An Approach for the Reduction of Boundary Effects in Time-Frequency Representations

Adrien Meynard, Hau-Tieng Wu

***Abstract*—Time-frequency representations are intrinsically subject to the boundary effects. As a consequence, the structures of signals that are highlighted by the representations are garbled when approaching the boundaries of the time-frequency domain. In this paper, we propose an approach for the reduction of such boundary effects, whose objective is to be able to be used in real time. That one relies on an extension of the analyzed signal obtained by a forecasting technique. In the case of the study of a class of locally oscillating signals, we provide a theoretical study of the performance of our approach. Following a numerical verification of the performance of the algorithm, we implement it on real-life biomedical signals.**

***Index Terms*—Boundary effects, time-frequency, forecasting**

I. Introduction

In any digital acquisition system, the study and the interpretation of the measured signals generally require the use an analysis tool, which enable to point out the useful characteristics of the signal. For example, any measured biomedical signal, such as photoplethysmogram (PPG), may not be interpreted as it is, from its run chart. An analysis tool would make possible the extraction of some useful characteristics such as heart rate or blood pressure [1].

In general, the observed quantities are produced by transient phenomena, that can vary rapidly and irregularly. As a consequence, the measured signals exhibit nonstationary behavior. In order to adapt the analysis to nonstationarities, local analysis is generally performed. The short time Fourier transform (STFT) is a typical tool which is build that way, and enables the determination of the local frequency content of a nonstationary signal.

Windowing is a common method for performing local analysis. Among others, STFT, wavelet transform [2], synchrosqueezing transform (SST), and reassignment [3] (RS) are representations that fall back on the use of an analysis window. Let $x : I \to \mathbb{R}$ denote the observed signal, where I denotes the finite interval where the signal is measured. Let $g_s : \mathbb{R} \to \mathbb{R}$ denote the analysis window, where s is a shape parameter. The support of g_s is is localized around the origin, and is small with respect to $|I|$. The translation operator is T_τ defined as:

$$T_\tau f = f(t - \tau) \ , \quad \forall \ f : \mathbb{R} \to \mathbb{R} \ .$$

A. Meynard and H.-T. Wu are with the Department of Mathematics, Duke University, Durham, NC, 27708 USA e-mail: adrien.meynard@duke.edu

Then, the local analysis of x around the instant $\tau \in I$ rely on the evaluation of the following dot product:

$$V_x(s, \tau) = \langle x, T_\tau g_s \rangle_I \ . \tag{1}$$

A major shortcoming of this technique occurs when analyzing the signal x near the boundaries of the interval I. Indeed, at these points, half of the information is missing. Consequently the results of the dot product 1 is disturbed by this lack of information. This phenomenon is the so-called *boundary effect*. We display, on the left of Fig. 1 the result of the SST of a PPG (see section IV-B3 for a comprehensive description). The distortion resulting from the boundary effect is clearly visible on both sides of this representation. Indeed, while in the major central part of the image, clear lines stand out, they become blurred as they approach the left and right boundaries of the image. A zoom on the right boundary of this SST, displayed on the right of Fig. 1, emphasizes the result of boundary effect. More generally, signal characteristics, like instantaneous frequencies or amplitudes appear to be imprecisely determinable from the SST in the vicinity of the boundaries.

The purpose of this paper is to provide a fast algorithm allowing to tackle boundary effect and limit the disruptions it may cause on the signals representations based on windowing. Attempts to minimize the boundary effects generally consists in softening the discontinuity on signals edges. In that goal, a natural idea is to extend to signal beyond its edges. Hence, to implement our boundary effects reduction procedure, we proceed in two steps.

1) *Extend the signal by forecasting it.* The aim is to use a dynamic model to predict the values taken by the measured signal outside the measurement interval. Then, once this operation is done, we have access to an extended signal defined on a larger interval I_Δ, where Δ denotes the size of the extension on both boundaries of I.
2) *Run the local analysis tool on the extended signal.* Assuming that the support of the analysis window is smaller that 2Δ, the local analysis near the boundary of I is now possible without lack of information thanks to knowledge brought by the extension on both sides.

Thus, assuming that the quality of the extension step is sufficient, the representation obtained that way will be less sensitive to the boundary effect than the result of the analysis tool applied directly to the non-extended signal.

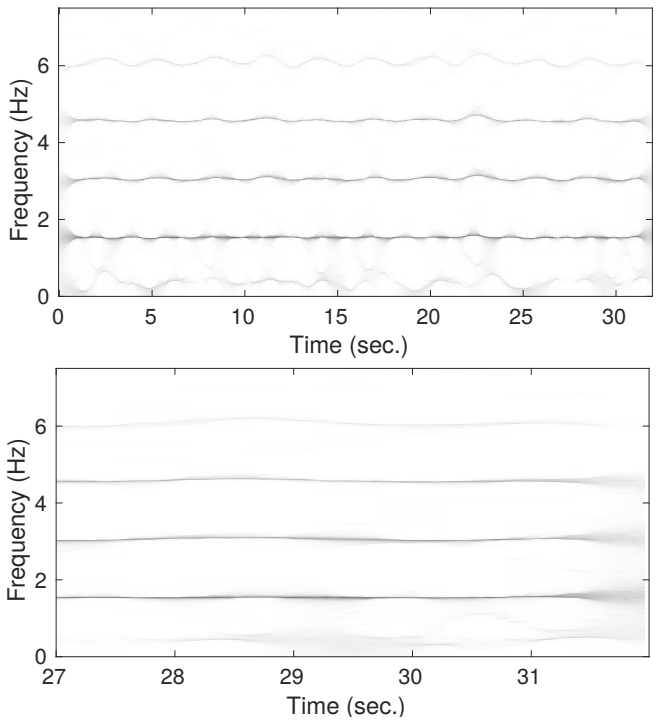

Figure 1: Synchrosqueezing transform of a PPG (top) with a zoom on its right boundary (bottom).

Various extension schemes have been proposed. On one hand, there exists simple extension schemes such as zero-padding, periodic extension, symmetric extension [4], that do not take into account the dynamical behavior of the signal. On the other hand, there exists extension schemes based on a dynamical model, such as the Empirical Dynamic Mode Decomposition [5] (EDMD), the Gaussian process regression [6] (GPR), the TBATS algorithm [7], or interpolation algorithm [8]. While the second class of extension schemes give better extended signals than the first class, they generally have a great computational cost. We propose here a fast extension algorithm based on a simple dynamical model, in order to optimize the trade-off between the extension quality and the computational cost. Furthermore, we prove that the dynamical model we consider is sufficient to forecast locally oscillating signals.

Therefore, in section II, we provide an extension method based on a linear dynamic model. We derive the corresponding algorithm for boundary effect reduction. In section III, we show that, even though we consider a simplistic dynamic model, it is sufficient to extend signals takings the form of sums of sine waves. An evaluation of the theoretical performance of our algorithm on a class of signals, the sums of sine waves, is given in section III. In section IV, we compare our extension method with more sophisticated methods such as EDMD, GPR, or TBATS. We show that our algorithm gives fast results of reasonable quality. Finally, we evaluate the performance of our boundary effect reduction algorithm on biomedical signals such as respiratory signals, and compare it to the theoretical results.

II. Algorithm

As explained above, the algorithm for the reduction of boundary effects on time-frequency representations relies on two steps. These ones are detailed in the current section.

A. Forecasting

Let $x : \mathbb{R} \to \mathbb{R}$ denote a continuous-time signal. In this work, we consider a finite-length discretization of that one. Thus, the sampled signal $\mathbf{x}$, whose length is denoted by N, is such that

$$\mathbf{x}[n] = x\left(\frac{n}{f_\mathsf{s}}\right) \ , \quad \forall n \in \{0, \dots, N-1\} \ ,$$

where f_s denotes the sampling frequency.

a) Notations: Let M and K be two integers such that $M < N$ and $K + M < N$. Then, for all $k \in \{0, \dots, K-1\}$, we extract from $\mathbf{x} \in \mathbb{R}^N$ the sub-signal $\mathbf{x}_k \in \mathbb{R}^M$ given by:

$$\mathbf{x}_k = \begin{pmatrix} \mathbf{x}[N-K+(k-1)-(M-1)] \\ \vdots \\ \mathbf{x}[N-K+(k-1)] \end{pmatrix} \ . \tag{2}$$

These sub-signals are gathered into the matrix $\mathbf{X} \in \mathbb{R}^{M \times K}$ such that:

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_0 & \cdots & \mathbf{x}_{K-1} \end{pmatrix} \ .$$

Notice that these sub-signals are overlapping each other. Indeed, $\mathbf{x}_{k+1}$ is a shifting of $\mathbf{x}_k$ from one sample. We also consider the matrix $\mathbf{Y} \in \mathbb{R}^{M \times K}$ given by:

$$\mathbf{Y} = \begin{pmatrix} \mathbf{x}_1 & \cdots & \mathbf{x}_K \end{pmatrix} \ .$$

b) Dynamical model and forecasting: Establishing a dynamical model consists in determining the relation linking $\mathbf{Y}$ to $\mathbf{X}$, that is finding a function f so that

$$\mathbf{Y} = f(\mathbf{X}) \ .$$

We consider here a naive dynamical model, assuming that we have the following relation:

$$\mathbf{Y} = \mathbf{A}\mathbf{X} \ , \tag{3}$$

where $\mathbf{A} \in \mathbb{R}^{M \times M}$. We adopt a classical strategy in the study of dynamical systems, that is the linearization of a nonlinear phenomenon. Notice that this linear dynamical model can be written equivalently in function of the sub-signals $\mathbf{x}_k$, as:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k, \ , \forall k \in \{0, \dots, K-1\} \ . \tag{4}$$

The forecasting method consists in estimating the unknown matrix $\mathbf{A}$. Indeed, let $\tilde{\mathbf{A}}$ denotes the estimate of $\mathbf{A}$, we then obtain the forecasting of the signal at time $\frac{N-1+\ell}{f_\mathsf{s}}$ by:

$$\tilde{\mathbf{x}}[N-1+\ell] = \boldsymbol{\alpha}^{(\ell)}\mathbf{x}_K \ , \tag{5}$$

where $\boldsymbol{\alpha}^{(\ell)}$ denotes the last row of $\tilde{\mathbf{A}}^{\ell}$, that is to say:

$$\boldsymbol{\alpha}^{(\ell)} = \mathbf{e}_M^T \tilde{\mathbf{A}}^{\ell} \ , \tag{6}$$

where $\mathbf{e}_M$ is the vector of length M given by $\mathbf{e}_M = \begin{pmatrix} 0 & \cdots & 0 & 1 \end{pmatrix}^T$.

c) Model estimation: To estimate the matrix $\mathbf{A}$, we basically implement the least square estimator. Thus, we solve the following problem:

$$\tilde{\mathbf{A}} = \arg\min_{\boldsymbol{\alpha}} \mathcal{L}(\mathbf{A}) \ , \tag{7}$$

where the loss function $\mathcal{L}$ is given by:

$$\mathcal{L}(\mathbf{A}) = \|\mathbf{Y} - \mathbf{A}\mathbf{X}\|^2 = \sum_{k=0}^{K-1} \|\mathbf{x}_{k+1} - \mathbf{A}\mathbf{x}_k\|^2.$$

Therefore, solving the problem (7), i.e. $\nabla\mathcal{L}(\tilde{\mathbf{A}}) = \mathbf{0}$, gives the following estimate $\tilde{\mathbf{A}}$ of the dynamical model matrix $\mathbf{A}$:

$$\tilde{\mathbf{A}} = \mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1} \ . \tag{8}$$

Remark 1. *This expression clearly shows that the matrix* $\tilde{\mathbf{A}}$ *takes the following form:*

$$\tilde{\mathbf{A}} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ \alpha_1 & \cdots & \cdots & \cdots & \alpha_M \end{pmatrix}.$$

Then, except the row vector $\boldsymbol{\alpha} = (\alpha_1 \cdots \alpha_M)$*, the matrix* $\mathbf{A}$ *is fully determined by the dynamical model.*

d) Signal extension: In order to reduce the boundary effects on both "sides" of the time-frequency (or time-scale) representation, we finally construct the extended signal $\tilde{\mathbf{x}} \in \mathbb{R}^{N+2L}$ concatenating the backward prediction $\tilde{\mathbf{x}}_{\text{bw}}$, the observed signal $\mathbf{x}$, and the forward prediction $\tilde{\mathbf{x}}_{\text{fw}}$. We summarize the extension step in Algorithm 1. Notice that we handle the backward estimation using the same strategy than described above, but applying it to the reverse signal $\mathbf{x}^{\text{r}} = \begin{pmatrix} \mathbf{x}[N-1] & \cdots & \mathbf{x}[0] \end{pmatrix}^T$.

Algorithm 1 Signal extension. $\tilde{\mathbf{x}} = \text{SigExt}(\mathbf{x}, M, K, L)$

Inputs: $\mathbf{x}$, M, K, L

Forward forecasting.
- LS estimation of the forward matrix $\tilde{\mathbf{A}}_{\text{fw}}$ via equation (8).
- Forward forecasting $\tilde{\mathbf{x}}_{\text{bw}}$ obtained applying equation (5) with $\ell \in \{1, \ldots, L\}$.

Backward forecasting.
- Reverse signal $\mathbf{x}$ to $\mathbf{x}^{\text{r}}$.
- LS estimation of the backward matrix $\tilde{\mathbf{A}}_{\text{bw}}$ via equation (8) applied to $\mathbf{x}^{\text{r}}$.
- Reversed backward forecasting $\tilde{\mathbf{x}}^{\text{r}}_{\text{bw}}$ obtained applying equation (5) to $\mathbf{x}^r$ with $\ell \in \{1, \ldots, L\}$.
- Reverse $\tilde{\mathbf{x}}^{\text{r}}_{\text{bw}}$ to obtain the estimate $\tilde{\mathbf{x}}_{\text{bw}}$.

Output: Extended signal $\tilde{\mathbf{x}} = \begin{pmatrix} \tilde{\mathbf{x}}_{\text{bw}} & \mathbf{x} & \tilde{\mathbf{x}}_{\text{fw}} \end{pmatrix}^T$.

B. Representation

Let $\mathscr{F}_N : \mathbb{R}^N \to \mathbb{R}^{F \times N}$ generically denotes the time-frequency or time-scale representation we are interested in. It can be, for instance, such as short-time Fourier transform (STFT), the continuous wavelet transform (CWT), the synchrosqueezing transform (SST), or the reassignment (RS). Here, F typically denotes the size of the representation along the frequency axis. Due to the boundary effects, the representation $\mathscr{F}_N(\mathbf{x})$ shows undesired patterns when approaching its edges.For example, the instantaneous frequencies highlighted by the SST can be blurred near that edges. To limit these phenomena, we apply the representation to the estimated extended signal $\tilde{\mathbf{x}}$. This strategy moves the boundary effects out of the time interval $I = [0, \frac{N-1}{f_{\text{s}}}]$. Finally, the boundary-effects insensitive representation $\mathscr{F}_N^{\text{ext}} : \mathbb{R}^N \to \mathbb{R}^{F \times N}$ of $\mathbf{x}$ is given for all $f \in \{0, \cdots, F-1\}$, $n \in \{0, \cdots, N-1\}$ by:

$$\mathscr{F}_N^{\text{ext}}(\mathbf{x})[f, n] = \mathscr{F}_{N+2L}(\tilde{\mathbf{x}})[f, L+n] \ . \tag{9}$$

This amounts to restricting the representation $\mathscr{F}_{N+2L}(\tilde{\mathbf{x}})$ to the original measurement interval of $\mathbf{x}$. For the sake of simplicity, we denotes the restriction operator by $\mathcal{R}$, where $\mathcal{R} : \mathbb{R}^{F \times (N+2L)} \to \mathbb{R}^{F \times N}$. Consequently, we have:

$$\mathscr{F}_N^{\text{ext}}(\mathbf{x}) = \mathcal{R}\left(\mathscr{F}_{N+2L}(\tilde{\mathbf{x}})\right) \ .$$

C. Global algorithm

Finally, the global procedure we implement to reduce boundary effects on windowing-based representations is summarized by the pseudo-code of Algorithm 2.

Algorithm 2 Tackling boundary effects. $\mathbf{F}_{\mathbf{x}} = \text{BoundEffRed}(\mathbf{x}, M, K, L, \mathscr{F})$

Inputs: $\mathbf{x}$, M, K, L, $\mathscr{F}$

Forecasting step.
- Signal extension: $\tilde{\mathbf{x}} = \text{SigExt}(\mathbf{x})$.

Representation step.
- Representation evaluation: $\mathscr{F}_{N+2L}(\tilde{\mathbf{x}})$.
- Restriction of $\mathscr{F}_{N+2L}(\tilde{\mathbf{x}})$ to the central time interval (see (9)) to obtain $\mathbf{F}_{\mathbf{x}} = \mathscr{F}_N^{\text{ext}}(\mathbf{x})$.

Output: Signal representation $\mathbf{F}_{\mathbf{x}}$

III. Theoretical performance

A. Signal model

We model the deterministic part of the observed signal as a multicomponent harmonic signal, that is a sum of sine waves. Then:

$$\mathbf{z}[n] = \sum_{j=1}^{J} \Omega_j \cos\left(2\pi f_j \frac{n}{f_\mathsf{s}}\right) \ , \tag{10}$$

where J denotes the number of components, Ω_j the amplitude of the j-th component, and f_j its frequency.

Remark 2. *For the sake of simplicity, we make an additional assumption on the frequencies of each component. We assume that for all $j \in \{1, \dots, J\}$:*

$$\exists\, p_j, p_j' \in \mathbb{N}^* : \quad f_j = \frac{p_j}{M} f_\mathsf{s} = \frac{p_j'}{K} f_\mathsf{s} \ . \tag{11}$$

In addition, the observed signal is assumed to be corrupted by an additive Gaussian white noise. Therefore, the measured discrete signal $\mathbf{x}$ is written as:

$$\mathbf{x} = \mathbf{z} + \sigma \mathbf{w} \ , \tag{12}$$

where $\mathbf{z}$ follows model (10), $\mathbf{w}$ is a Gaussian white noise, whose variance is normalized to one. Thus, σ^2 denotes the variance of the additive noise $\sigma\mathbf{w}$.

B. Forecasting error

On the forecasting interval, we decompose the estimated signal $\tilde{\mathbf{x}}$ as follows:

$$\tilde{\mathbf{x}}[n] = \mathbf{z}[n] + \boldsymbol{\epsilon}[n] \ , \tag{13}$$

where $\boldsymbol{\epsilon}$ is a the forecasting error. When $n \in I = \{0, \dots, N-1\}$, this error is only containing the measurement noise, that is $\boldsymbol{\epsilon}[n] = \sigma \mathbf{w}[n]$. Outside the interval I, the importance of the forecasting error $\boldsymbol{\epsilon}$ is also affected by the loss of information resulting from the linearization of dynamical model we consider in (3). To evaluate the actual behavior of the forward forecasting error $\boldsymbol{\epsilon}[n]$ when $n \geq N$, we determine its two first moments.

1) The mean, which is also the estimation bias, is such that:

$$\begin{aligned} \boldsymbol{\mu}[n] &\triangleq \mathbb{E}\{\boldsymbol{\epsilon}[n]\} \\ &= \mathbb{E}\{\tilde{\mathbf{x}}[n]\} - \mathbf{z}[n] \ . \end{aligned}$$

Given the forecasting strategy, we have $\boldsymbol{\mu}[n] = 0$ when $n \in I$ and:

$$\boldsymbol{\mu}[n] = \mathbb{E}\{\boldsymbol{\alpha}^{(\ell)}\}\mathbf{z}_K + \sigma \mathbb{E}\{\boldsymbol{\alpha}^{(n-N+1)}\mathbf{w}_K\} - \mathbf{z}[n] \tag{14}$$

when $n \geq N$.

2) The covariance is given by:

$$\begin{aligned} \boldsymbol{\gamma}[n,n'] &\triangleq \mathbb{E}\{\left(\boldsymbol{\epsilon}[n] - \boldsymbol{\mu}[n]\right)\left(\boldsymbol{\epsilon}[n'] - \boldsymbol{\mu}[n']\right)\} \\ &= \mathbb{E}\{\tilde{\mathbf{x}}[n]\tilde{\mathbf{x}}[n']\} - \mathbf{z}[n]\mathbf{z}[n'] - \boldsymbol{\mu}[n]\mathbf{z}[n'] \\ &\quad - \boldsymbol{\mu}[n']\mathbf{z}[n] - \boldsymbol{\mu}[n]\boldsymbol{\mu}[n'] \ . \end{aligned}$$

Thus by definition of the noise, we have $\boldsymbol{\gamma}[n,n'] = \sigma^2 \delta_{n,n'}$ when $(n,n') \in I^2$. When $n \geq N$, let us denote $\ell = n - N + 1$. Then, we have two cases.

(i) If $n' \in I$:

$$\begin{aligned} \boldsymbol{\gamma}[n,n'] = \sigma \mathbb{E}\{\mathbf{w}[n']\boldsymbol{\alpha}^{(\ell)}\}\mathbf{z}_K + \sigma^2 \mathbb{E}\{\mathbf{w}[n']\boldsymbol{\alpha}^{(\ell)}\mathbf{w}_K\} \\ - \mathbf{z}[n]\boldsymbol{\mu}[n'] - \boldsymbol{\mu}[n]\boldsymbol{\mu}[n'] \ . \end{aligned}$$

(ii) If $n' = N - 1 + \lambda \geq N$:

$$\begin{aligned} \boldsymbol{\gamma}[n,n'] = \mathbf{z}_K^T \mathbb{E}\left\{{\boldsymbol{\alpha}^{(\ell)}}^T \boldsymbol{\alpha}^{(\lambda)}\right\}\mathbf{z}_K + \sigma \mathbb{E}\{\boldsymbol{\alpha}^{(\ell)}\mathbf{w}_K\boldsymbol{\alpha}^{(\lambda)}\}\mathbf{z}_K \\ + \sigma \mathbb{E}\{\boldsymbol{\alpha}^{(\lambda)}\mathbf{w}_K\boldsymbol{\alpha}^{(\ell)}\}\mathbf{z}_K + \sigma^2 \mathbb{E}\{\boldsymbol{\alpha}^{(\ell)}\mathbf{w}_K\boldsymbol{\alpha}^{(\lambda)}\mathbf{w}_K\} \\ - \mathbf{z}[n]\mathbf{z}[n'] - \mathbf{z}[n]\boldsymbol{\mu}[n'] - \mathbf{z}[n]\boldsymbol{\mu}[n'] - \boldsymbol{\mu}[n]\boldsymbol{\mu}[n'] \ . \end{aligned} \tag{15}$$

Besides, we recall that $\boldsymbol{\gamma}[n,n'] = \boldsymbol{\gamma}[n',n]$.

Expressions (14) and (15) show that these quantities depend on the behavior of the forecasting random vector $\boldsymbol{\alpha}^{(\ell)}$. First, notice that the forecasting matrix $\tilde{\mathbf{A}}$ would ideally take the form:

$$\mathbf{A}_0 = \left(\frac{1}{K}\mathbf{Z}'\mathbf{Z}^T + \sigma^2\mathbf{D}\right)\left(\frac{1}{K}\mathbf{Z}\mathbf{Z}^T + \sigma^2\mathbf{I}\right)^{-1} \ , \tag{16}$$

with $\mathbf{Z} = \begin{pmatrix} \mathbf{z}_0 & \cdots & \mathbf{z}_{K-1} \end{pmatrix}$ and $\mathbf{Z}' = \begin{pmatrix} \mathbf{z}_1 & \cdots & \mathbf{z}_K \end{pmatrix}$, where $\mathbf{z}_k$ is the k-th sub-signal extracted from $\mathbf{z}$ in the same way as $\mathbf{x}_k$ is defined from $\mathbf{x}$ in (2). Besides, $\mathbf{D} \in \mathbb{R}^{M \times M}$ is the Toeplitz matrix such that:

$$\mathbf{D}[m,m'] = \delta_{m+1,m'} \ .$$

Lemma 1 specify the asymptotic behavior of the forecasting vector $\boldsymbol{\alpha}^{(\ell)}$ when the dataset size K is great.

Lemma 1. *Let $\mathbf{x}$ be a random vector defined by* (12)*. Let $\boldsymbol{\alpha}^{(\ell)}$ be the associated forecasting vector for the estimation of $\mathbf{x}[N-1+\ell]$, given by* (6) *and obtained from the least square estimation* (8)*. Let $\boldsymbol{\alpha}_0^{(\ell)}$ be the last row of the ideal forecasting matrix $\mathbf{A}_0^\ell$. Let $\mathbf{h}^{(\ell)}$ be the error vector given by:*

$$\mathbf{h}^{(\ell)} = \boldsymbol{\alpha}^{(\ell)} - \boldsymbol{\alpha}_0^{(\ell)} \ .$$

Then, the random vector $\mathbf{h}^{(\ell)}$ converges in law to a zero-mean Gaussian random vector when $K \to \infty$, and we have:

$$\sqrt{K}\,\mathbf{h}^{(\ell)} \xrightarrow[K\to\infty]{\mathcal{D}} \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Gamma}^{(\ell,\ell)}\right) \ , \tag{17}$$

with $\boldsymbol{\Gamma}^{(\ell,\ell)} = {\mathbf{F}^{(\ell)}}^T \boldsymbol{\Gamma}_0 \mathbf{F}^{(\ell)}$, where $\boldsymbol{\Gamma}_0 \in \mathbb{R}^{M(M+1)\times M(M+1)}$ is a covariance matrix and $\mathbf{F}^{(\ell)} \in \mathbb{R}^{M(M+1)\times M}$ is a Jacobian matrix. The expressions of these two matrices do not depend on K or σ.

Proof. See the Supplementary Material. The proof is based on the multivariate delta method (see paragraph 7.2 in [9]), which allows to asymptotically approximate a random vector normal as a Gaussian random vector. ∎

Consequently, the covariance between $\sqrt{K}\mathbf{h}^{(\ell)}$ and $\sqrt{K}\mathbf{h}^{(\lambda)}$ remains bounded, i.e.:

$$K\,\mathbb{E}\left\{{\mathbf{h}^{(\ell)}}^T \mathbf{h}^{(\lambda)}\right\} \xrightarrow[K\to\infty]{} \boldsymbol{\Gamma}^{(\ell,\lambda)} = {\mathbf{F}^{(\ell)}}^T \boldsymbol{\Gamma}_0 \mathbf{F}^{(\lambda)} \ .$$

Theorem 1. *Let $\mathbf{x} \in \mathbb{R}^N$ be a discrete-time random signal following model (12). Let $\tilde{\mathbf{x}}$ denotes its forecasting, obtained using the extension Algorithm 1. Let $n \geq N$ be a sample index. Then, the first-order moment of the forecasting error $\epsilon[n]$ in (13) is approximated by:*

$$\boldsymbol{\mu}[n] \underset{K\to\infty}{\sim} o(\sigma^2) \tag{18}$$

Its second-order moment $\boldsymbol{\gamma}[n,n']$ verify the following approximation equations:

(i) if $n' \in I = \{0,\ldots,N-1\}$:

$$\boldsymbol{\gamma}[n,n'] \underset{K\to\infty}{\sim} \sigma^2 \boldsymbol{\alpha}_0^{(n-N-1)}[m-(N-M)]\mathbb{1}_{(m\geq N-M)} \tag{19}$$

(ii) if $n' \geq N$:

$$\boldsymbol{\gamma}[n,n'] \underset{K\to\infty}{\sim} \frac{1}{K}\mathbf{z}_K^T \boldsymbol{\Gamma}^{(l,\lambda)} \mathbf{z}_K + \frac{\sigma^2}{K}\mathrm{Tr}\left(\boldsymbol{\Gamma}^{(\ell,\lambda)}\right) + \sigma^2 \left\langle \boldsymbol{\alpha}_0^{(\ell)}, \boldsymbol{\alpha}_0^{(\lambda)} \right\rangle, \tag{20}$$

where $\ell = n - N + 1$ and $\lambda = n' - N + 1$.

Proof. See the Supplementary Material. The proof is mainly based on the results provided by Lemma 1, combined with the Isserlis' theorem [10], which provides a formula for the computation of higher-order moments of Gaussian random variables. ∎

Ideally, the forecasting error would behave like the measurement noise $\sigma\mathbf{w}$, i.e. a zero-mean noise whose variance is of the order of σ^2. Theorem 1 shows that the forecasting error is asymptotically unbiased. Concerning the covariance of the forecasting error, although equations (19) and (20) are not easily readable, one can evaluate the dependence of the variance in function of the tuning parameters, that are adjusted by the user. Let us focus on the forecasting error variance $\gamma[n,n]$ when $n \geq N$. First, as expected, the variance increases linearly with the noise variance σ^2. Second, it asymptotically depends linearly on the ratio $\frac{1}{K}$. This shows the need to use a sufficiently large dataset to obtain an accurate forecast. Third, the dependency on the sub-signals lengths M and the forecasting index $\ell = n - N + 1$ is hidden in the expression of the covariance matrix $\boldsymbol{\Gamma}^{(\ell,\ell)}$. We discuss this dependency in more detail in section IV-A1.

Remark 3 (Adaptive Harmonic Model)**.** *One can extend the previous result to the case where the instantaneous frequencies and amplitudes of the components of the deterministic part of the observed signal are slowly varying. We therefore handle the adaptive harmonic model which, in its is continuous-time version, takes the following form:*

$$z(t) = \sum_{j=1}^{J} a_j(t)\cos(2\pi\phi_j(t)) \ , \tag{21}$$

where a_j and ϕ_j' are smooth function. In the case, the forecasting error is additionally sensitive to the speed of variation of the instantaneous amplitudes a_j and frequencies ϕ_j'.

C. Performance of the boundary effects reduction

IV. Numerical results

A. Evaluation the forecasting performance

In that section, we first evaluate the quality of the forecasting step and compare it the the theoretical results provided by Theorem 1. The level of the forecasting error one depends on at least two parameters:

- The noise variance σ^2.
- The size of the training dataset K.

In subsections IV-A1 and IV-A2, we study the influence of these parameters. A comparison with the theoretical results of section III is also available.

1) Sum of sine waves: We proved that the linear dynamic model is sufficient to catch the dynamical behavior of signals taking the form (10). In order to validate this theoretical result, we apply the forecasting Algorithm 1 to a large number of realizations of the random vector $\mathbf{x}$ following the model (12), and such that the deterministic component $\mathbf{z}$ takes the form:

$$\mathbf{z}[n] = \cos\left(2\pi p_1 \frac{n}{M}\right) + R\cos\left(2\pi p_2 \frac{n}{M}\right), \quad \forall n \in \{1,\ldots,N\},$$

with $N = 10^4$, $M = 150$, $p_1 = 10$, $p_2 = 33$ and $R = 1.4$. Besides, the additive noise is chosen to be Gaussian: $\mathbf{w} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$.

a) Influence of the noise variance σ^2: Here, the size of the training dataset is set to $K = 450$. Then, the forecasting algorithm is run on 1000 realizations on the discrete signal $\mathbf{x}$ for three different values of σ, logarithmically spaced from 10^{-3} to 10^{-1}. For each of these values, we determine the experimental bias $\mu_{xp}[N-1\ell]$ and variance $\gamma_{xp}[N-1+\ell, N-1+\ell]$ in function of the forecasting sample index ℓ (going from 1 to $L = 100$).

The experimental results show that the bias is neither depending on the noise variance σ^2 nor the forecasting length ℓ. Indeed, independently of *sigma*, we always have $\mu_{xp}[N-1+\ell] \in [-0.03\sigma, 0.03\sigma]$, which is negligible with respect to the magnitude of $\mathbf{z}$. This result confirms the theoretical result (18).

On Figure 2, we display the experimental variance $\gamma_{xp}[N-1+\ell, N-1+\ell]$ for each value of σ (+ markers). The associated theoretical asymptotic forecasting variance (20) is also displayed in solid line. As expected, this result highlights the fact that the forecasting variance increases linearly with respect to σ^2. More surprisingly, this result shows that the forecasting variance slightly decreases with ℓ, what is counterintuitive. It should be noted that, contrary to what expression (20) suggests, smaller values of σ do not cause a decrease of the experimental variance, but an increase. This comes from the fact that when σ is small, the matrix $\mathbf{X}\mathbf{X}^T$ becomes ill-conditioned. The calculation of the forecasting matrix $\tilde{\mathbf{A}}$ in (8) is then strongly disturbed.

b) Influence of the training dataset size K: Here, the noise variance σ is set to $\sigma = 10^{-2}$. Then, the forecasting algorithm is run on 3000 realizations on the discrete signal $\mathbf{x}$ for three different values of K, logarithmically

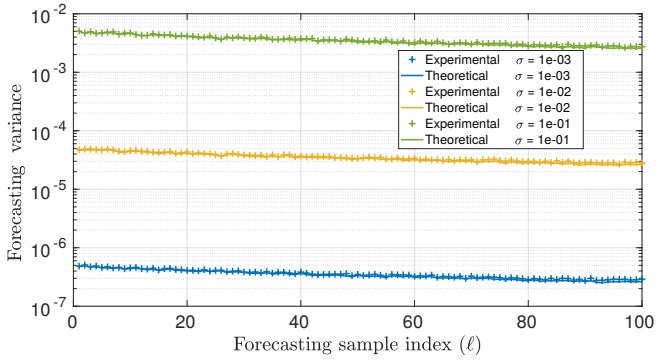

Figure 2: Evolution of the experimental and theoretical forecasting variance in function of the forecasting sample index for different values of σ.

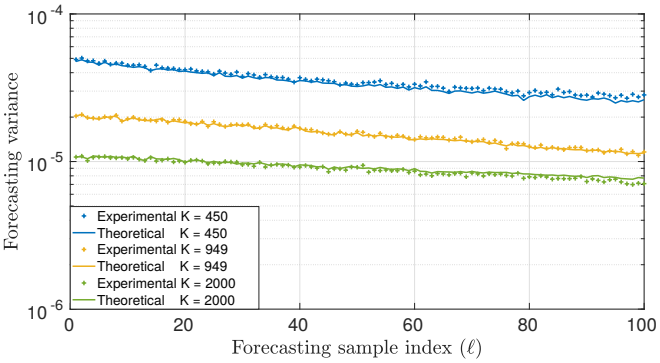

Figure 3: Evolution of the experimental and theoretical forecasting variance in function of the forecasting sample index for different values of K.

spaced from 4.5×10^2 to 2×10^3. For each of these values, we determine the experimental bias $\mu_{xp}[N-1+\ell]$ and variance $\gamma_{xp}[N-1+\ell, N-1+\ell]$ in function of the forecasting sample index ℓ (going from 1 to 500).

As in the previous study, the experimental bias vanishes when K increases, what confirms the approximation result (18). Besides, the experimental variance is displayed on Figure 3 (+ markers), and compared with the associated theoretical variance (20). Each color corresponds to these experimental results obtained for a given value of K. This result validates the asymptotic behavior provided by (20), and we can show that, as soon as K is sufficiently high, the product $K\gamma[\ell, \ell]$ is approximately independent of K.

c) Conclusion: Both previous experimental results combined with the theoretical asymptotic equation (20) allow us to describe the influence of the noise variance and the size of the training dataset on the variance of the forecasting noise, which is empirically summarized as follows:

$$\gamma[N-1+\ell, N-1+\ell] \underset{K\to\infty}{\approx} \frac{\sigma^2}{K} g[\ell]. \qquad (22)$$

where g is a bounded positive function. The empirical result is coherent with the theoretical result provided by Theorem 1.

This study neglects the analysis of the influence of the parameter M, whose influence on the value of the experimental variance is numerically not significant as long as $M \ll 2K$. The choice of this parameter is especially crucial when the deterministic component of the signal is no longer stationary. The adaptive harmonic model, discussed below, is an example.

2) Adaptive harmonic model: We now consider a signal whose instantaneous frequencies and amplitudes of its components vary over time. The deterministic component $\mathbf{z}$ of the random vector $\mathbf{x}$ (constructed following the model (12)) takes the following form, for all $n \in \{1, \ldots, N\}$:

$$\mathbf{x}[n] = \cos\left(2\pi\phi_1[n]\right) + R[n]\cos\left(2\pi\phi_2[n]\right) \ ,$$

where the instantaneous amplitude R is given by:

$$R[n] = 1.4 + 0.2\cos\left(4\pi\frac{n}{N}\right) \ ,$$

and the instantaneous phases are such that:

$$\phi_1[n] = \frac{p_1}{M}\left(n + \frac{0.01}{2\pi}\cos\left(2\pi\frac{n}{N}\right)\right)$$
$$\phi_2[n] = p_2\frac{n}{M} + \frac{20}{2Nf_s}n^2$$

Besides, the additive noise is chosen to be Gaussian: $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Numerically, we take: $N = 10^4$, $M = 750$, $p_1 = 10$, $p_2 = 23$.

To highlight the fact that the linear dynamical model is sufficient to catch most of the dynamical behavior of signals following the AHM, we compare the performance of the Algorithm 2 with reference forecasting algorithm that could be used for extending such signals. These methods are:

- The *Extended Dynamic Mode Decomposition (EDMD)* has been developped by Williams *et al.* [5]. The proposed algorithm is a way to obtain an approximation of the so-called Koopman operator of the observed system, which theoretically allows to catch dynamic of nonlinear systems [11].
- The *Gaussian Process Regression (GPR)* [6] is a method relying on a probabilistic dynamical model. That one is based on the Gaussian process structure, and therefore offer more flexibility in the type of dynamic that could be modeled than the linear model (3).
- The *Trigonometric , Box-Cox transformation, ARMA errors, Trend and Seasonal components (TBATS)* method [6] is based on a classical decomposition of times series into a trend, a seasonal and an ARMA components, with a specific dynamic for the seasonal component. This model demands the estimation of numerous parameters and, by implication, may be slow.

To quantify the global quality (i.e. not depending on ℓ) of the forecasting approaches, we evaluate the Ex-

Table I: Performance of the extension methods.

Algorithm	MSE		Computing time (sec.)
	Mean	Standard deviation	
SigExt	1.433×10^{-3}	4.361×10^{-4}	0.15
EDMD	3.076×10^{-2}	8.095×10^{-2}	2.53
GPR	1.436×10^{-3}	4.346×10^{-4}	146.33
TBATS	X	X	X

perimental Mean Square Error $\mathrm{MSE}_{\mathrm{xp}}(\tilde{\mathbf{x}})$ of the forward forecast extended signals, namely:

$$\mathrm{MSE}_{\mathrm{xp}}(\tilde{\mathbf{x}}) = \frac{1}{L}\|\tilde{\mathbf{x}} - \mathbf{x}^{\mathrm{ext}}\|^2 \qquad (23)$$
$$= \frac{1}{L}\sum_{\ell=1}^{L} \boldsymbol{\mu}_{\mathrm{xp}}[N-1+\ell]^2 + \boldsymbol{\gamma}_{\mathrm{xp}}[N-1+\ell, N-1+\ell].$$

where $\mathbf{x}^{\mathrm{ext}}$ is the ground-truth extended signal, that is: $\mathbf{x}^{\mathrm{ext}} = \begin{pmatrix}\mathbf{x}[-L] & \cdots & \mathbf{x}[N-1+L]\end{pmatrix}$. Then, as long as the bias $\boldsymbol{\mu}[N-1+\ell]$ and the variance $\boldsymbol{\gamma}[N-1+\ell, N-1+\ell]$ of the forecasting estimator remain small for all ℓ, the MSE takes small values either. Corresponding results are given in Table I. They show that the naive extension we propose gives satisfying results, even though the other methods, more sophisticated, give MSE values that are somewhat smaller. Nevertheless, a major limit of those methods is the computing time they require, which prevent them from being used to exploit real-time data. Thus, SigExt is the extension method that optimize the trade-off between the forecasting quality and the computing time. That is why, it is implemented in our algorithm for the reduction of boundary effects.

B. Evaluation of the quality of the boundary effect reduction

1) Metrics: The quality of the boundary effect reduction must be evaluated directly on the time-frequency representation. To that aim, we compare the obtained representation to the optimal representation $\mathscr{F}_N^{\mathrm{opt}}(\mathbf{x})$. The optimal representation is defined as the restriction of the representation of the ground-truth extended signal $\mathbf{x}^{\mathrm{ext}}$. Therefore, we have:

$$\mathscr{F}_N^{\mathrm{opt}}(\mathbf{x}) = \mathcal{R}\left(\mathscr{F}_{N+2L}(\mathbf{x}^{\mathrm{ext}})\right) \ .$$

In the aim of comparing the different techniques, we uses a criterion, proposed in [12], that quantify the distance between a given time-frequency representation and the optimal one. It is built in analogy with the optimal transport distance, which enables quantifying the distance between two probability density functions. Let us generically denote a time frequency representation $\mathscr{Q}$. Then, for t fixed, we consider the following pseudo-probability density function:

$$p^t_{\mathscr{Q}}(\xi) = \frac{|\mathscr{Q}(\xi,t)|^2}{\int_{\mathbb{R}} |\mathscr{Q}(\nu,t)|^2 \mathrm{d}\nu} \ . \qquad (24)$$

Then, at each instant t, we can then determine the optimal transport distance d_t between the two pseudo densities. It is given by the L^1 norm of the difference between the associated distribution functions. In other words, we have:

$$d_t(\mathscr{Q}, \mathscr{F}_0) = \int_{\mathbb{R}} \left| \tilde{P}^t_{\mathscr{Q}}(\xi) - P^t_{\mathscr{F}_0}(\xi) \right| \mathrm{d}\xi \ ,$$

where $P^t_{\mathscr{Q}}(\xi) = \int_{-\infty}^{\xi} p^t_{\mathscr{Q}}(\nu)\mathrm{d}\nu$ and $\tilde{P}^t_{\mathscr{F}_0}(\xi) = \int_{-\infty}^{\xi} \tilde{p}^t_{\mathscr{F}_0}(\nu)\mathrm{d}\nu$. Finally, the distance between the two time-frequency representations is obtained by averaging all the optimal transport distances with respect to time:

$$D(\mathscr{Q}, \mathscr{F}_0) = \frac{1}{|I|}\int_I d_t(\mathscr{Q}, \mathscr{F}_0)\mathrm{d}t \ . \qquad (25)$$

The *Optimal Transport Distance* (OTD) quantifies the proximity between the estimated and actual instantaneous frequencies while favouring the sparsity of the estimated time-frequency representation.

Let us evaluate the quality of the boundary effects reduction on biomedical signals.

2) Respiratory signal: We first consider a 50 minutes-long respiratory signal. This signal is sampled at $f_s = 100$ Hz. A zoom on a small portion of the signal is displayed in Figure 4.

From that large signal, we build a dataset of 48 non-overlapping signals of 60 seconds, i.e. $N = 6000$. On each of these pieces of signal, we implement the forecasting method introduced in section IV-A2, including the SigExt method detailed in Algorithm 1. However, the TBATS extension method in not implemented here because of its excessive computing time. The extensions of 7 seconds-long on each boundary, corresponding to $L = 700$. Thus, in order to catch slowly varying dynamical behaviors, the size of the training signal M is chosen so that $M = \lfloor 1.5L \rfloor$. As a result of section IV-A1, we take: $K = \lfloor 2.5M \rfloor$. The average and the median of the MSE (23) with respect to the simulations are given in Table II. The averaged MSE of SigExt method is

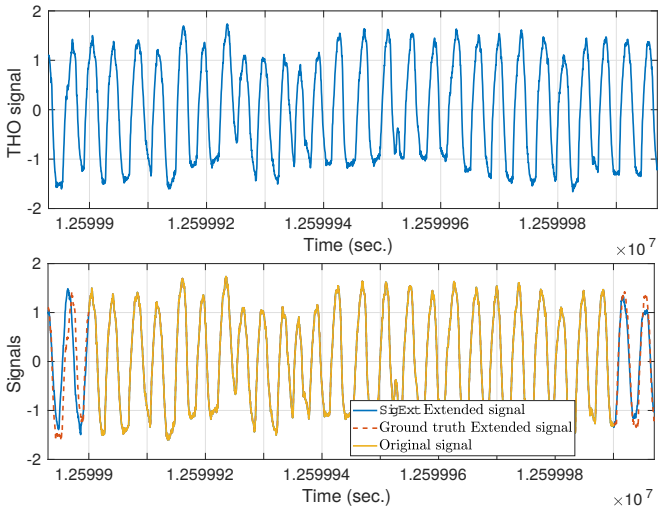

Figure 4: Zoom on the respiratory signal. Top: original measured signal. Extended signal obtained by SigExt forecasting superimposed with the ground truth signal.

Table II: MSE of the respiratory signal extensions.

Algorithm	MSE	
	Mean	Median
SigExt	1.704×10^{1}	7.346×10^{-1}
EDMD	2.172×10^{-2}	7.349×10^{-1}
GPR	4.476×10^{-2}	1.998×10^{-2}

Table III: Respiratory signal: OTD of the boundary effects reduction methods on diverse TFR.

Extension method	Time-Frequency Representation		
	STFT	SST	RS
Without extension	2.16×10^{-2}	5.26×10^{-3}	3.07×10^{-2}
SigExt	1.72×10^{-2}	4.00×10^{-3}	2.43×10^{-2}
EDMD	1.75×10^{-2}	4.23×10^{-3}	2.45×10^{-2}
GPR	1.84×10^{-2}	4.10×10^{-3}	2.48×10^{-2}

overwhelmingly higher than the averaged MSE of the other methods. This result is caused by the presence of outliers in some extensions by SigExt. Indeed, the value median of the MSE is of the same order than those of the other methods. These outliers are probably due to the presence of pulse in the respiratory signal. That ones make the adaptive harmonic model temporary irrelevant, and break the validity of the linear dynamical model (3) used to extend the signal. That is why SigExt is less robust to the vanishing of oscillations that the GPR or EDMD extensions.

Then, we apply the boundary effects reduction method BoundEffRed (Algorithm 2) on these extensions for diverse time-frequency representations (STFT, SST and RS). In Table III, the results are compared in terms of OTD. They are also compared with the strategy consisting in a zero-padding extension of the signal. Even though the performance of the forecasting Algorithm 1 is somehow moderate, the boundary effects can be reduced dramatically on the time-frequency representations. Notice that the extension length L has been set accordingly to the window length used by the time-frequency analysis tool. For instance, here the window length we use to evaluate the STFT is of 1500 samples. To prevent the STFT from being sensitive to the boundary effect, we set $L = 750$. In this way, when evaluating the spectral content of the signal near its boundaries, the analysis is not limited by a lack of information all along the window support. From now on, all results are given for L at equal to the half of the width of the window used in the time-frequency transform.

3) Photoplethysmogram: We perform a study similar to the previous one on a 640 second-long photoplethysmogram (PPG) signal, sampled at $f_s = 125$ Hz. A 32 second-long piece of this signal is displayed on the top of Figure 5. The estimated 2-second extension obtained by SigExt on both boundary of this signal is superimposed to the ground-truth signal in the bottom of Figure 5.

We divide the signal into 32-second long pieces, and apply Algorithm 2 on each piece. We provide in Table IV the OTD to the optimal time-frequency representation averaged over the signals. For all the considered time-

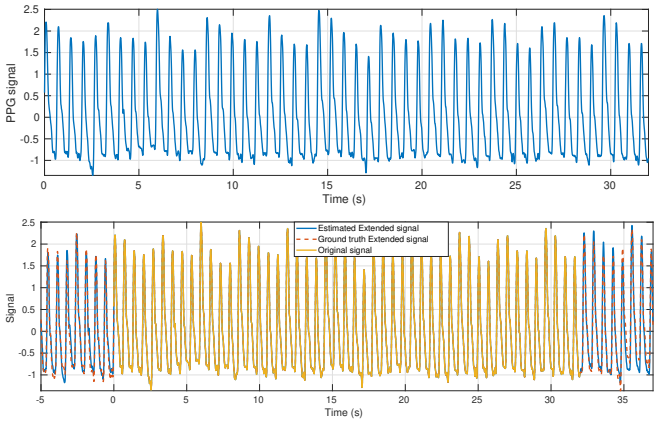

Figure 5: PPG signal. Top: original measured signal. Extended signal obtained by forecasting superimposed with the ground truth signal.

Table IV: Performance of the Boundary effect reduction on TFR.

Extension method	Time-Frequency Representation		
	STFT	SST	RS
Without extension	2.52×10^{-2}	9.41×10^{-2}	1.03×10^{-1}
SigExt	1.22×10^{-3}	7.31×10^{-2}	1.11×10^{-1}
EDMD	9.83×10^{-4}	5.59×10^{-2}	9.80×10^{-2}
GPR	1.14×10^{-3}	7.02×10^{-2}	1.07×10^{-1}

frequency representations, the results clearly shows that our algorithm reduce the influence of the boundary effect. This highlights the ability of our approach to limit the distortion due the boundary effect and provide a more accurate representations. Even though the SIGEXT extension yields time-frequency representations slightly more sensitive to boundary effects that the extensions given by EDMD or GPR, it is the only technique that allows a real-time implementation.

On Fig. 6, we display the SST resulting from the BoundEffRed strategy, applied to the same portion of PPG than what is used to display Fig. 5. We clearly observe an improvement of the quality of the SST near boundaries. Indeed, the blurring visible when zooming on the right boundary of the SST has almost vanished (see bottom of Fig. 6). The real-time tracking of the instantaneous frequencies contained in the measured signal is therefore largely facilitated.

V. CONCLUSION

The MATLAB code and datasets used to produce the numerical results of this paper are available online[1].

REFERENCES

[1] M. Akay, *Detection and Estimation Methods for Biomedical Signals*, 1st ed. USA: Academic Press, Inc., 1996.

[2] S. Mallat, *A Wavelet Tour of Signal Processing (Second Edition)*. San Diego: Academic Press, 1999.

[1]https://github.com/AdMeynard/BoundaryEffectsReduction

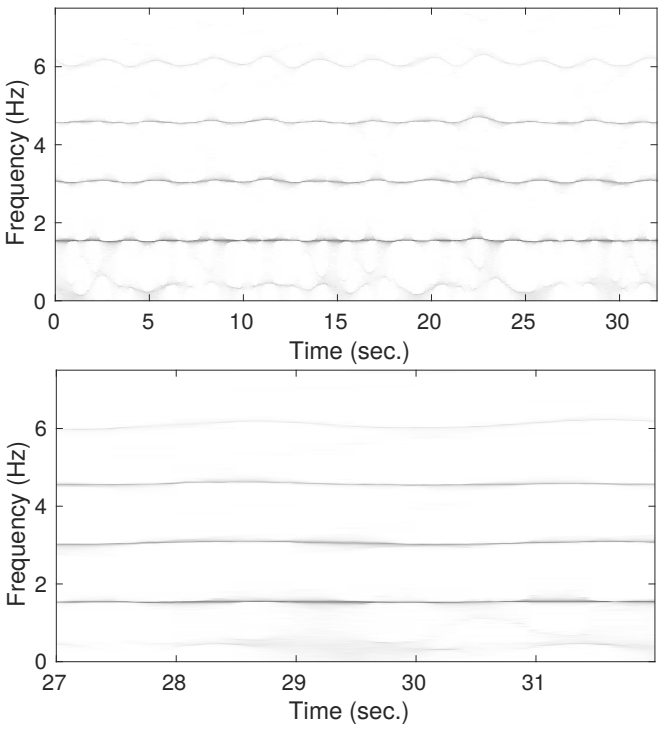

Figure 6: Result of BoundEffRed on the synchrosqueezing transform of a PPG (top) with a zoom on its right boundary (bottom).

[3] F. Auger, P. Flandrin, Y.-T. Lin, S. McLaughlin, S. Meignen, T. Oberlin, and H.-T. Wu, "Time-frequency reassignment and synchrosqueezing: An overview," *IEEE Signal Processing Magazine*, vol. 30, no. 6, pp. 32–41, 2013.

[4] I. Kharitonenko, X. Zhang, and S. Twelves, "A wavelet transform with point-symmetric extension at tile boundaries," *IEEE Transactions on Image Processing*, vol. 11, no. 12, pp. 1357–1364, 2002.

[5] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A Data-Driven Approximation of the Koopman Operator: Extending Dynamic Mode Decomposition," *Journal of Nonlinear Science*, vol. 25, pp. 1307–1346, 2015.

[6] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, ser. Adaptive Computation and Machine Learning series. The MIT Press, 2006.

[7] A. M. D. Livera, R. J. Hyndman, and R. D. Snyder, "Forecasting time series with complex seasonal patterns using exponential smoothing," *Journal of the American Statistical Association*, vol. 106, no. 496, pp. 1513–1527, 2011. [Online]. Available: https://doi.org/10.1198/jasa.2011.tm09771

[8] H. Su, Q. Liu, and J. Li, "Adaptive approach for boundary effects reduction in rotating machine signals analysis," *Advances in Mechanical Engineering*, vol. 5, p. 540172, 2013. [Online]. Available: https://doi.org/10.1155/2013/540172

[9] J. Alho and B. Spencer, *Statistical Demography and Forecasting*, ser. Springer Series in Statistics. Springer-Verlag New York, 2005.

[10] L. Isserlis, "On a formula for the product-moment coefficient of any order of a normal frequency distribution in any number of variables," *Biometrika*, vol. 12, no. 1-2, pp. 134–139, 11 1918. [Online]. Available: https://doi.org/10.1093/biomet/12.1-2.134

[11] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149 – 160, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S000510981830133X

[12] I. Daubechies, Y. G. Wang, and H.-T. Wu, "ConceFT: concentration of frequency and time via a multitapered synchrosqueezed transform," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150193, 2016. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2015.0193