# Sequential local FRI sampling of infinite streams of Diracs

Jon Oñativia, Jose Antonio Urigüen and Pier Luigi Dragotti

Imperial College London

ICASSP 2013, Vancouver (Canada)

May 30, 2013

# Outline

# Signals with Finite Rate of Innovation (FRI)

- Signals that have a finite number of free parameters

$$x(t) = \sum_{k \in \mathbb{Z}} \sum_{r=0}^{R-1} a_{k,r} \, g_r(t - t_k).$$

If the set of functions $\{g_r(t)\}_{r=0,1,\dots,R-1}$ is known, the signal $x(t)$ is perfectly determined by the coefficients $(a_{k,r}, t_k)$.

# Signals with Finite Rate of Innovation (FRI)

- Signals that have a finite number of free parameters

$$x(t) = \sum_{k\in\mathbb{Z}} \sum_{r=0}^{R-1} a_{k,r}\, g_r(t - t_k).$$

If the set of functions $\{g_r(t)\}_{r=0,1,\ldots,R-1}$ is known, the signal $x(t)$ is perfectly determined by the coefficients $(a_{k,r}, t_k)$.

- Let us constrain the input signal to a stream of K Diracs in an interval $\tau$ $x(t) = \sum_{k=1}^{K} a_k\, \delta(t - t_k)$, where $t_k \in [0, \tau]$.

# Signals with Finite Rate of Innovation (FRI)

- Signals that have a finite number of free parameters

$$x(t) = \sum_{k\in\mathbb{Z}} \sum_{r=0}^{R-1} a_{k,r}\, g_r(t-t_k).$$

If the set of functions $\{g_r(t)\}_{r=0,1,\dots,R-1}$ is known, the signal $x(t)$ is perfectly determined by the coefficients $(a_{k,r}, t_k)$.

- Let us constrain the input signal to a stream of K Diracs in an interval $\tau$ $x(t) = \sum_{k=1}^{K} a_k\, \delta(t-t_k)$, where $t_k \in [0,\tau]$.
  - This signal has $2K$ degrees of freedom in a temporal interval $\tau$
  - Local rate of innovation: $\rho = \frac{2K}{\tau}$

# Signals with Finite Rate of Innovation (FRI)

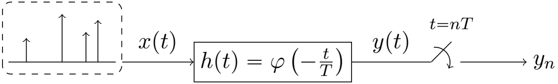
- Signals that have a finite number of free parameters

$$x(t) = \sum_{k\in\mathbb{Z}} \sum_{r=0}^{R-1} a_{k,r}\, g_r(t-t_k).$$

If the set of functions $\{g_r(t)\}_{r=0,1,\ldots,R-1}$ is known, the signal $x(t)$ is perfectly determined by the coefficients $(a_{k,r}, t_k)$.

- Let us constrain the input signal to a stream of K Diracs in an interval $\tau$ $x(t) = \sum_{k=1}^{K} a_k\, \delta(t-t_k)$, where $t_k \in [0,\tau]$.
  - This signal has $2K$ degrees of freedom in a temporal interval $\tau$
  - Local rate of innovation: $\rho = \frac{2K}{\tau}$

- We acquire the signal with a sampling device at regular intervals of time $t = nT$




- The output samples can be expressed as $y_n = \langle x(t), \varphi(t/T-n)\rangle$.

$$x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$$

- Classical sampling theory does not allow to sample and perfectly reconstruct a stream of Diracs because it is not a bandlimited signal.

$$x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$$

- Classical sampling theory does not allow to sample and perfectly reconstruct a stream of Diracs because it is not a bandlimited signal.
- The FRI framework can achieve perfect reconstruction under some conditions.

$$x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$$

- Classical sampling theory does not allow to sample and perfectly reconstruct a stream of Diracs because it is not a bandlimited signal.

- The FRI framework can achieve perfect reconstruction under some conditions.

- State of the art FRI algoritms do not deal well with infinite streams:
  - Based on isolating bursts of Diracs
  - Require high sampling rates

$$x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$$

- Classical sampling theory does not allow to sample and perfectly reconstruct a stream of Diracs because it is not a bandlimited signal.
- The FRI framework can achieve perfect reconstruction under some conditions.
- State of the art FRI algoritms do not deal well with infinite streams:
  - Based on isolating bursts of Diracs
  - Require high sampling rates
- We present a novel sequential algorithm that is able to reconstruct these type of signals:
  - Able to recover 1k Diracs from 10k samples
  - Robust under high noise conditions
  - Works in real time
  - Succesfully applied in neuroscience to infere spiking activity of individual neurons from calcium fluorescence imaging

# Sampling process

- We sample $x(t)$ with a very specific kernel: $\varphi(t)$ together with its shifted versions can reproduce exponentials of the form $e^{\alpha_m t}$

$$\sum_{n\in\mathbb{Z}} c_{m,n}\varphi(t-n) = e^{\alpha_m t}, \quad m = 0, 1, \ldots, P$$

# Sampling process

- We sample $x(t)$ with a very specific kernel: $\varphi(t)$ together with its shifted versions can reproduce exponentials of the form $e^{\alpha_m t}$

$$\sum_{n\in\mathbb{Z}} c_{m,n}\varphi(t-n) = e^{\alpha_m t}, \quad m = 0, 1, \dots, P$$

- A family of functions that satisfy the exponential reproducing property are the exponential splines (E-splines). The Fourier transform of the $P$-th order E-Spline with parameter $\vec{\alpha}_P = (\alpha_0, \alpha_1, \dots, \alpha_P)$ is given by

$$\hat{\varphi}_{\vec{\alpha}_P}(\omega) = \prod_{m=0}^{P} \left( \frac{1 - e^{\alpha_m - j\omega}}{j\omega - \alpha_m} \right)$$

# Sampling process

- We sample $x(t)$ with a very specific kernel: $\varphi(t)$ together with its shifted versions can reproduce exponentials of the form $e^{\alpha_m t}$

$$\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t-n) = e^{\alpha_m t}, \quad m = 0, 1, \ldots, P$$

- A family of functions that satisfy the exponential reproducing property are the exponential splines (E-splines). The Fourier transform of the $P$-th order E-Spline with parameter $\vec{\alpha}_P = (\alpha_0, \alpha_1, \ldots, \alpha_P)$ is given by

$$\hat{\varphi}_{\vec{\alpha}_P}(\omega) = \prod_{m=0}^{P} \left( \frac{1 - e^{\alpha_m - j\omega}}{j\omega - \alpha_m} \right)$$

- If coefficients $\alpha_m$ are real, or complex but appear in complex conjugate pairs, the kernel is real valued.
- E-splines present the advantage of being of compact support $P+1$.

Input signal: $x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$

Samples ($T = 1$): $y_n = \langle x(t), \varphi(t-n) \rangle$

$\varphi(t)$ satisfies: $\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t-n) = e^{\alpha_m t}, \quad \alpha_m = \alpha_0 + m\lambda$ and $m = 0, \ldots, P$

Input signal: $x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$

Samples ($T = 1$): $y_n = \langle x(t), \varphi(t-n) \rangle$

$\varphi(t)$ satisfies: $\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t-n) = e^{\alpha_m t}, \quad \alpha_m = \alpha_0 + m\lambda$ and $m = 0, \ldots, P$

- If we combine samples $y_n$ with coefficients $c_{m,n}$ we obtain a new set of measurements $s_m$ which can be expressed as a sum of exponentials:

$$s_m = \sum_n c_{m,n} \, y_n$$

Input signal: $x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$

Samples ($T = 1$): $y_n = \langle x(t), \varphi(t-n) \rangle$

$\varphi(t)$ satisfies: $\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t-n) = e^{\alpha_m t}$, $\alpha_m = \alpha_0 + m\lambda$ and $m = 0, \ldots, P$

- If we combine samples $y_n$ with coefficients $c_{m,n}$ we obtain a new set of measurements $s_m$ which can be expressed as a sum of exponentials:

$$s_m = \sum_n c_{m,n} \, y_n = \sum_{k=1}^{K} \underbrace{a_k \, e^{\alpha_0 t_k}}_{b_k} \left( \underbrace{e^{\lambda t_k}}_{u_k} \right)^m$$

Input signal: $x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$

Samples ($T = 1$): $y_n = \langle x(t), \varphi(t-n) \rangle$

$\varphi(t)$ satisfies: $\sum_{n\in\mathbb{Z}} c_{m,n} \varphi(t-n) = e^{\alpha_m t}, \quad \alpha_m = \alpha_0 + m\lambda$ and $m = 0, \dots, P$

- If we combine samples $y_n$ with coefficients $c_{m,n}$ we obtain a new set of measurements $s_m$ which can be expressed as a sum of exponentials:

$$s_m = \sum_n c_{m,n} \, y_n = \sum_{k=1}^{K} \underbrace{a_k \; e^{\alpha_0 t_k}}_{b_k} \left( \underbrace{e^{\lambda t_k}}_{u_k} \right)^m = \sum_{k=1}^{K} b_k \; u_k^m$$

Input signal: $x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$

Samples ($T = 1$): $y_n = \langle x(t), \varphi(t - n) \rangle$

$\varphi(t)$ satisfies: $\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t - n) = e^{\alpha_m t}$, $\alpha_m = \alpha_0 + m\lambda$ and $m = 0, \ldots, P$

- If we combine samples $y_n$ with coefficients $c_{m,n}$ we obtain a new set of measurements $s_m$ which can be expressed as a sum of exponentials:

$$s_m = \sum_n c_{m,n} \, y_n = \sum_{k=1}^{K} \underbrace{a_k \, e^{\alpha_0 t_k}}_{b_k} \Big( \underbrace{e^{\lambda t_k}}_{u_k} \Big)^m = \sum_{k=1}^{K} b_k \, u_k^m$$

- Retrieval of $a_k$ and $t_k$ from samples $s_m$ is a classical problem in spectral estimation or in direction of arrival (DOA) estimation
  - Can be solved for instance applying the annihilating filter method (a.k.a. Prony's method) or the matrix pencil method (inspired from ESPRIT)

Input signal: $x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$

Samples ($T = 1$): $y_n = \langle x(t), \varphi(t-n) \rangle$

$\varphi(t)$ satisfies: $\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t-n) = e^{\alpha_m t}, \quad \alpha_m = \alpha_0 + m\lambda$ and $m = 0, \ldots, P$

- If we combine samples $y_n$ with coefficients $c_{m,n}$ we obtain a new set of measurements $s_m$ which can be expressed as a sum of exponentials:

$$s_m = \sum_n c_{m,n} \, y_n = \sum_{k=1}^{K} \underbrace{a_k \, e^{\alpha_0 t_k}}_{b_k} \left( \underbrace{e^{\lambda t_k}}_{u_k} \right)^m = \sum_{k=1}^{K} b_k \, u_k^m$$

- Retrieval of $a_k$ and $t_k$ from samples $s_m$ is a classical problem in spectral estimation or in direction of arrival (DOA) estimation
  - Can be solved for instance applying the annihilating filter method (a.k.a. Prony's method) or the matrix pencil method (inspired from ESPRIT)

- These methods require a minimum number of values $s_m$ and this in turn imposes a minimal order $P$ for the sampling kernel: $P + 1 \geq 2K$.
  - Critical sampling is achieved for $P + 1 = 2K$

Input signal: $x(t) = \sum_{k=1}^{K} a_k \, \delta(t - t_k)$

Samples ($T = 1$): $y_n = \langle x(t), \varphi(t-n) \rangle$

$\varphi(t)$ satisfies: $\sum_{n \in \mathbb{Z}} c_{m,n} \varphi(t-n) = e^{\alpha_m t}, \quad \alpha_m = \alpha_0 + m\lambda$ and $m = 0, \ldots, P$

- If we combine samples $y_n$ with coefficients $c_{m,n}$ we obtain a new set of measurements $s_m$ which can be expressed as a sum of exponentials:

$$s_m = \sum_n c_{m,n} \, y_n = \sum_{k=1}^{K} \underbrace{a_k \; e^{\alpha_0 t_k}}_{b_k} \left( \underbrace{e^{\lambda t_k}}_{u_k} \right)^m = \sum_{k=1}^{K} b_k \; u_k^m$$

- Retrieval of $a_k$ and $t_k$ from samples $s_m$ is a classical problem in spectral estimation or in direction of arrival (DOA) estimation
  - Can be solved for instance applying the annihilating filter method (a.k.a. Prony's method) or the matrix pencil method (inspired from ESPRIT)

- These methods require a minimum number of values $s_m$ and this in turn imposes a minimal order $P$ for the sampling kernel: $P + 1 \geq 2K$.
  - Critical sampling is achieved for $P + 1 = 2K$

- If we have an infinite stream we face some problems:
  - This approach requires knowledge of all samples $y_n$ in order to compute $s_m$
  - The number of Diracs is infinite so the order of the E-spline must be infinite as well

# Sampling an infinite sequence of Diracs

- We consider a continuous time signal $x(t)$ formed by an infinite stream of Diracs, $\sum_{k\in\mathbb{Z}} a_k \, \delta\left(t-t_k\right)$.

# Sampling an infinite sequence of Diracs

- We consider a continuous time signal $x(t)$ formed by an infinite stream of Diracs, $\sum_{k\in\mathbb{Z}} a_k\,\delta\,(t-t_k)$.
- There are an infinite number of Diracs, but with a limited rate of at most $K$ Diracs per $\tau$ interval.

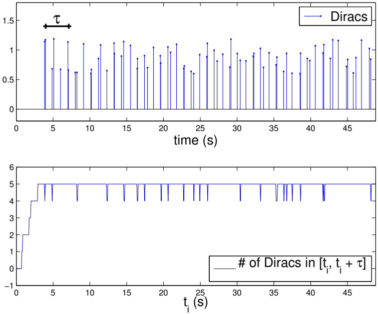



Figure: Infinite stream. Local maximum rate of innovation $\rho = 2K/\tau$ ($K = 5$, $\tau = 3.125$ s).

- We take advantage of the fact that the sampling kernel is of compact support $(P+1)T$. Thus, a Dirac can influence at most $P+1$ samples $y_n$.

- We take advantage of the fact that the sampling kernel is of compact support $(P+1)T$. Thus, a Dirac can influence at most $P+1$ samples $y_n$.
- The sequential algorithm estimates the locations of the Diracs within a sliding window that covers the interval of time $\tau = NT$.

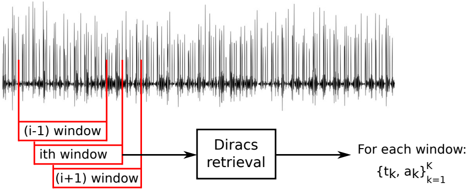



Figure: Sequential processing.

- We take advantage of the fact that the sampling kernel is of compact support $(P+1)T$. Thus, a Dirac can influence at most $P+1$ samples $y_n$.
- The sequential algorithm estimates the locations of the Diracs within a sliding window that covers the interval of time $\tau = NT$.

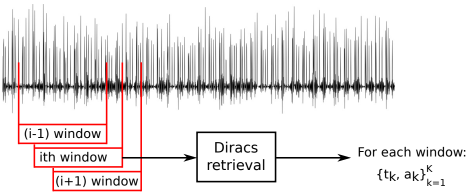



Figure: Sequential processing.

- Problem $\Rightarrow$ if we only process $N$ samples at a time there are border effects when Diracs are located near the borders of the sliding window

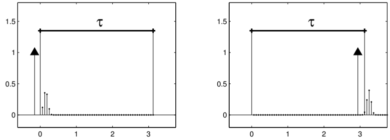



Figure: Border effects.

- The border effect in the left side is due to Dircas before the $\tau$ interval that leak into the $N$ samples $y_n$ of the current window.

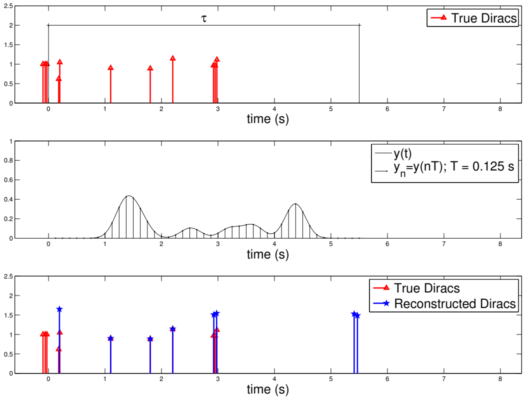



Figure: Diracs are not perfectly recovered because past Diracs corrupt current samples.

- The border effect in the left side is due to Dircas before the $\tau$ interval that leak into the $N$ samples $y_n$ of the current window.
- If we assume that we have already recovered Diracs up to the current position of the sliding window we can remove the contribution to $y_n$ of nearby Diracs that happened before.

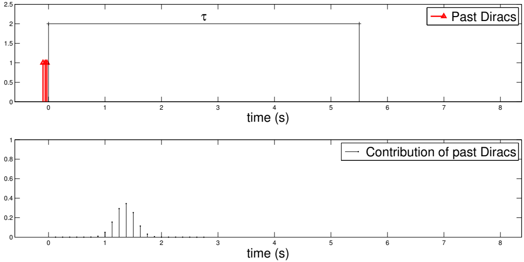



Figure: Contribution of past Diracs to samples $y_n$.

- The border effect in the left side is due to Dircas before the $\tau$ interval that leak into the $N$ samples $y_n$ of the current window.
- If we assume that we have already recovered Diracs up to the current position of the sliding window we can remove the contribution to $y_n$ of nearby Diracs that happened before.

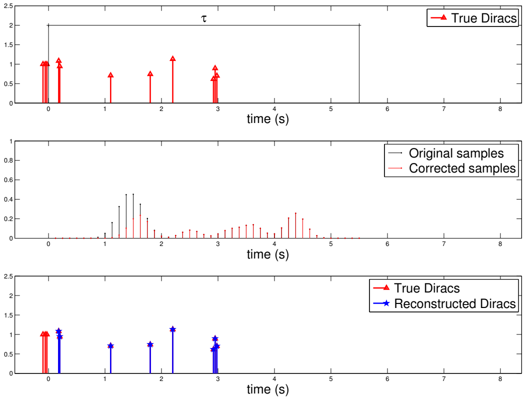



Figure: Perfect reconstruction after correcting past Diracs effect.

- The border effect on the right side is due to Diracs inside the $\tau$ interval that leak outside the $N$ samples $y_n$ of the current window.

- The border effect on the right side is due to Diracs inside the $\tau$ interval that leak outside the $N$ samples $y_n$ of the current window.
- To make sure that these Diracs will be recovered for a certain position of the sliding window we have to impose:

$$\boxed{T \leq \frac{1}{K\,\rho}} \qquad \text{and} \qquad \boxed{P + 1 = 2K}$$

- The border effect on the right side is due to Diracs inside the $\tau$ interval that leak outside the $N$ samples $y_n$ of the current window.
- To make sure that these Diracs will be recovered for a certain position of the sliding window we have to impose:

$$\boxed{T \leq \frac{1}{K\,\rho}} \qquad \text{and} \qquad \boxed{P+1=2K}$$

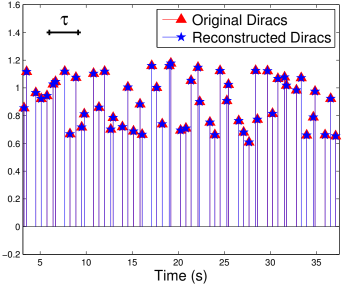



Figure: Sequential perfect reconstruction of a noiseless stream of Diracs. Section of a stream of $1000$ Diracs and $10220$ samples $y_n$. Rate $K = 5$ Diracs per $\tau = 3.125$ s, $N = 50$ samples, $T = 1/16$ s and order of the E-spline $P = 9$.

# The noisy scenario

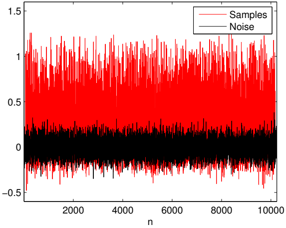
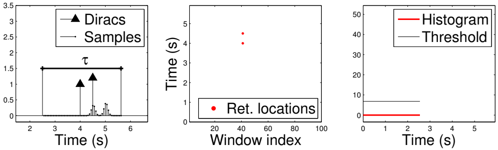


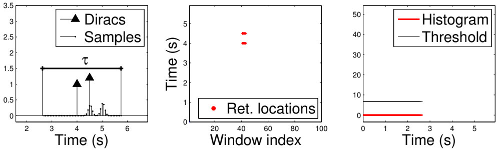

Figure: 1k Diracs, 10k samples, SNR = 10 dB.

- Perfect reconstruction conditions do not hold anymore.
- We can relax conditions on $T$ and $P$
  - We allow the sampling kernel to be of higher order in order to be more robust against noise.
- The idea is to estimate Diracs by analysing the consistency of the retrieved locations among different positions of the sliding window.

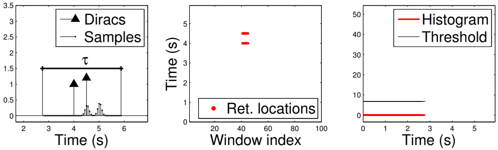
- A Dirac is captured among different positions of the sliding window:
  - If a retrieved location corresponds to a true Dirac this location will be consistent among different positions of the sliding window.



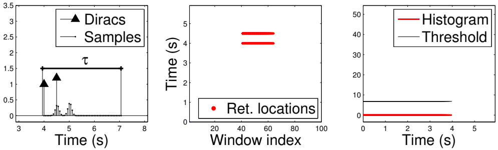
- A Dirac is captured among different positions of the sliding window:
  - If a retrieved location corresponds to a true Dirac this location will be consistent among different positions of the sliding window.



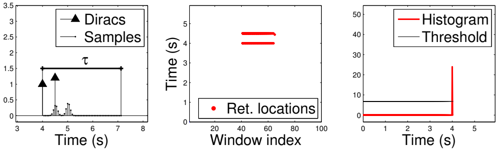
- A Dirac is captured among different positions of the sliding window:
  - If a retrieved location corresponds to a true Dirac this location will be consistent among different positions of the sliding window.



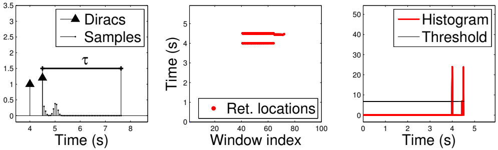
- A Dirac is captured among different positions of the sliding window:
  - If a retrieved location corresponds to a true Dirac this location will be consistent among different positions of the sliding window.



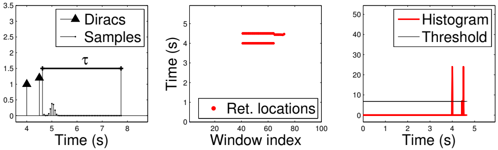
- A Dirac is captured among different positions of the sliding window:
  - If a retrieved location corresponds to a true Dirac this location will be consistent among different positions of the sliding window.



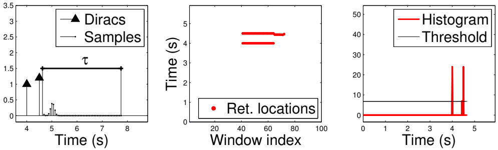
- A Dirac is captured among different positions of the sliding window:
  - If a retrieved location corresponds to a true Dirac this location will be consistent among different positions of the sliding window.



- A Dirac is captured among different positions of the sliding window:
  - If a retrieved location corresponds to a true Dirac this location will be consistent among different positions of the sliding window.



- A Dirac is captured among different positions of the sliding window:
  - If a retrieved location corresponds to a true Dirac this location will be consistent among different positions of the sliding window.




- If we analyse the consistency of the retrieved locations we can estimate the Diracs from the peaks of the histogram of the locations:

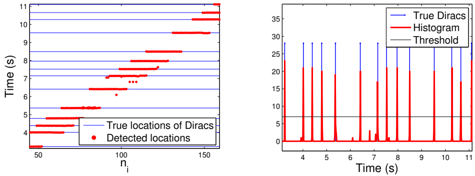



Figure: Retrieved locations among different positions of the sliding window and histogram of locations.

- The consistency analysis makes the retrieval algorithm robust against noise.

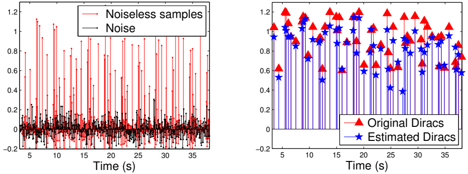



Figure: Sequential reconstruction of a noisy stream of Diracs (SNR = 10 dB). Section of a stream of $1000$ Diracs and $10220$ samples $y_n$. Rate $K = 5$ Diracs per $\tau = 3.125$ s, $N = 50$ samples, $T = 1/16$ s and order of the E-spline $P = 22$.

- The consistency analysis makes the retrieval algorithm robust against noise.

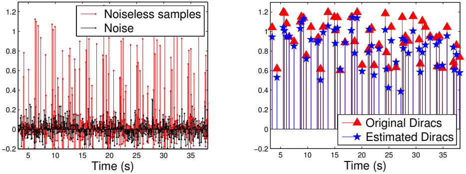


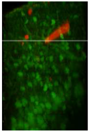

Figure: Sequential reconstruction of a noisy stream of Diracs (SNR = 10 dB). Section of a stream of $1000$ Diracs and $10220$ samples $y_n$. Rate $K = 5$ Diracs per $\tau = 3.125$ s, $N = 50$ samples, $T = 1/16$ s and order of the E-spline $P = 22$.

- Some results for differents levels of noise (experiment repeated 100 times for each level of noise):

| SNR (dB) | 5 | 10 | 15 | 20 |
|---|---|---|---|---|
| Detection rate | 97.69 % | 99.97 % | 100.00 % | 100.00 % |
| False positives | 351.7 | 37.8 | 0.5 | 0.3 |
| Precision (s) | 0.0086 | 0.0049 | 0.0028 | 0.0018 |

# Application: neural activity detection

- This framework has been successfully applied to the detection of neural activity in calcium concentration movies [1].



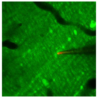


Figure: Simultaneous multiphoton calcium imaging of a region of the cortex and electrophisiological recording of a targeted cell with a micropipette.

[1] Jon Oñativia, Simon R. Schultz and Pier Luigi Dragotti. *A finite rate of innovation algorithm for fast and accurate spike detection from two-photon calcium imaging.* to appear in Journal of Neural Engineering

# Application: neural activity detection

- This framework has been successfully applied to the detection of neural activity in calcium concentration movies [1].

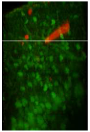
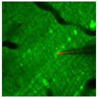



Figure: Simultaneous multiphoton calcium imaging of a region of the cortex and electrophisiological recording of a targeted cell with a micropipette.

- Fluorescence sequences obtained by averaging the pixel values of a ROI can be modeled as a stream of decaying exponentials:

$$c(t) = A \sum_k e^{-\alpha(t-t_k)} u(t - t_k)$$

[1] Jon Oñativia, Simon R. Schultz and Pier Luigi Dragotti. *A finite rate of innovation algorithm for fast and accurate spike detection from two-photon calcium imaging.* to appear in Journal of Neural Engineering

# Application: neural activity detection

- This framework has been successfully applied to the detection of neural activity in calcium concentration movies [1].

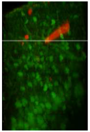
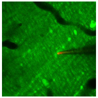



Figure: Simultaneous multiphoton calcium imaging of a region of the cortex and electrophisiological recording of a targeted cell with a micropipette.

- Fluorescence sequences obtained by averaging the pixel values of a ROI can be modeled as a stream of decaying exponentials:

$$c(t) = A \sum_k e^{-\alpha(t-t_k)}\, u(t-t_k) = \underbrace{\sum_k \delta(t-t_k)}_{x(t)} * \underbrace{A\, e^{-\alpha t}\, u(t)}_{\rho_\alpha(t)}$$

[1] Jon Oñativia, Simon R. Schultz and Pier Luigi Dragotti. *A finite rate of innovation algorithm for fast and accurate spike detection from two-photon calcium imaging.* to appear in Journal of Neural Engineering

# Application: neural activity detection

- This framework has been successfully applied to the detection of neural activity in calcium concentration movies [1].

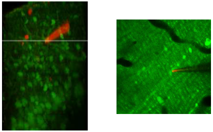


Figure: Simultaneous multiphoton calcium imaging of a region of the cortex and electrophisiological recording of a targeted cell with a micropipette.

- Fluorescence sequences obtained by averaging the pixel values of a ROI can be modeled as a stream of decaying exponentials:

$$c(t) = A \sum_k e^{-\alpha(t-t_k)} u(t-t_k) = \underbrace{\sum_k \delta(t-t_k)}_{x(t)} * \underbrace{A\, e^{-\alpha t} u(t)}_{\rho_\alpha(t)} = x(t) * \rho_\alpha(t).$$

[1] Jon Oñativia, Simon R. Schultz and Pier Luigi Dragotti. *A finite rate of innovation algorithm for fast and accurate spike detection from two-photon calcium imaging.* to appear in Journal of Neural Engineering

# Application: neural activity detection

- This framework has been successfully applied to the detection of neural activity in calcium concentration movies [1].

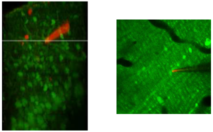


Figure: Simultaneous multiphoton calcium imaging of a region of the cortex and electrophisiological recording of a targeted cell with a micropipette.

- Fluorescence sequences obtained by averaging the pixel values of a ROI can be modeled as a stream of decaying exponentials:

$$c(t) = A \sum_k e^{-\alpha(t-t_k)} u(t-t_k) = \underbrace{\sum_k \delta(t-t_k)}_{x(t)} * \underbrace{A\, e^{-\alpha t} u(t)}_{\rho_\alpha(t)} = x(t) * \rho_\alpha(t).$$

- This is a Finite Rate of Innovation signal and with a correct processing of the fluorescence samples we can apply our sequential algorithm.

[1] Jon Oñativia, Simon R. Schultz and Pier Luigi Dragotti. *A finite rate of innovation algorithm for fast and accurate spike detection from two-photon calcium imaging.* to appear in Journal of Neural Engineering

- We achieve 84 % detection rates with real data (calcium fluorescence sequence) for electrophysiologically confirmed action potentials.

- We achieve 84 % detection rates with real data (calcium fluorescence sequence) for electrophysiologically confirmed action potentials.

- We outperform state of the art real time spike train inference algorithms:

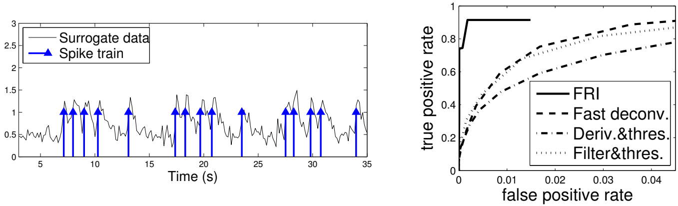



Figure: Receiver operating characteristic (ROC) curves for various algorithms with surrogate data (SNR = 10 dB).

- We achieve 84 % detection rates with real data (calcium fluorescence sequence) for electrophysiologically confirmed action potentials.

- We outperform state of the art real time spike train inference algorithms:

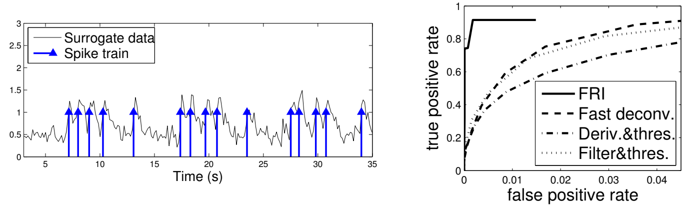



Figure: Receiver operating characteristic (ROC) curves for various algorithms with surrogate data (SNR = 10 dB).

- This technique can be used to monitor tens of neurons simultaneously since the fluorsecence movie captures a volume that contains many neurons.

- We achieve 84 % detection rates with real data (calcium fluorescence sequence) for electrophysiologically confirmed action potentials.

- We outperform state of the art real time spike train inference algorithms:

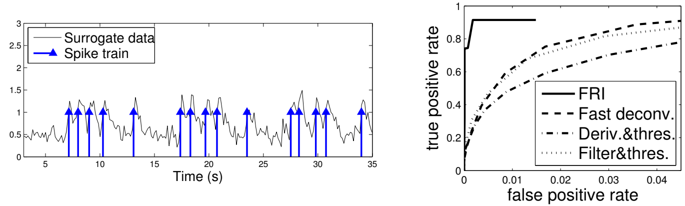



Figure: Receiver operating characteristic (ROC) curves for various algorithms with surrogate data (SNR = 10 dB).

- This technique can be used to monitor tens of neurons simultaneously since the fluorsecence movie captures a volume that contains many neurons.

- The algorithm is fast enough to perform real-time spike inference:
  - The current MATLAB implementation can process more than 80 datastreams in parallel on a commercial laptop (2.5 GHz Intel Core i5 CPU).

# Questions?