuron of origin

# The New Algorithm: MCI4SC

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

**MCI4SC: $\underbrace{\text{Multi-Channel}}$ Inversion $\underbrace{\text{For Spike Classification}}$**

assigns each spike
to its own neuron of origin

exploits the
amplitude ratios information
[*Wavelet Packet Transform*]

# The New Algorithm: MCI4SC

**MCI4SC: $\underbrace{\text{Multi-Channel}}$ $\underbrace{\text{Inversion}}$ $\underbrace{\text{For Spike Classification}}$**

assigns each spike
to its own neuron of origin

exploits the
amplitude ratios information
[*Wavelet Packet Transform*]

inverts the mixing matrix
associated to
the measurement channel
[*Moore-Penrose Pseudoinverse*]

# Assumptions about Recordings



- The neural signal arrives with the **same delay** on each sensor.
- Each channel records the **same spike waveform**.
- The intracellular **medium is linear**:

$$\underbrace{Y^{(m\times K)}}_{\text{noisy measurements}} = \underbrace{\tilde{A}^{(m\times n)}}_{\text{mixing matrix}} \underbrace{\tilde{S}^{(n\times K)}}_{\text{unknown signals}} + \underbrace{H^{(m\times K)}}_{\text{additive noise}}$$

- **The probe holds steady** during all the recording time.
- Additive, Gaussian, and stationary noise, independent of the signal.

# Assumptions about Neural Activity



- Each neuron has a **different position** with respect to electrodes.
- Each neuron fires a **finite set of spike waveforms.**
- Each neuron **fires more than one time**.
- **No more than $m$ neurons fire simultaneously**, where $m$ is the number of employed sensors.

# The Mixing Matrix

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Mixing Matrix
$\tilde{A}^{(m \times n)}$

$m \longrightarrow$ number of sensors
$n \longrightarrow$ number of neurons

represents the **measurement channel** between neuron and sensors

typical situation:
more recorded neurons than recording sensors
$m < n$

# The Mixing Matrix

Mixing Matrix
$\tilde{A}^{(m \times n)}$

$m \longrightarrow$ number of sensors
$n \longrightarrow$ number of neurons

represents the **measurement channel** between neuron and sensors

$$
\begin{array}{cc}
 & n \text{ columns} \\
m \text{ rows} & \left[ \quad \tilde{A} \quad \right]
\end{array}
$$

$\boxed{\text{RECTANGULAR :-(}}$

typical situation:
more recorded neurons than recording sensors
$m < n$

# MCI4SC: Two Phases



Mixing Matrix
$\tilde{A}^{(m \times n)}$

**A - The Learning Phase:**

- estimates the number of neuron $n$
- estimates the waveform templates
- estimates the mixing matrix $\tilde{A}$

**B - The Classification Phase:**

- inverts many matrices derived by $\tilde{A}$
- analyzes each detected spikes to determine the corresponding firing neurons and the arrival time

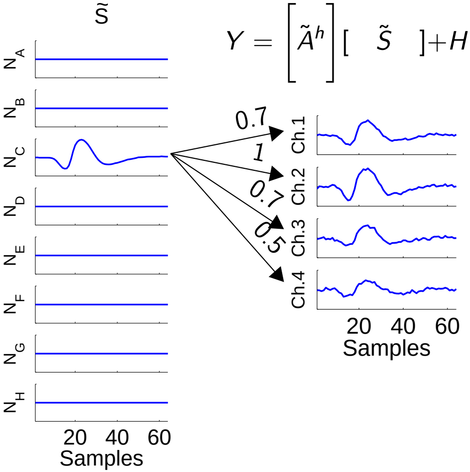
# The Learning Phase: Main Ideas

Crucial point:

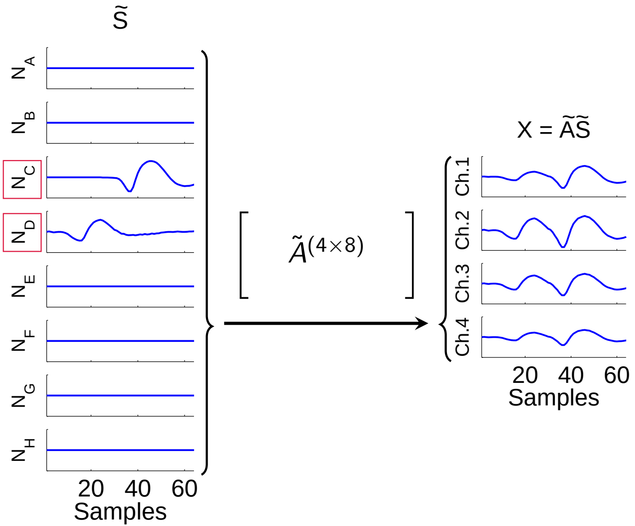
**sparsity** and **finite duration** of neural spikes allow to isolate intervals in which the data contain **the signal coming from just one neuron $h$**, while other neural signals are equal to zero.

This allows to **directly estimate** the normalized mixing matrix, column by column: the amplitude ratios correspond to the matrix components.

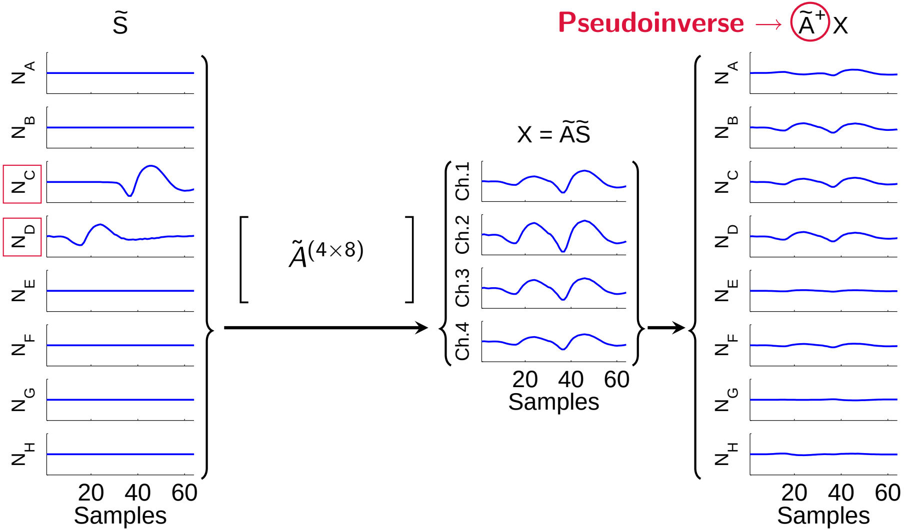
$$Y = \left[\tilde{A}^h\right][\quad \tilde{S} \quad] + H$$



# The Classification Phase: Main Ideas



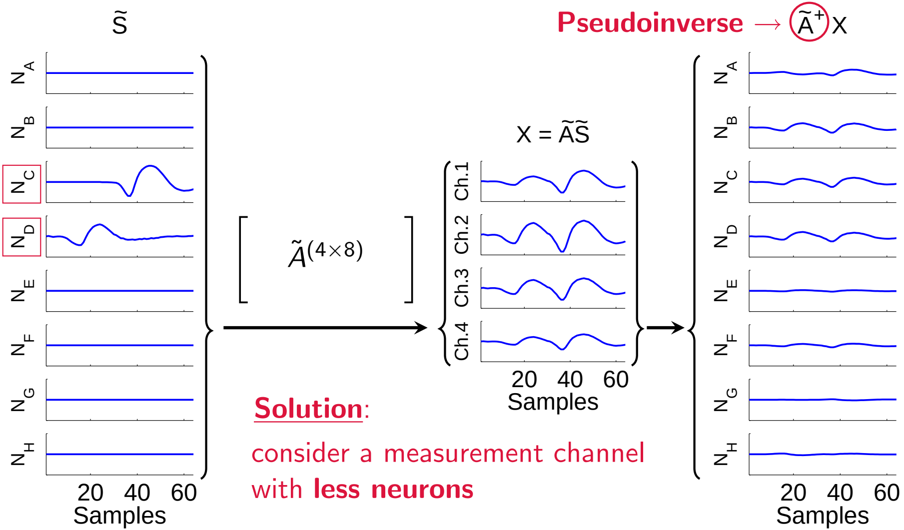
# The Classification Phase: Main Ideas



# The Classification Phase: Main Ideas



**Pseudoinverse** $\longrightarrow \tilde{A}^{+}X$




**Solution**:

consider a measurement channel with **less neurons**

# The Matrix $C$

Since all neurons are not simultaneously active
it is possible to consider an equivalent measurement model,
different from time interval to time interval,
that disregards some mixing matrix columns
associated to neurons actually inactive:

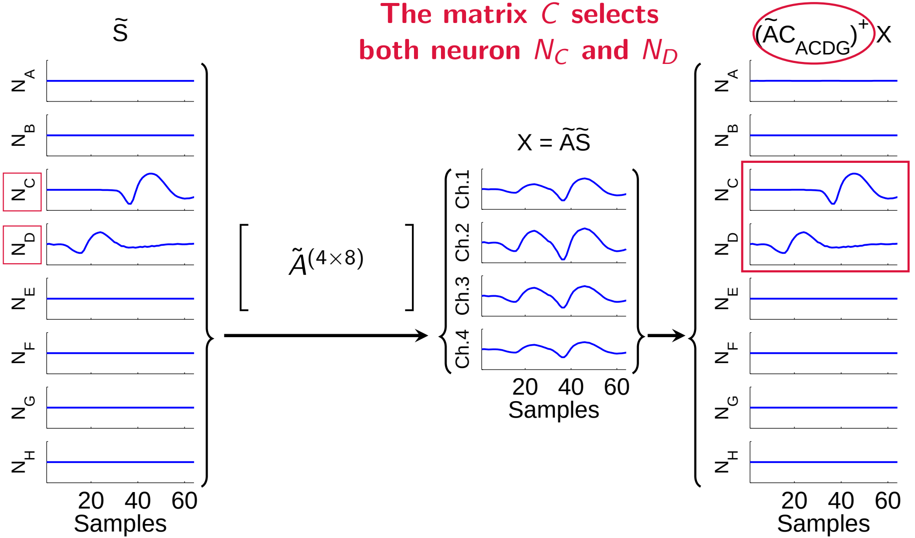
$$\boxed{Y = \tilde{A} C \tilde{S} + H}$$

$$C^{n \times n} = \begin{bmatrix} c_1 & & & & \mathbf{0} \\ & \ddots & & & \\ & & c_j & & \\ & & & \ddots & \\ \mathbf{0} & & & & c_n \end{bmatrix} \qquad c_j = \begin{cases} 1 & \text{if neuron } j \text{ is selected} \\ 0 & \text{if neuron } j \text{ is unselected} \end{cases}$$

# The Classification Phase: Main Ideas

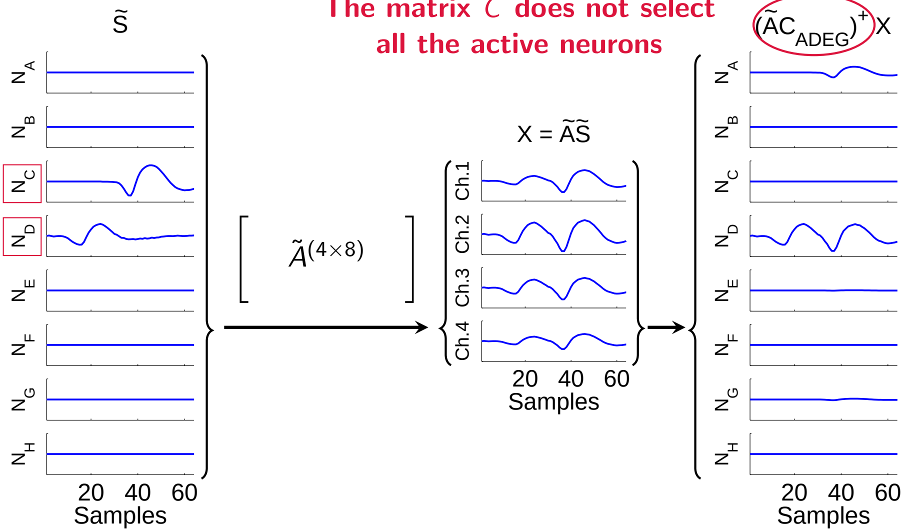
DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

**The matrix $C$ selects both neuron $N_C$ and $N_D$**



# The Classification Phase: Main Ideas

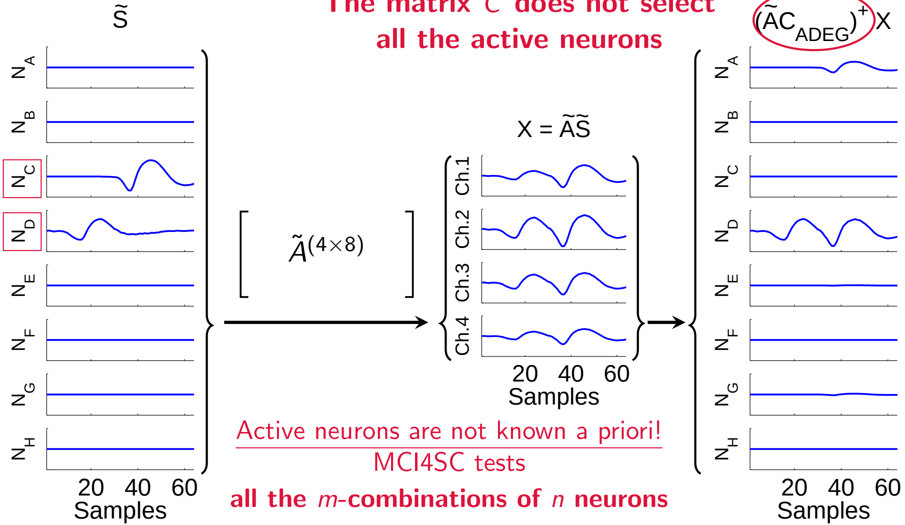
**The matrix $C$ does not select all the active neurons**



# The Classification Phase: Main Ideas



**The matrix $C$ does not select all the active neurons**




Active neurons are not known a priori!

MCI4SC tests

**all the $m$-combinations of $n$ neurons**

# MCI4SC Steps

A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.
A2. Estimation of spike-amplitude ratios for each single spike.
A3. Single spikes sorting.
A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^\dagger$ for each $C$.
B2. Estimation of the hypothetical signal $\hat{S}^C$ for each $C$.
B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^C$ re-mixed by the recording matrix $\tilde{A}$.
B4. Spike classification.

# MCI4SC Steps

A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.
A2. Estimation of spike-amplitude ratios for each single spike.
A3. Single spikes sorting.
A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^{\dagger}$ for each $C$.
B2. Estimation of the hypothetical signal $\hat{S}^C$ for each $C$.
B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^C$ re-mixed by the recording matrix $\tilde{A}$.
B4. Spike classification.

# MCI4SC Steps



A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.

A2. Estimation of spike-amplitude ratios for each single spike.

A3. Single spikes sorting.

A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^{\dagger}$ for each $C$.

B2. Estimation of the hypothetical signal $\hat{S}^C$ for each $C$.

B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^C$ re-mixed by the recording matrix $\tilde{A}$.

B4. Spike classification.

# MCI4SC Steps

A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.

A2. Estimation of spike-amplitude ratios for each single spike.

A3. Single spikes sorting.

A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

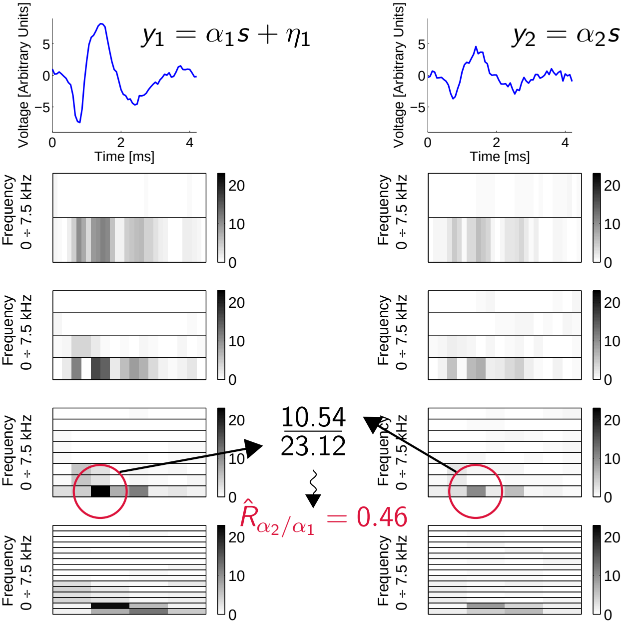
B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^{\dagger}$ for each $C$.

B2. Estimation of the hypothetical signal $\hat{S}^{C}$ for each $C$.

B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^{C}$ re-mixed by the recording matrix $\tilde{A}$.

B4. Spike classification.

# Amplitude Ratios Estimation




For each single spike, $m-1$ **amplitude ratios** are estimated with respect to a reference channel.

# MCI4SC Steps

A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.

A2. Estimation of spike-amplitude ratios for each single spike.

A3. Single spikes sorting.

A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^{\dagger}$ for each $C$.

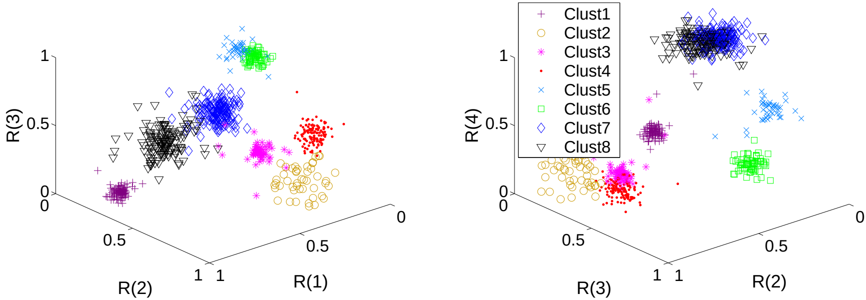
B2. Estimation of the hypothetical signal $\hat{S}^C$ for each $C$.

B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^C$ re-mixed by the recording matrix $\tilde{A}$.

B4. Spike classification.

# Single Spike Sorting

The $m - 1$ amplitudes ratios (related to neuron position) are used as *distinctive features* to cluster single spikes.




As a result, every single spike is associated to its neuron of origin.

# MCI4SC Steps



A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.

A2. Estimation of spike-amplitude ratios for each single spike.

A3. Single spikes sorting.

A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

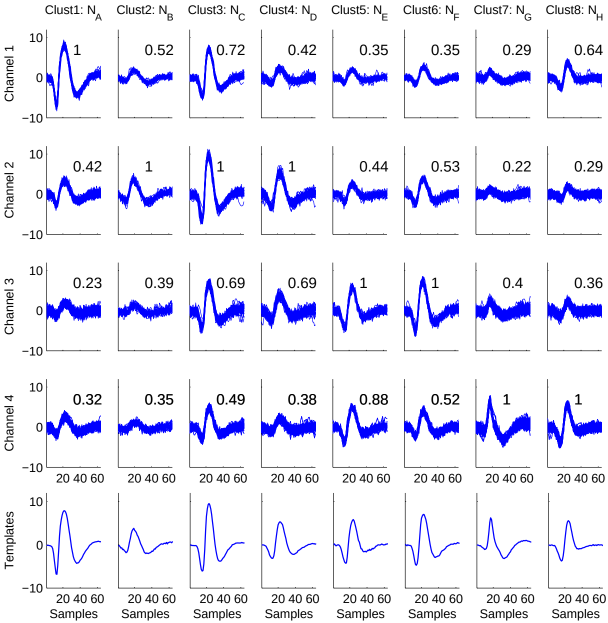
B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^\dagger$ for each $C$.

B2. Estimation of the hypothetical signal $\hat{S}^C$ for each $C$.

B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^C$ re-mixed by the recording matrix $\tilde{A}$.

B4. Spike classification.

# Waveforms & Mixing Matrix Estimation




$\Longrightarrow$ mean amplitude ratios

$\Longrightarrow$ mean waveforms

# MCI4SC Steps



A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.

A2. Estimation of spike-amplitude ratios for each single spike.

A3. Single spikes sorting.

A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

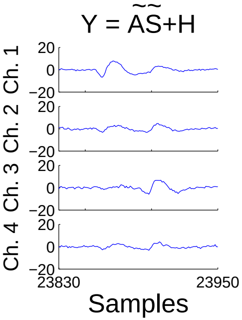
B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^{\dagger}$ for each $C$.

B2. Estimation of the hypothetical signal $\hat{S}^C$ for each $C$.

B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^C$ re-mixed by the recording matrix $\tilde{A}$.

B4. Spike classification.

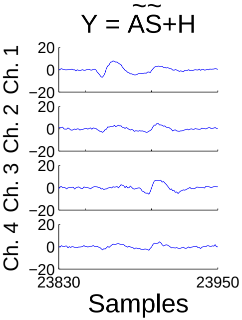
# Computation/Application of the Pseudoinverses



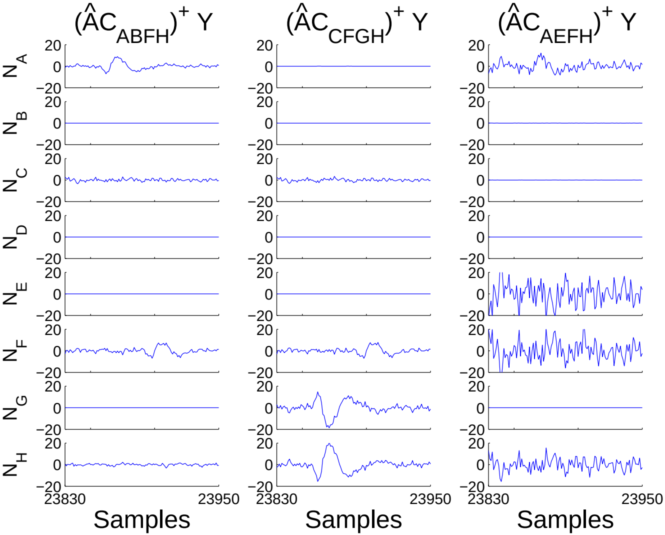

There are 70 4-combinations of 8 neurons

# Computation/Application of the Pseudoinverses

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA




There are 70 4-combinations of 8 neurons



# MCI4SC Steps



A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

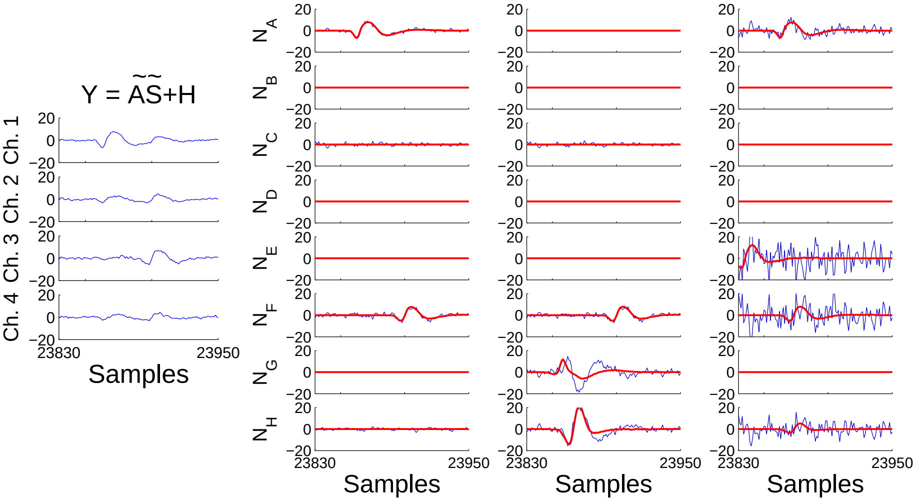
A1. Rejection of Overlapping spikes.
A2. Estimation of spike-amplitude ratios for each single spike.
A3. Single spikes sorting.
A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^\dagger$ for each $C$.
B2. Estimation of the hypothetical signal $\hat{S}^C$ for each $C$.
B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^C$ re-mixed by the recording matrix $\tilde{A}$.
B4. Spike classification.

# Estimation of the Hypothetical Signal

In each trace there is either noise or one spike



# MCI4SC Steps

A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.

A2. Estimation of spike-amplitude ratios for each single spike.

A3. Single spikes sorting.

A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^\dagger$ for each $C$.

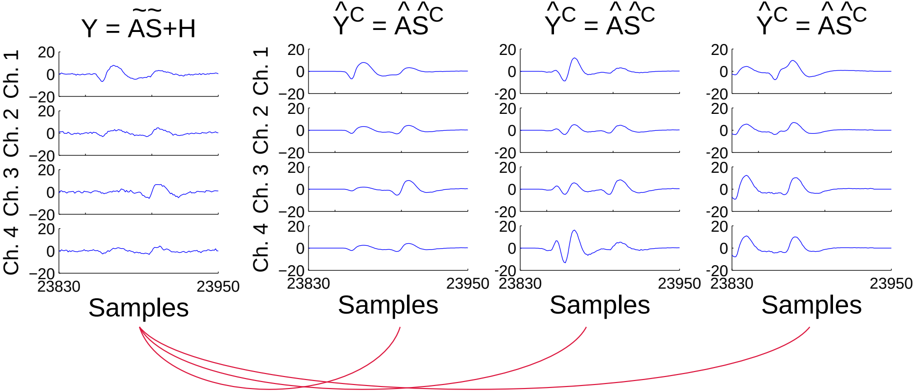
B2. Estimation of the hypothetical signal $\hat{S}^C$ for each $C$.

B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^C$ re-mixed by the recording matrix $\tilde{A}$.

B4. Spike classification.

# Selection of $C$ with Minimum Distance

The hypothetical signals $\hat{S}^C$ are remixed by $\hat{A}$

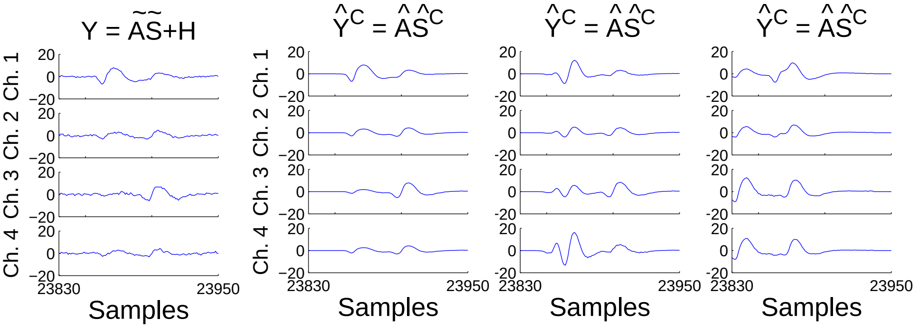



$$d_F(Y, \hat{Y}^C) = ||\Sigma^{-1}(Y - \hat{Y}^C)||_F$$

It normalizes different noise levels in the channels

# Selection of $C$ with Minimum Distance

The hypothetical signals $\hat{S}^C$ are remixed by $\hat{A}$




Minimum Distance between hypothetical and original measurements

It best explains the measured data, both in terms of active neurons, and in terms of conditioning.

# MCI4SC Steps

A0. Preliminary steps: filtering, spike detection and window selection.

THE LEARNING PHASE

A1. Rejection of Overlapping spikes.

A2. Estimation of spike-amplitude ratios for each single spike.

A3. Single spikes sorting.

A4. Waveform template and mixing matrix estimations.

THE CLASSIFICATION PHASE

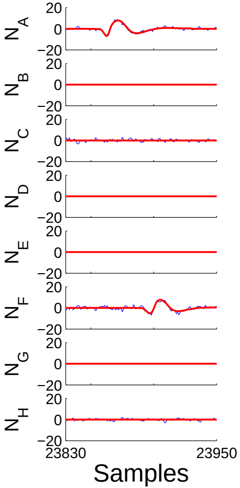
B1. Computation and Application of the pseudoinverse $(\tilde{A}C)^{\dagger}$ for each $C$.

B2. Estimation of the hypothetical signal $\hat{S}^{C}$ for each $C$.

B3. Selection of the matrix $\hat{C}$ that minimizes the distance between the measured data $Y$ and the hypothetical signal $\hat{S}^{C}$ re-mixed by the recording matrix $\tilde{A}$.

B4. Spike classification.

# Spike Classification



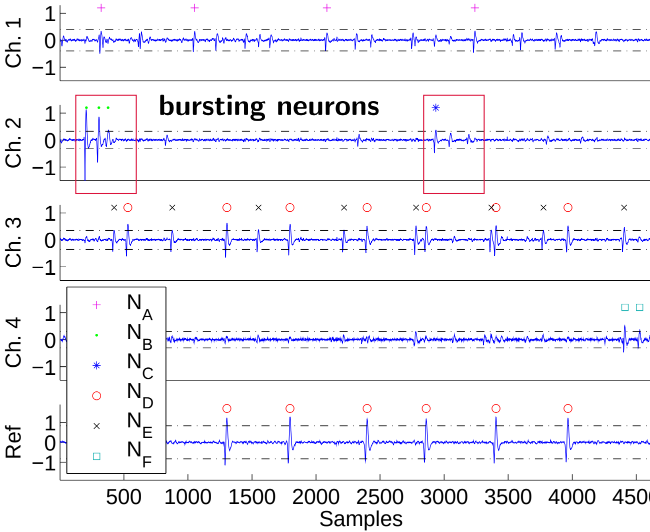

The rows of $(\tilde{A}\hat{C})^{\dagger}Y$ show which are the active neurons and the respective arrival times.

# Experimental Data

58 seconds of a recording in the Purkinje cells layer of a young rat cerebellar slice (linear probe)

reference channel

⟹



# Comparison with MCMC Algorithm

DEPARTMENT OF INFORMATION ENGINEERING
UNIVERSITY OF PADOVA

Comparison with the Markov Chain Monte Carlo algorithm described in [*"Efficient spike-sorting of multi-state neurons using inter-spike intervals information."*, Delescluse and Pouzat, **150**, J. Neurosci. Methods, 2006].

| | MCI4SC | MCMC |
|---|---|---|
| Detected spikes | 2739 | 2739 |
| Classified spikes | 2850 | 2739 |
| Reference spikes correctly classified | $636/641 \approx 99.2\%$ | $629/641 \approx 98.1\%$ |
| Non reference spikes wrongly classified | $8/644 \approx 1.2\%$ | $8/637 \approx 1.3\%$ |
| Computational Time | 134 sec. (1.73 *GHz*, Windows + MATLAB) | 33 min. (3 *GHz* PC, Linux + C) |

The two algorithms results almost comparable in efficiency, but MCI4SC takes much **lower computational time**, and it can resolve **overlapping spikes with complete superposition**.

# Conclusions 1/2



A new spike sorting method has been developed.

- It makes original use of the mixing matrix associated to the measurement channel.
- It handles the unfavorable situation where $m \leq n$ (under the hypothesis that the number of simultaneously firing neurons is $\leq m$).
- It can resolve overlapping of up to $m$ spikes, even when the superposition is complete.
- It can correctly classify bursting neurons that fire spikes with amplitude and waveform variations.
- The Wavelet Packet Transform provides a consistent estimation of the amplitude ratios even in case of low signal to noise ratio.
- It is independent from sensor geometry, even though it has better performances with non-planar configurations, that allow to univocally locate the neuron position.

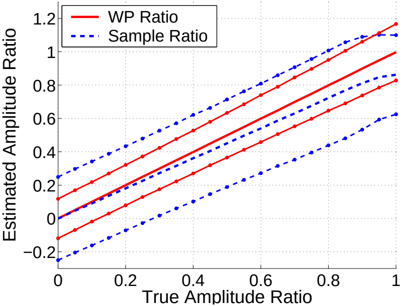
# Conclusions 2/2

Good spike sorting results have been obtained
for **noisy** neurons in the Locust antennal lobe,
and for **bursting** Purkinje Cells.

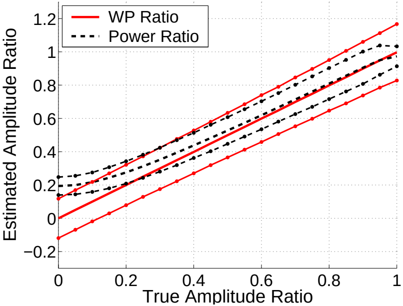
MCI4SC exploits either **neuron position** and **waveform information**;
improvements are expected including **temporal information**
related to the Inter-Spike Interval.

# Amplitude Ratios Estimation

Comparison with **Sample-based** estimator



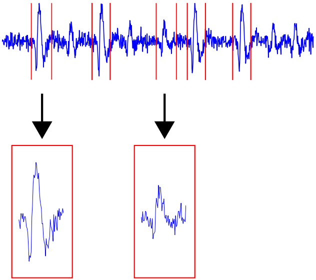

Comparison with **Power-based** estimator




The WP-based estimator gives, in mean, an estimation closer to the true value with an intermediate dispersion.

# Preliminary Steps

- Data are **filtered** with a band pass filter between 300 *Hz* and some *kHz*.
- All the events to put under analysis are **detected** in the data.
- Suitable **window** are cut around each detected spike to entirely contain the transient signal.



# Rejection of Overlapping Spikes



**Overlapping Spikes**

spikes with time length larger than the maximum single spike duration

spikes recorded with different spike waveforms by the electrodes (threshold on cross-correlation with zero lag)