REDUCTION OF NECESSARY DATA RATE FOR NEURAL DATA THROUGH EXPONENTIAL AND SINUSOIDAL SPLINE DECOMPOSITION USING THE FINITE RATE OF INNOVATION FRAMEWORK

Tobias Schnier, Carsten Bockelmann, Armin Dekorsy

Department of Communications Engineering
University of Bremen, Germany
Email: schnier@ant.uni-bremen.de

ABSTRACT

The sampling of neural signals plays an important role in modern neuroscience, especially for prosthetics. However, due to hardware and data rate constraints, only spike trains can get recovered reliably. State of the art prosthetics can still achieve impressive results, but to get higher resolutions the used data rate needs to be reduced. In this paper, this is done by expressing the data with exponential and sinusoidal splines. As these signals have a finite number of degrees of freedom per unit of time, they can be analyzed and reconstructed with the Finite Rate of Innovation (FRI) framework. We show, that we can reduce the needed data rate by 90% to achieve the same resolution as without compression. Additionally, we propose analytic boundaries for the reconstruction of these splines and present an algorithm that guarantees the reconstruction within these boundaries. Furthermore, we test the algorithm on real neural stimuli.

***Index Terms*—** FRI,exponential,sinusoidal,splines,neural

1. INTRODUCTION

Progress in modern neuroscience lead to highly functioning prostheses that can be controlled directly by observing and interpreting spike trains of neural signals [1],[2]. Today, sampling spike trains is sufficient, but resolutions should be increased for next generation products. This way, even delicate movements may be implemented, leading to an overall improvement of life for affected persons. This possible advance inspired us to take the signal processing perspective on this problem. To solve the resolution problem, two frameworks come to mind. First, Compressed Sensing (CS), which relies on sparse representation of the data in question. Sadly, neural data are not easily representable by common dictionaries, so the use of CS is limited. Nevertheless, several publications have been done in this field. For example, Zhang et al. [3] propose a neural spike reconstruction with a compression rate of 8 to 16, but are unable to reconstruct the whole neural signal. Sun et al. [4] use co-sparsity to improve on this result, gaining a compression rate of 8 for the complete neural signal, but rely on learning a dictionary with a significantly higher number of samples. Second, the *Finite Rate of Innovation* (FRI) framework is applicable, as introduced in the paper of Vetterli et al. [5]. The main task of FRI is the reconstruction of functions that can be described by a finite number of coefficients. Examples include streams of Diracs, nonuniform polynomial splines and piecewise polynomials [6]. Sadly, similar to CS, the state of the art FRI algorithms can not be applied directly, as neural signals can not be described well with these functions. Due to this, we generalized the reconstructable functions in the FRI framework to exponential and sinusoidal spline. As a huge share of natural processes are decay, growth or oscillation processes these function play an important role in nature. Especially neural signals are easily described as a combination of both families. As all of these signals have finite degrees of freedom, a finite amount of samples is enough for perfect reconstruction. Hereby a generalization of the famous Shannon sampling theorem [7] for bandlimited signals is reached, that helps us in the reconstruction of neural data with fewer samples then state of the art algorithms would allow.

This work was funded by the German Research Foundation (DFG) under grant DE 759/5-1.

1.1. Main Contribution

We present a FRI algorithm, which is capable of reconstructing exponential and sinusoidal splines and especially neural signals. The reached compression rate for perfect recovery is 10 and no additional dictionary learning step is needed. For the easier spike reconstruction task, the compression rate can be increased to 200. The algorithm is applicable for *any* nonzero mean sampling filter design, including localized filters, that do not fulfill the Strang-Fix conditions [8]. Additionally, the optimal filter design regarding noise suppression is analyzed and optimal solutions are proposed. Furthermore, we present an analysis of the reconstruction problem to find theoretical bounds for the given problem. We show that our design works with the theoretical minimum of samples (the rate of innovation of the function), but can exploit arbitrary many more in order to better cope with noise.

2. RECONSTRUCTION OF FRI SIGNALS

2.1. Exponential functions

For clarification, we first reconstruct single exponential and sinusoidal functions and then later expand the system to corresponding splines. First, let $x(t)$ be an exponential function of the type

$$x(t) = a\exp(bt) + d \tag{1}$$

and let

$$y_i = \langle x(t+iT), \Phi(t)\rangle \tag{2}$$

be the corresponding equidistant samples, using the sampling filter $\Phi(t)$. The rate of innovation ρ is equal to 3 (unknown a, b and d), thus we only require three samples $y_1, \cdots, y_3$ for the reconstruction. With known b the reconstruction may be cast into a linear regression problem, which can be easily solved [9]. The nonlinearity introduced through b forbids this direct approach. Luckily, the annihilating filter [10] can be used to solve this problem, as it can directly find sums of unknown exponential bases from equidistant samples. The function $x(t)$ can be described as

$$x(t) = a\exp(b)^t + d\exp(0)^t, \tag{3}$$

thus equidistant samples $x(iT)$ can be used to build an annihilating filter for $x(t)$. This way we can reconstruct $\exp(b)$ as part of the sum of exponential bases and thus retrieve b. The corresponding annihilating filter to fulfill $H(t) * x(t) = 0$ is

$$\begin{aligned} H(t) &= (t-\exp(b))(t-e^0) \\ &= t^2 - t\underbrace{(e^b+1)}_{h_1} + \underbrace{e^b}_{h_2}. \end{aligned}$$

As the coefficients h_1 and h_2 of $H(t)$ can be computed by the formula (see [10] for more details)

$$\begin{pmatrix} y_3 & y_2 & y_1 \\ \vdots & \ddots & \vdots \\ y_m & y_{m-1} & y_{m-2} \end{pmatrix} \begin{pmatrix} 1 \\ h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \tag{4}$$

we can rewrite the problem in order to directly find b by inserting the known formulas for h_1 and h_2 as

$$\begin{pmatrix} y_3 & y_2 & y_1 \\ \vdots & \ddots & \vdots \\ y_m & y_{m-1} & y_{m-2} \end{pmatrix} \begin{pmatrix} 1 \\ -e^b-1 \\ e^b \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{5}$$

with the solution for the minimal number of three samples being

$$b = \log\left(\frac{y_3-y_2}{y_2-y_1}\right)/T. \tag{6}$$

As we never used any aspect of the filter $\Phi(t)$, this reconstruction formula holds for *any* filter. After b is computed, a standard regression can be computed by solving

$$\begin{pmatrix} \langle e^{b(t+1T)}, \Phi(t)\rangle & \langle \mathbf{1}, \Phi(t)\rangle \\ \vdots & \vdots \\ \langle e^{b(t+3T)}, \Phi(t)\rangle & \langle \mathbf{1}, \Phi(t)\rangle \end{pmatrix} \begin{pmatrix} a \\ d \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_3 \end{pmatrix} \tag{7}$$

to calculate the remaining coefficients and thus reconstruct the complete function. Note that $\langle \mathbf{1}, \Phi(t)\rangle$ needs to be unequal to zero to obtain a full rank system. In other words, this scheme only works with non zero mean filter designs.

2.2. Sinusoidal functions

Now, let $x(t)$ be a sinusoidal function of the type

$$x(t) = a\cos(bt+c) + d \tag{8}$$

and let

$$y_i = \langle x(t+iT), \Phi(t)\rangle \tag{9}$$

be the corresponding samples for $i = 1, \ldots, 4$. Again, four samples are sufficient due to the number of unknowns. As $\cos(x) = \frac{1}{2}(\exp(ix) + \exp(-ix))$ holds, we can create a similar scheme as before. Note, that the function $x(t)$ can be described as

$$x(t) = a\exp(ic)\exp(ib)^t + a\exp(-ic)\exp(-ib)^t + d\exp(0), \tag{10}$$

therefore the same idea can be used to retrieve the exponential parts of the function. The corresponding annihilating filter is

$$\begin{aligned} H(t) &= (t-e^{ib})(t-e^{-ib})(t-e^0) \\ &= (t^2 - 2t\cos(b) + 1)(t-1) \\ &= t^3 - t^2\underbrace{(1+2\cos(b))}_{h_1} + t\underbrace{(1+\cos(b))}_{h_2} - \underbrace{1}_{h_3} \end{aligned}$$

As before, the coefficients h_i for $i = 1, \ldots, 3$ of $H(t)$ can be computed by the formula

$$\begin{pmatrix} y_4 & y_3 & y_2 & y_1 \\ \vdots & \vdots & \vdots & \vdots \\ y_m & y_{m-1} & y_{m-2} & y_{m-3} \end{pmatrix} \begin{pmatrix} 1 \\ h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{11}$$

With the definition above, this leads to

$$\begin{pmatrix} y_4 & y_3 & y_2 & y_1 \\ \vdots & \vdots & \vdots & \vdots \\ y_m & y_{m-1} & y_{m-2} & y_{m-3} \end{pmatrix} \begin{pmatrix} 1 \\ -1-2\cos(b) \\ 1+2\cos(b) \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \tag{12}$$

with the solution for a minimum of four samples being

$$b = \operatorname{acos}\left(\frac{y_4-y_1}{2(y_3-y_2)} - \frac{1}{2}\right)/T. \tag{13}$$

After b is computed, we can use the linear regression with

$$\mathbf{A} := \begin{pmatrix} \langle e^{ib(t+1T)}, \Phi(t)\rangle & \langle e^{-ib(t+1T)}, \Phi(t)\rangle & \langle \mathbf{1}, \Phi(t)\rangle \\ \vdots & \vdots & \vdots \\ \langle e^{ib(t+4T)}, \Phi(t)\rangle & \langle e^{-ib(t+4T)}, \Phi(t)\rangle & \langle \mathbf{1}, \Phi(t)\rangle \end{pmatrix} \quad (14)$$

to solve

$$\mathbf{A} \begin{pmatrix} \frac{a}{2}e^{ic} \\ \frac{a}{2}e^{-ic} \\ d \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_4 \end{pmatrix} \quad (15)$$

to get the remaining coefficients. Additionally, the sinusoidal reconstruction scheme reconstructs exponential functions as well, because they can be interpreted as complex sinusoidal functions. In other words, if given at least 4 samples of the unknown kind of model, compute b as in (13). If b is imaginary, an exponential function was sampled. If b is real, the corresponding sinusoidal function was sampled.

2.3. Generalization to exponential/sinusoidal splines

Typically, signals (e.g. neural data) only have constant exponents over short time frames before changing to the next constant, so an *exponential* or *sinusoidal* spline formulation is useful. Formally an *exponential spline* is defined as

$$\begin{aligned} x_i(t) &= a_i \exp(b_i t) + d_i \text{ for } t_i \le t \le t_{i+1} \\ x_i(t_{i+1}) &= x_{i+1}(t_i) \quad \text{for } 1 \le i \le N \end{aligned} \quad (16)$$

Another important type of function is the *sinusoidal spline*. It is formally defined as

$$\begin{aligned} x_i(t) &= a_i \cos(b_i t + c_i) + d_i \text{ for } t_i \le t \le t_{i+1} \\ x_i(t_{i+1}) &= x_{i+1}(t_i) \quad \text{for } 1 \le i \le N \end{aligned} \quad (17)$$

This leads to a new issue, because the switching points t_i between parts are unknown a priori. Luckily, this can be solved by a piece-wise approach. First, compute the parameters for the first 4 samples with the above mentioned method. Then, find the first sample that cannot be expressed by the reconstructed function within some noise tolerance and then repeat the parameter identification. Through this scheme every part of the signal can be reconstructed. As the knots of the spline typically do not fall together with the sampling points, intersections of the parts have to be computed (e.g. Newton method) to obtain the full reconstruction of the signal. The full algorithm is presented in Algorithm 1.

The complexity of the reconstruction of each part is dominated by the solution of one linear equation system of size 4×3. In other words for N parts, the complexity of the parameter estimation in Algorithm 1 is $\mathcal{O}(4^2 3N) = \mathcal{O}(N)$. In the spline case, the complexity of the intersection search is added.

Algorithm 1 Spline Reconstruction

Require: y, T, Φ, tol

```
j = 1 (sample index), i = 0 (part index)
while j ≤ length(y) do
    b_i = acos( (1/2) ( (y_{j+3} − y_j)/(y_{j+2} − y_{j+1}) − 1 ) ) / T
    i = i + 1
    if isreal(b_i) then
        Solve (14) for a_i, c_i, d_i
        x_i(t) = a_i cos(b_i t + c_i) + d_i
    else
        b_i = imag(b_i)
        Solve (7) for a_i, d_i (set c_i = 0)
        x_i(t) = a_i exp(b_i t) + d_i
    end if
    Increment j until ||⟨x_i(t + jT), Φ(t)⟩ − y_j|| > tol
end while
for all parts x_i(t) do
    Find intersections I_j between parts (Newton method)
end for
Reconstruct x with parameters a_{1:i}, b_{1:i}, c_{1:i}, d_{1:i} and intersections I_j
return x
```

3. DEALING WITH NOISE

In reality samples are never noise free or the proposed model did not account for every aspect of the sample, which can be interpreted as a model mismatch. As shown before, the filter has no influence on the recovered function at all, but only because the scheme directly utilizes the exponential/sinusoidal model. This does not hold true for the noise as it is typically introduced as

$$y_i = \langle x(t + iT) + n(t + iT), \Phi(t)\rangle \quad (18)$$

Therefore, the question for the optimal filter arises that maximizes the signal to noise ratio. Assuming a normalized filter and noise with zero mean and σ_n^2 variance, the noise power is only dependent on σ_n^2 because of

$$\begin{aligned} E_n &= \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} (n(iT - t) * \Phi(t))^2 dt \Big|_{i=0} \\ &\overset{(a)}{=} \lim_{T\to\infty} \frac{1}{T} \left(\int_{-T/2}^{T/2} N^2(-j\omega)) d\omega \int_{-T/2}^{T/2} \phi^2(j\omega) d\omega \right) \\ &\overset{(b)}{=} \sigma_n^2 \cdot 1 = \sigma_n^2 \end{aligned}$$

where (a) comes from the Parseval identity and (b) holds, because of the flat spectrum of n and the normalized filter. Similarly, the signal power is

$$E_s = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} (x(-t) * \Phi(t))^2 dt$$

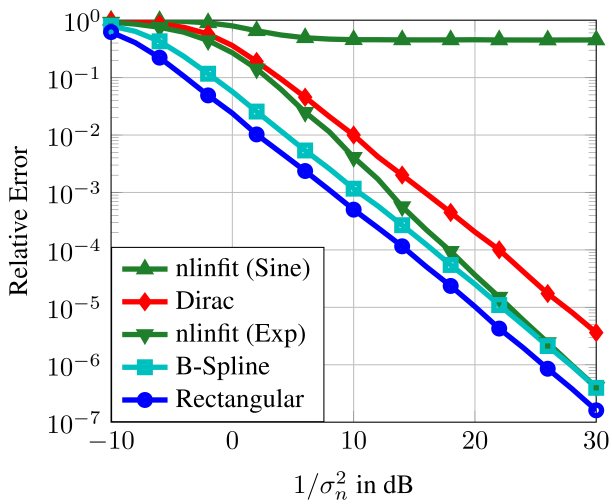

Fig. 1. Relative error of the reconstruction of one function for different filter designs and nlinfit, plotted against the noise variance

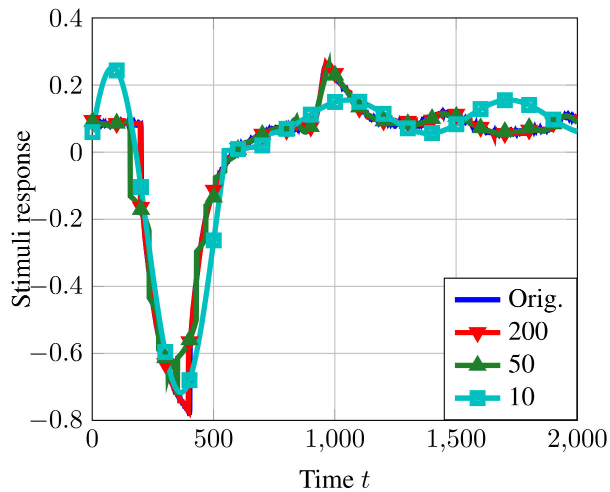

Fig. 2. Reconstruction of *neural data* with a rectangular filter for different sample sizes, plotted against the original data

In order to maximize the signal to noise ratio, due to Cauchy-Schwarz the equation $\Phi(t) = x(-t)$ should be fulfilled. As only the signal class of $x(t)$ is known at the filter stage, an overall good filter should match every spline part of $x(t)$ in the mean. In other words, $x_{\text{mean}}(t) = a\cos(0t + c) + d = a\cos(c) + d$ should be matched as $\Phi_{\text{opt}}(t) = x_{\text{mean}}(-t) =$ const. In conclusion, the best filter to maximize the signal to noise ratio should be an integrate and dump filter, that adds up the signal until the next sample begins.

4. NUMERICAL RESULTS

This section will compare the different filter designs with the MATLAB® function *'nlinfit'* (nonlinear fit) as benchmark. Three different filter designs (rectangular, dirac filter, linear B-spline) are used for the reconstruction of a unknown exponential or sinusoidal function in the presence of noise. In Fig. 1, the relative error of the reconstructed function is plotted against the noise variance in dB. It is directly visible, that filter design has a massive impact on the reconstruction. As mentioned in the previous section, the rectangular filter outperforms other filter designs by a vast margin, e.g. 8dB in comparison to the Dirac filter. The only other filter design capable of comparable results is the B-spline filter (6 dB gain) which is still a good match for the original function. As a comparison the MATLAB® function *'nlinfit'* is used, which can be used to fit any given model to data points. Due to the fact that *'nlinfit'* needs to know the underlying model, we divided the results into two curves for sinusoidal and exponential fits. In both cases the algorithm was given the correct model to match to. The results show, that the algorithm is able to find exponential functions and even performs better than just sampling and using our proposed algorithm (asymptotically 7dB gain). In comparison to more sophisticated filter designs it falls short by a large margin. This is due to the fact, that *'nlinfit'* only works directly with data points and is thereby unable to cope with noise which a rectangular filter would allow. The fit of sinusoidal functions does not work at all due to the non monotonicity of the underlying model function that makes it extremely ill posed.

Fig.2 shows the reconstruction of neural data of the auditory cortex as it poses the problem of several sinusoidal or exponential parts with unknown knots but known continuity between the parts. Here, the reconstruction of a rectangular filter with three different sample sizes (200, 50 and 10) is depicted against the original data for the rectangular filter design. As the data has intrinsic noise, it showcases a real life application and gives a good test for the algorithm in a non-simulated scenario. The results show, that a sample size of 200 samples is enough for perfect reconstruction of the original data despite already reducing the sample size by 90% in comparison to the 2000 original time instances given. Reducing even further to 50 samples (97,5% reduction) has nearly no visible difference, only sharp edges and spikes are not detected perfectly. The highest reduction of 99,5% (10 out of 2000 samples used) can still be used to obtain the overall shape and is more than enough for spike detection, while not being able to identify fine details.

5. CONCLUSION

We proposed an algorithm, that can reconstruct exponential and sinusoidal splines in the presence of noise with a very low amount of samples, outperforming state of the art CS algorithms by a large margin. The algorithm can work with any sampling filter design, but specific designs (e.g. a rectangular function) can increase the performance drastically.

6. REFERENCES

[1] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding," *Nature*, vol. 453, no. 7198, pp. 1098–1101, 2008.

[2] A. B. Schwartz, X. T. Cui, D. J. Weber, and D. W. Moran, "Brain-controlled interfaces: movement restoration with neural prosthetics," *Neuron*, vol. 52, no. 1, pp. 205–220, 2006.

[3] J. Zhang, Y. Suo, S. Mitra, S. P. Chin, S. Hsiao, R. F. Yazicioglu, T. D. Tran, and R. Etienne-Cummings, "An efficient and compact compressed sensing microsystem for implantable neural recordings," *IEEE transactions on biomedical circuits and systems*, vol. 8, no. 4, pp. 485–496, 2014.

[4] B. Sun, W. Zhao, and X. Zhu, "Compressed sensing for implantable neural recordings using co-sparse analysis model and weighted l-1-optimization," *arXiv preprint arXiv:1602.00430*, 2016.

[5] M. Vetterli, P. Marziliano, and T. Blu, "Sampling signals with finite rate of innovation," *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1417–1428, 2002.

[6] I. Maravic and M. Vetterli, "Sampling and reconstruction of signals with finite rate of innovation in the presence of noise," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2788–2805, 2005.

[7] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, 1949.

[8] P. L. Dragotti, M. Vetterli, and T. Blu, "Sampling moments and reconstructing signals of finite rate of innovation: Shannon meets strang–fix," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 1741–1757, 2007.

[9] N. R. Draper and H. Smith, *Applied regression analysis*. John Wiley & Sons, 2014.

[10] R. Prony, "Essai experimental–,-," *J. de l'Ecole Polytechnique*, 1795.