An Efficient Forecasting Approach to Reduce Boundary Effects in Real-Time Time-Frequency Analysis

Adrien Meynard, Hau-Tieng Wu

Abstract—Time-frequency (TF) representations of time series are intrinsically subject to the boundary effects. As a result, the structures of signals that are highlighted by the representations are garbled when approaching the boundaries of the TF domain. In this paper, for the purpose of real-time TF information acquisition of nonstationary oscillatory time series, we propose a numerically efficient approach for the reduction of such boundary effects. The solution 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 guarantee of the performance of our approach. Following a numerical verification of the algorithmic performance of our approach, we validate it by implementing it on biomedical signals.

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

I. INTRODUCTION

IN any digital acquisition system, the study and the interpretation of the measured signals generally require an analysis tool, which enables researchers to point out the useful characteristics of the signal. The need for signal analysis arises from various signals, ranging from audio [1], [2], mechanical [3], or biomedical signals [4]. For instance, biomedical signals, such as photoplethysmogram (PPG), contain several characteristics, including respiratory rate or blood pressure, that cannot be interpreted from its run-sequence plot in the time domain. An analysis tool would make possible the extraction of these useful characteristics.

Usually, the measured signals exhibit nonstationary behavior, and the observed quantities might be interfered by transient phenomena that can vary rapidly and irregularly. In this paper, we focus on oscillatory time series. These signals might, for example, oscillate fast with large amplitude at one moment, and then oscillate slowly with small amplitude at the next moment. In order to adapt the analysis to nonstationarities, local spectral analysis is generally performed [5], [6]. The short-time Fourier transform [7] (STFT), a typical tool built for this purpose, enables the determination of the local frequency content of a nonstationary signal.

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

Windowing is a common method for performing local analysis. Among many others, STFT [8], continuous wavelet transform (CWT) [9], synchrosqueezing transform (SST) [10], and reassignment [11] (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 localized around the origin and is small with respect to $|I|$. The translation operator is T_τ defined as:

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

Then, the local analysis of x around the instant $\tau \in I$ rely on the evaluation of the following inner 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. Clearly, at these points, half of the information is missing. Consequently, the results of the inner product (1) are distorted. This phenomenon is usually understood as the *boundary effect*. The result of the SST of a PPG is shown in the lower-left corner of Fig. 1 (see section IV-D2 for a comprehensive description). The distortion resulting from boundary effects is clearly visible on the right side of this representation. Indeed, while in the major left part of the image, clear lines stand out, they become blurred as they approach the right boundary of the image. Therefore, estimations of signal characteristics, like instantaneous frequencies [12] or amplitudes, from this TF representation appear to be imprecisely determinable (or even likely to fail) in the vicinity of the boundaries. Moreover, this boundary effect would unavoidably limit applying these window-based TF analysis tools for real-time analysis purposes. It is thus desirable to have a solution to eliminating the boundary effects.

Attempts to minimize the boundary effects generally consists in softening the discontinuity on signal edges. Usually the approaches can be classified into two main classes, choosing a proper analysis window and extending/forecasting the signal. In the first class, judicious choice of analysis window whose support does not interfere with the boundary points can minimize the occurrence of aberrant patterns near the boundaries of
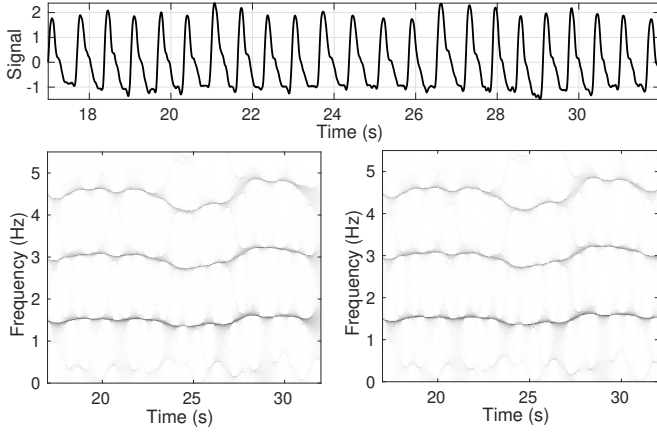
Fig. 1. A segment of PPG signal (top) and the right boundary of a TF representation determined by the SST without extension (bottom left), and the right boundary of a TF representation determined by the SST with the proposed boundary effect reduction algorithm by forecasting (bottom right). The window length for the SST is 12 seconds.

the TF plane, for instance, [13], [14]. Due to the specific relationship of chosen analysis windows and TF analysis tool, these techniques do not make it possible to reduce the boundary effects of all TF representations. Another natural idea, the second class, consists of carrying out a preliminary step of extending, or forecasting, the signal beyond its boundaries. Due to its flexibility, various forecasting schemes have been proposed. For example, there exist simple extension schemes that do not take into account the dynamical behavior of the signal, such as zero-padding, periodic extension, symmetric extension [15], [16], or polynomial extrapolation [17]. There exist extension schemes based on physically relevant dynamical models, such as the Extended Dynamic Mode Decomposition [18] (EDMD) and the Gaussian process regression [19], [20] (GPR). There also exist extension schemes based on stochastic models, such as the Trigonometric, Box-Cox transformation, ARMA errors, Trend and Seasonal components (TBATS) algorithm [21], the dynamic linear models [22]. In the physically relevant dynamical models, the oscillation or trend, are usually modeled as the mean of a stochastic process, while in the stochastic models, the oscillation or trend are modeled by the covariance structure of the stochastic process. It is also possible to consider polynomial regression [23] or modeling the mean by splines [24], kernel functions [25], and wavelets [26], or nonparametric regression [23], to estimate the mean function before forecasting the signal. Neural network models, such as the long short-term memory model [27], is another approach. The above is a far from exhaustive list, and we refer readers with interest to [28] for a friendly monograph on the general forecasting topic. While the model-based extension scheme gives better-extended signals than those simple extension without dynamical models, they generally have a great computational cost.

In this paper, we propose a fast extension algorithm

based on a simple dynamical model that optimizes the trade-off between the extension quality and the computational cost, so that the real-time analysis can be achieved. The algorithm is composed of two steps.

1) *Extend the signal by forecasting it.* The aim is to use a simple 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 than 2Δ, the local analysis near the boundary of I is now possible without lack of information thanks to knowledge brought by the extension.

Thus, assuming that the quality of the extension step is sufficient, the analysis results obtained that way will be less sensitive to the boundary effects than the result of the analysis tool applied directly to the nonextended signal. We claim and prove that forecasting oscillatory signals based on the simple dynamical model combined with the simple least square approach is sufficient for reducing the boundary effects for the TF analysis, or other kernel-based analysis. To our knowledge, such theoretical analysis does not exist in the literature. See the bottom right of Fig. 1 for a snapshot of the result. The main benefit of this simple approach is a numerically efficient solution with a theoretical guarantee for real-time analysis purposes.

The paper is organized in the following way. In section II, we provide an extension method based on a linear dynamic model. We derive the corresponding algorithm for boundary effects reduction. In section III, we show that the dynamic model we consider 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 effects 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 TF representations relies on extending the signal by forecasting it before applying the TF analysis.

We start with the notation. 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

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

where $f_{\mathbf{s}}$ denotes the sampling frequency. Let M and K be two integers such that $M < N$ and $K + M < N$. Then, for all $k \in \{0, \ldots, K-1\}$, we extract from $\mathbf{x} \in \mathbb{R}^N$ the subsignal $\mathbf{x}_k \in \mathbb{R}^M$ given by:

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

These subsignals 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 subsignals 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:

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

The boundary effect reduction algorithm is based on manipulating $\mathbf{X}$ and $\mathbf{Y}$.

The pseudo-code of the proposed real-time algorithm to reduce boundary effects on windowing-based TF representations is shown in Algorithm 1. We coined the algorithm BoundEffRed. Below, we detail the algorithm, particularly the signal extension SigExt in Algorithm 2.

Algorithm 1 Tackling boundary effects of a TF representation in real-time. $\mathbf{F_x} = \mathsf{BoundEffRed}(\mathbf{x}, M, K, L, \mathscr{F})$

Inputs: $\mathbf{x}_N$, M, K, N_0, L, $\mathscr{F}$

while N increases **do**
  Real-time input: $\mathbf{x}_N$

  Forecasting step.
    • Signal extension: $\tilde{\mathbf{x}}_{N+L} = \mathsf{SigExt}(\mathbf{x}_N)$.
  Representation estimation step.
    • Extended representation evaluation: $\mathscr{F}(\tilde{\mathbf{x}}_{N+L})$.
    • Restriction of $\mathscr{F}(\tilde{\mathbf{x}}_{N+L})$ to the current time interval (see (10)) to obtain $\mathbf{F}_\mathbf{x}^{(N)} = \mathscr{F}^{\mathrm{ext}}(\mathbf{x}_N)$.

  Real-time output: Signal representation $\mathbf{F}_\mathbf{x}^{(N)}$
end while

A. Step 1: Extension by Forecasting

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

$$\mathbf{Y} = f(\mathbf{X}) . \quad (3)$$

In a general framework, forecasting means estimating the function f from the observed values taken by the signal, in order to predict its future values. For instance, the dynamic mode decomposition [29], [18] or other more complicated models [20], [21], [22], [27], allow this by setting additional constraints on the behavior of f. We will see, in section III, that considering such a complex dynamic model is not necessary for the study of the oscillatory signals of interest to us. That is why we

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

$$\mathbf{Y} = \mathbf{A}\mathbf{X} , \quad (4)$$

where $\mathbf{A} \in \mathbb{R}^{M \times M}$. In other words, we adopt a classical strategy in the study of dynamical systems, the *linearization of a nonlinear system*, when the system is sufficiently regular. Notice that this linearized dynamical model can be written equivalently in function of the subsignals $\mathbf{x}_k$, as:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k, \;, \forall k \in \{0, \ldots, K-1\} . \quad (5)$$

From the Takens' embedding perspective [30], $\mathbf{X}$ can be viewed as a lag map. Under the manifold assumption of the intrinsic phase space that hosts the dynamics, f describes the dynamics supported on the manifold space, and $\mathbf{A}$ approximates the dynamics on the manifold.

b) Forecasting: 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 $\frac{N-1+\ell}{f_{\mathbf{s}}}$ by:

$$\tilde{\mathbf{x}}[N - 1 + \ell] = \boldsymbol{\alpha}^{(\ell)} \mathbf{x}_K , \quad (6)$$

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

$$\boldsymbol{\alpha}^{(\ell)} = \mathbf{e}_M^T \tilde{\mathbf{A}}^\ell , \quad (7)$$

where $\mathbf{e}_M$ is the unit 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 consider the simple but numerically efficient least square estimator. That is, we solve the following problem:

$$\tilde{\mathbf{A}} = \arg\min_{\boldsymbol{\alpha}} \mathcal{L}(\mathbf{A}) , \quad (8)$$

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

$$\tilde{\mathbf{A}} = \mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1} . \quad (9)$$

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

d) Signal Extension: Since we are building a real-time algorithm, we consider that only the right "side" of the signal has to be extended, the left "side" being fixed since it only concerns the past values of the signal. We therefore construct the extended signal $\tilde{\mathbf{x}} \in \mathbb{R}^{N+L}$ concatenating the observed signal $\mathbf{x}$, and the forward forecasting $\tilde{\mathbf{x}}_{\text{fw}} \in \mathbb{R}^L$. We summarize the extension step in Algorithm 2.

Algorithm 2 Signal extension. $\tilde{\mathbf{x}} = \mathsf{SigExt}(\mathbf{x}, M, K, L)$

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

Forward forecasting.
- Estimation of the matrix $\tilde{\mathbf{A}}$ via equation (9).
- Forecasting $\tilde{\mathbf{x}}_{\text{fw}} \in \mathbb{R}^L$ obtained applying equation (6) with $\ell \in \{1, \ldots, L\}$.

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

B. Step 2: Extended Time-Frequency Representation

Let $\mathscr{F} : \mathbb{R}^N \to \mathbb{C}^{F \times N}$ generically denotes the TF representation of interest to us, which could be, for instance, STFT, CWT, SST, or RS. Here, F typically denotes the size of the discretization along the frequency axis. Due to the boundary effects, the representation $\mathscr{F}(\mathbf{x}_N)$ shows undesired patterns near its edges. To alleviate the boundary effects, we apply the representation to the estimated extended signal $\tilde{\mathbf{x}}$. This strategy moves the boundary effects out of the time interval $I = [0, \frac{N-1}{f_{\mathsf{s}}}]$. Finally, the boundary-effects insensitive representation $\mathscr{F}^{\text{ext}} : \mathbb{R}^N \to \mathbb{C}^{F \times N}$ of $\mathbf{x}_N$ is given for all $\nu \in \{0, \cdots, F-1\}$, $n \in \{0, \cdots, N-1\}$ by:

$$\mathscr{F}^{\text{ext}}(\mathbf{x}_N)[\nu, n] = \mathscr{F}(\tilde{\mathbf{x}}_{N+L})[\nu, n] . \qquad (10)$$

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

$$\mathscr{F}^{\text{ext}}(\mathbf{x}_N) = \mathcal{R}\left(\mathscr{F}(\tilde{\mathbf{x}}_{N+L})\right) . \qquad (11)$$

We call $\mathscr{F}^{\text{ext}}$ the *boundary-free TF representation*.

$\mathbf{F}_{\mathbf{x}}^{(N)} \in \mathbb{C}^{F \times N}$ is the estimation of the boundary-free TF representation at iteration N in Algorithm 1. For numerical purposes, and to make the real-time implementation achievable, we do not perform a full re-estimation of $\mathscr{F}^{\text{ext}}$ at iteration $N + 1$. Instead, the additional knowledge provided by $\mathbf{x}[N+1]$ only influences the values of the last L_{win} columns of $\mathscr{F}^{\text{ext}}(\mathbf{x}_{N+1})$, where L_{win} denotes the window half-length used by the TF representation. Thus, $\mathbf{F}_{\mathbf{x}}^{(N+1)}$ is obtained by the concatenation of the first $N - L_{\text{win}} + 1$ columns of $\mathbf{F}_{\mathbf{x}}^{(N)}$ with the last L_{win} columns of $\mathscr{F}^{\text{ext}}(\mathbf{x}_{N+1})$.

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:

$$\mathbf{z}[n] = \sum_{j=1}^{J} \Omega_j \cos\left(2\pi f_j \frac{n}{f_{\mathsf{s}}} + \varphi_j\right) , \qquad (12)$$

where J denotes the number of components, $\Omega_j > 0$ the amplitude of the j-th component, f_j its frequency, and $\varphi_j \in [0, 2\pi)$ its initial phase. For the sake of simplicity, we make an additional assumption on the frequencies of each component. We assume that for all $j \in \{1, \ldots, J\}$:

$$\exists\, p_j, p'_j \in \mathbb{N}^* : \quad f_j = \frac{p_j}{M} f_{\mathsf{s}} = \frac{p'_j}{K} f_{\mathsf{s}} . \qquad (13)$$

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:

$$\mathbf{x} = \mathbf{z} + \sigma \mathbf{w} , \qquad (14)$$

where $\mathbf{z}$ follows model (12), $\mathbf{w}$ is a Gaussian white noise, whose variance is normalized to one, and $\sigma > 0$. Clearly, 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:

$$\tilde{\mathbf{x}}[n] = \mathbf{z}[n] + \epsilon[n] , \qquad (15)$$

where ϵ is the forecasting error. When $n \in I = \{0, \ldots, N-1\}$, this error contains only the measurement noise, that is $\epsilon[n] = \sigma \mathbf{w}[n]$. Outside the interval I, the importance of the forecasting error ϵ is also affected by the loss of information resulting from the linearization of the dynamical model we consider in (4). To evaluate the actual behavior of the forward forecasting error $\epsilon[n]$ when $n \geq N$, we determine its first two moments.

1) The mean, or estimation bias, is such that:

$$\mu[n] \stackrel{\Delta}{=} \mathbb{E}\{\epsilon[n]\} = \mathbb{E}\{\tilde{\mathbf{x}}[n]\} - \mathbf{z}[n] . \qquad (16)$$

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

$$\mu[n] = \mathbb{E}\{\boldsymbol{\alpha}^{(n-N+1)}\}\mathbf{z}_K + \sigma \mathbb{E}\{\boldsymbol{\alpha}^{(n-N+1)}\mathbf{w}_K\} - \mathbf{z}[n] \qquad (17)$$

when $n \geq N$.

2) The covariance is given by:

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

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

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

$$\gamma[n, n'] = \sigma \mathbb{E}\{\mathbf{w}[n']\boldsymbol{\alpha}^{(\ell)}\}\mathbf{z}_K + \sigma^2 \mathbb{E}\{\mathbf{w}[n']\boldsymbol{\alpha}^{(\ell)}\mathbf{w}_K\}. \tag{18}$$

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

$$\gamma[n, n'] = \mathbf{z}_K^T \mathbb{E}\left\{\boldsymbol{\alpha}^{(\ell)T}\boldsymbol{\alpha}^{(\lambda)}\right\}\mathbf{z}_K + \sigma \mathbb{E}\{\boldsymbol{\alpha}^{(\ell)}\mathbf{w}_K\boldsymbol{\alpha}^{(\lambda)}\}\mathbf{z}_K$$
$$+ \sigma \mathbb{E}\{\boldsymbol{\alpha}^{(\lambda)}\mathbf{w}_K\boldsymbol{\alpha}^{(\ell)}\}\mathbf{z}_K + \sigma^2 \mathbb{E}\{\boldsymbol{\alpha}^{(\ell)}\mathbf{w}_K\boldsymbol{\alpha}^{(\lambda)}\mathbf{w}_K\}$$
$$- \mathbf{z}[n]\mathbf{z}[n'] - \mathbf{z}[n]\boldsymbol{\mu}[n'] - \mathbf{z}[n]\boldsymbol{\mu}[n'] - \boldsymbol{\mu}[n]\boldsymbol{\mu}[n']. \tag{19}$$

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

In Theorem 1, we specify the asymptotic behavior of the forecasting bias and covariance when the dataset size K is great.

Theorem 1. *Let $\mathbf{x} \in \mathbb{R}^N$ be a discrete-time random signal following model (14). Let $\tilde{\mathbf{x}}$ denotes its forecasting, obtained using the extension Algorithm 2. Let $n \geq N$ be a sample index. Then, the first-order moment of the forecasting error $\epsilon[n]$ in (15) defined in (16) satisfies*

$$|\boldsymbol{\mu}[n]| \leq a_0^{(n)}\sigma^2 + \frac{1}{K}\left(\frac{a_1^{(n)}}{\sigma^2} + a_2^{(n)}\right) + o\left(\frac{1}{K}\right) \tag{20}$$

as $K \to \infty$, where $a_0^{(n)}$, $a_1^{(n)}$, and $a_2^{(n)}$ are positive quantities, independent of K and σ. Its second-order moment $\gamma[n, n']$ satisfies the following approximation equations:

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

$$\left|\gamma[n, n']\right| \leq b_0^{(n,n')}\sigma^2 + \frac{1}{K}\left(b_1^{(n,n')} + b_2^{(n,n')}\sigma^2\right) + o\left(\frac{1}{K}\right), \tag{21}$$

as $K \to \infty$, where $b_0^{(n,n')}$, $b_1^{(n,n')}$, and $b_2^{(n,n')}$ are positive quantities, independent of K and σ;

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

$$\left|\gamma[n, n']\right| \leq c_0^{(n,n')}\sigma^2$$
$$+ \frac{1}{K}\left(\frac{c_1^{(n,n')}}{\sigma^2} + c_2^{(n,n')} + c_3^{(n,n')}\sigma^2\right) + o\left(\frac{1}{K}\right) \tag{22}$$

as $K \to \infty$, where $c_0^{(n,n')}$, $c_1^{(n,n')}$, $c_2^{(n,n')}$ and $c_3^{(n,n')}$ are positive quantities, independent of K and σ.

Proof. See the Supplementary Material. The proof is mainly based on Taylor expansions of the forecasting error. Nonoptimal values of $a_0^{(n)}, a_1^{(n)}, \ldots, b_0^{(n,n')}, \ldots, c_3^{(n,n')}$ are also provided in the proof. ■

The foracasting bias and covariance asymptotically depend linearly on the ratio $\frac{1}{K}$. This shows the need to use a sufficiently large dataset to obtain an accurate forecast. Ideally when $K \to \infty$, the forecasting error would behave like the measurement noise $\sigma\mathbf{w}$, i.e. a zero-mean white noise whose variance is of the order of σ^2. However, Theorem 1 shows that the estimator may remain asymptotically biased (the first term in the bound of equation (20) being independent of K). Nonetheless,

we will see in the numerical discussion, in section IV-C1, that the bias is negligible when $K \to \infty$. Concerning the covariance of the forecasting error, the expected asymptotic behavior is verified. Indeed, when $K \to \infty$, the variance of the forecasting estimator increases at most linearly with the noise variance σ^2, since is bounded by $b_0^{(n,n')}\sigma^2$.

Assume now that K is sufficiently great and fixed. The noise influences the quality of the forecasting estimator in two ways. On one hand, when the noise variance σ^2 is great, the bias and the variance of the estimator potentially increase linearly with σ^2. This reflects the classical behavior of an estimator of a signal degraded by additive noise. On the other hand, due to terms in $1/\sigma^2$, when the noise variance σ^2 is small, the bias and the variance of the forecasting estimator are no longer controlled via equations (20), and (22). This illustrates the necessary presence of noise to obtain a accurate prediction. Indeed, if σ is too small, the matrix $\mathbf{XX}^T$ is ill-conditioned and its inversion in (9) is sensitive. The forecasting is then of poor quality. Noise therefore appears as a regularization term to ensure the invertibility of $\mathbf{XX}^T$. This seemingly counterintuitive fact indicates the challenge we encounter to do forecasting with the dynamic model (3), the degeneracy. Note that from the lag map perspective, $\mathbf{X}$ and $\mathbf{Y}$ consist of points located on a low dimensional manifold. Thus, locally the rank of $\mathbf{X}$ and $\mathbf{Y}$ are low, which leads to the degeneracy. However, when noise exists, it naturally fills in deficient rank, and stabilizes the algorithm.

The dependencies of the forecasting quality on the subsignals lengths M and the forecasting index $\ell = n - N + 1$ are hidden in the expression of the parameters $a_0^{(n)}$, $a_1^{(n)}, \ldots, b_0^{(n,n')}, \ldots, c_3^{(n,n')}$. The numerical results, discussed in section IV-C1, allow us to better evaluate these dependencies.

C. 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*. It is an *AM-FM model* that, in its is continuous-time version, takes the following form

$$z(t) = \sum_{j=1}^{J} a_j(t)\cos(2\pi\phi_j(t)), \tag{23}$$

where $a_j(t)$ and $\phi'_j(t)$ describe how large and fast the signal oscillate at time t. Clearly, (12) is a special case satisfying the AM-FM model when the a_j and ϕ'_j are both constants. In many practical signals, the amplitude and frequency do not change fast. It is thus reasonable to further restrict the regularity and variations of the instantaneous amplitudes and frequencies of these AM-FM functions. When the speeds of variation of the instantaneous amplitudes a_j and frequencies ϕ'_j are small, the signal can be "locally" well approximated by a harmonic

function in (12); that is, locally at time t_0, $z(t)$ can be well approximated by $z_0(t) := \sum_{j=1}^{J} a_j(t_0)\cos(2\pi(\phi_j(t_0) - t_0\phi_j'(t_0) + \phi_j'(t_0)t))$ by the Taylor expansion. An AM-FM function satisfying the slow variation of the instantaneous amplitudes a_j and frequencies ϕ_j' is referred to the adaptive harmonic model (AHM) (see [31], [32] for mathematical details). It is thus clear that the forecasting error caused by the SigExt algorithm additionally depends on the speed of variation of the instantaneous amplitudes a_j and frequencies ϕ_j'. For the forecasting purpose, it is thus clear that when K is not too large, the signal can be well approximated by (12). Hence, Theorem 1 can still be applied to approximate the error. However, how to determine the optimal K based on the signal depends on the application and is out of the scope of this paper. It will be explored in future work for specific application problems.

D. Performance of the Boundary Effects Reduction

While we do not provide a generic proof of the boundary effect reduction on *any* TF analysis tools, we discuss a particular case of SST. Since SST is designed to analyze signals satisfying the AHM, as is discussed in Remark III-C, Theorem 1 ensures that the forecasting error ϵ defined in (15) is controlled and bounded in terms of mean and covariance. Recall Theorem 3 in [31], which states that when the additive noise is stationary and maybe modulated by a smooth function with a small fluctuation, the SST of the observed signal remains close to the SST of the clean signal, throughout the TF plane. We refer the reader to [31] for a precise quantification of the error made in the TF plane, which depends notably on the covariance of the additive noise and on the speed of variation of the amplitudes and instantaneous frequencies composing the signal. In our case, while the noise of the historical data is stationary, the forecasting error depends on the historical noise, and hence the overall "noise" is non-stationary. However, the dependence only appears in the extended part of the signal. We claim that the proof of Theorem 3 in [31] can be generalized to explain the robustness of SST to the noise, and a systematic proof will be reported in our future theoretical work. We verify experimentally that Algorithm 1 is efficient for a large number of representations in the following section IV. Therefore, in our case, this means that the boundary effect is strongly reduced since the impact of the forecasting error does SST is limited. An immediate application is that the instantaneous frequencies can now be well estimated continuously up to the edges of the TF plane, and a real-time acquisition of the instantaneous frequency and amplitude information is possible.

IV. NUMERICAL RESULTS

For the reproducibility purpose, the MATLAB code and datasets used to produce the numerical results in this section are available online at https://github.com/AdMeynard/BoundaryEffectsReduction.

A. Extension Methods for Comparison

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 Algorithm 1 with a simple extension obtained by pointwise symmetrization [15]. We also evaluate the performance of reference forecasting algorithms that could be used for extending such signals. These methods are:

- The EDMD has been developed by Williams *et al.* [18]. The proposed algorithm is a way to obtain an approximation of the so-called Koopman operator of the observed system, which theoretically allows catching dynamic of nonlinear systems [33].
- The GPR [19] is a method relying on a probabilistic dynamical model. That one is based on the Gaussian process structure, and therefore offers more flexibility in the type of dynamic that could be modeled than the linear model (4).
- The TBATS method [21] is based on a classical decomposition of times series into a trend, seasonal and ARMA components, with a specific dynamic for the seasonal component. This model demands the estimation of numerous parameters and, by implication, may be slow.

B. Evaluation Metrics

The first evaluation metric is quantifying the forecasting quality (i.e. not depending on ℓ) in the time domain. We evaluate the Experimental Mean Square Error $\mathrm{MSE}_{\mathrm{xp}}(\tilde{\mathbf{x}})$ of the forward forecast extended signals, namely,

$$\mathrm{MSE}_{\mathrm{xp}}(\tilde{\mathbf{x}}) = \frac{1}{L}\|\tilde{\mathbf{x}} - \mathbf{x}^{\mathrm{ext}}\|^2 \tag{24}$$
$$= \frac{1}{L}\sum_{\ell=1}^{L}\boldsymbol{\mu}_{\mathrm{xp}}[N-1+\ell]^2 + \gamma_{\mathrm{xp}}[N-1+\ell, N-1+\ell].$$

where $\mathbf{x}^{\mathrm{ext}}$ is the *ground-truth extended* signal; that is, $\mathbf{x}^{\mathrm{ext}} := (\mathbf{x}[-L] \quad \cdots \quad \mathbf{x}[N-1+L])$. Then, as long as the bias $\boldsymbol{\mu}[N-1+\ell]$ and the variance $\gamma[N-1+\ell, N-1+\ell]$ of the forecasting estimator remain small for all ℓ, the MSE takes small values either.

The second evaluation metric is evaluating the quality of the boundary effect reduction directly on the TF representation. We compare the obtained TF representation to the optimal TF representation $\mathscr{F}_N^{\mathrm{opt}}(\mathbf{x})$, defined as the restriction of the representation of the ground-truth extended signal $\mathbf{x}^{\mathrm{ext}}$; that is, we have

$$\mathscr{F}^{\mathrm{opt}}(\mathbf{x}_N) = \mathcal{R}\left(\mathscr{F}(\mathbf{x}^{\mathrm{ext}})\right), \tag{25}$$

where $\mathcal{R}$ is defined in (11). To compare different techniques, we use a criterion proposed in [32] that quantifies the optimal transport (OT) distance between a

given TF representation and the optimal one. In general, the OT distance is a quantity that measures how different two probability density functions are. Denote a TF representation as $\mathcal{Q}$. For t fixed, we consider the probability density function defined by $p_{\mathcal{Q}}^t(\xi) = |\mathcal{Q}(\xi, t)|^2 / \int_{\mathbb{R}} |\mathcal{Q}(\nu, t)|^2 d\nu$; that is, normalizing the spectral information at time t. At each time t, the OT distance d_t between two TF representations $\mathcal{Q}$ and $\mathscr{F}^{\mathrm{opt}}$ at time t is defined by

$$d_t(\mathcal{Q}, \mathscr{F}^{\mathrm{opt}}) = \int_{\mathbb{R}} \left| \tilde{P}_{\mathcal{Q}}^t(\xi) - P_{\mathscr{F}^{\mathrm{opt}}}^t(\xi) \right| d\xi \ ,$$

where $P_{\mathcal{Q}}^t(\xi) = \int_{-\infty}^{\xi} p_{\mathcal{Q}}^t(\nu) d\nu$ and $\tilde{P}_{\mathscr{F}^{\mathrm{opt}}}^t(\xi) = \int_{-\infty}^{\xi} \tilde{p}_{\mathscr{F}^{\mathrm{opt}}}^t(\nu) d\nu$. In other words, the OT distance between $\mathcal{Q}$ and $\mathscr{F}^{\mathrm{opt}}$ at time t is given by the L^1 norm of the associated cumulative density function. Note that $d_t(\mathcal{Q}, \mathscr{F}^{\mathrm{opt}})$ quantifies the proximity between the estimated and actual TF representations at time t. With the OT distance, we define the *performance index* $D(\mathcal{Q}, \mathscr{F})$ for the reduction of boundary effects of a given TF representation $\mathcal{Q}$ over another TF representation $\mathscr{F}$ by

$$D(\mathcal{Q}, \mathscr{F}) = \frac{\int_I d_t\left(\mathcal{Q}, \mathscr{F}^{\mathrm{opt}}\right) dt}{\int_I d_t\left(\mathscr{F}, \mathscr{F}^{\mathrm{opt}}\right) dt} \ . \tag{26}$$

In other words, $D(\mathcal{Q}, \mathscr{F})$ is the ratio between its averaged OTD to the optimal TF representation $\mathscr{F}^{\mathrm{opt}}$ (see (25)) and the averaged OTD of the original TF representation $\mathscr{F}$ to $\mathscr{F}^{\mathrm{opt}}$. Thus, $D(\mathcal{Q}) < 1$ means a reduction of the boundary effects. Let us evaluate the quality of the boundary effects reduction on biomedical signals.

C. Simulated Signals

We first evaluate the quality of the forecasting step and compare it to the theoretical results provided by Theorem 1. The level of the forecasting error depends on at least two parameters:

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

In subsections IV-C1 and IV-C2, 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 (12). In order to validate this theoretical result, we apply the forecasting Algorithm 2 to a large number of realizations of the random vector $\mathbf{x}$ of size $N = 10^4$, following model (14), and such that the deterministic component $\mathbf{z}$ takes the form:

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

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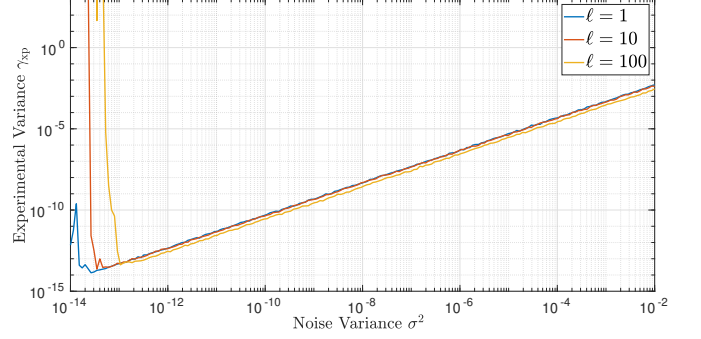

Fig. 2. Evolution of the experimental forecasting variance in function of the noise variance for three different values the forecasting sample index ℓ.

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 500 realizations on the discrete signal $\mathbf{x}$ for 200 different values of σ, logarithmically equispaced from 10^{-7} to 10^{-1}. For each of these values, we determine the experimental bias, denoted as $\mu_{\mathrm{xp}}[N - 1 + \ell]$, and experimental variance, denoted as $\gamma_{\mathrm{xp}}[N - 1 + \ell, N - 1 + \ell]$. Fig. 2 shows the the experimental forecasting variance in function of the noise variance for three different values the forecasting sample index: $\ell = 1$, $\ell = 10$, and $\ell = 100$.

First, when the noise is weak, the variance of the forecasting estimator is dramatically high. For instance, in this specific example, the estimator $\tilde{\mathbf{x}}[N - 1 + \ell]$ for $\ell = 10$ is inaccurate as soon as the noise variance σ^2 is lower than 4×10^{-14} (see the red line in Fig. 2). This validates the result of Theorem 1, highlighting the necessary presence of noise to ensure the numerical stability of the estimator. Furthermore, as soon as σ is higher than this critical threshold, the estimator variance increases linearly with σ^2, as predicted by the bound (22) provided in Theorem 1. Note that the parameter ℓ has little influence on the quality of the estimator. This is due to the *stationarity* of the studied signal, allowing long-range forecasts without deterioration of the estimator.

b) Influence of the Training Dataset Size K: Here, the noise variance σ is set to $\sigma = 10^{-2}$. Then, SigExt is run on 500 realizations of the discrete signal $\mathbf{x}$ for 200 different values of training dataset size K, logarithmically equispaced from 450 to 2000. For each of these values, we determine the experimental bias $\mu_{\mathrm{xp}}[N - 1 + \ell]$ and variance $\gamma_{\mathrm{xp}}[N - 1 + \ell, N - 1 + \ell]$. Fig. 3 shows the the experimental forecasting variance in function of the training dataset size for three different values the forecasting sample index: $\ell = 1$, $\ell = 10$, and $\ell = 100$.

As anticipated by Theorem 1, the variance of the estimator decreases linearly with $\frac{1}{K}$ as long as the dataset size is large enough. Indeed, in the log-log plot displayed in Fig. 3, as soon as $K > 700$, each of the three curves representing the forecasting variance tends towards an asymptote of slope -1.
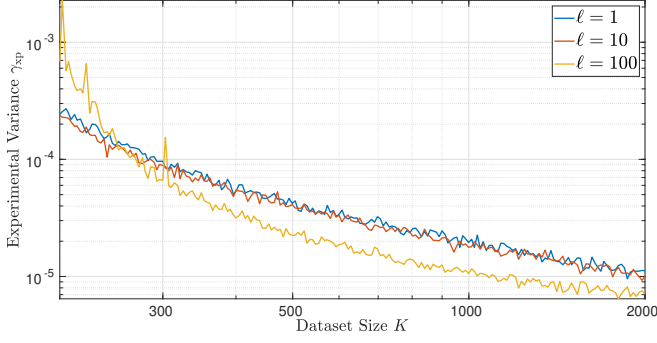
This study overlooks the analysis of the influence of

Fig. 3. Evolution of the experimental forecasting variance in function of the dataset size for three different values the forecasting sample index ℓ.

the parameter M, whose influence on the value of the experimental variance is numerically not significant as long as $M \ll 2K$ and $\ell < M$. The choice of this parameter is especially crucial when the deterministic component of the signal is no longer stationary. The AHM, discussed below, is an example.

2) Adaptive Harmonic Model: Consider a signal satisfying the AHM so that the 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 (14), 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:

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

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

The forecasting methods are applied to 1000 realizations of the above-described signal. The mean and the standard deviation (SD) of the resulting MSEs are given in Table I. They show that the naive extension we propose gives satisfying results, in particular in comparison with the point-symmetric extension. Besides, even though the other more sophisticated methods, like GPR, give MSE values that have a slightly smaller standard deviation, these methods are substantially limited by the computing time they require, which prevents them from being used to exploit real-time data. Thus, SigExt is the extension method that optimizes the trade-off between forecasting quality and computing time.

D. Real Physiological Signals

1) Respiratory Signal: We first consider a thoracic movement signal recorded by the piezo-electrical sensor

TABLE I
AHM Signal: Performance of the Extension Methods

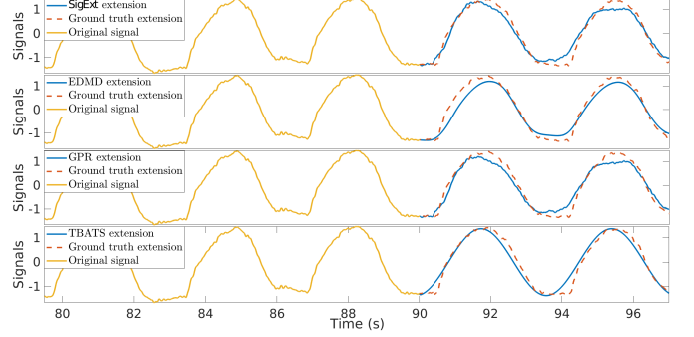| Extension method	MSE (mean $\pm$ SD)	Averaged computing time (seconds)
SigExt	1.028 ± 0.404	1.069
Symmetric	4.368 ± 0.001	0.001
EDMD	1.156 ± 0.003	11.10
GPR	1.025 ± 0.403	244.7
TBATS	1.501 ± 1.354	633.6

Fig. 4. Extended respiratory signal (blue) obtained by the SigExt forecasting (first panel), the EDMD forecasting (second panel), the GPR forecasting (third panel), and the TBATS forecasting (fourth panel), superimposed with the ground truth signal (red-dashed line).

of length 6 hours and 20 minutes. The signal is sampled at $f_{\mathsf{s}} = 100$ Hz. A small portion of this signal is displayed in Fig. 4.

From that long signal, we build a dataset of 379 nonoverlapping signals of 60 seconds, i.e. $N = 6000$. For each segment, we apply the forecasting methods introduced in section IV-C2, including the SigExt method detailed in Algorithm 2. We forecast 7-second-long extensions on each segment of the signal, corresponding to $L = 700$. Thus, in order to catch the dynamical behavior, the size of the training signal M is chosen so that $M = \lfloor 1.5L \rfloor$. As a result of section IV-C1, we take: $K = \lfloor 2.5M \rfloor$. The extensions obtained on one of these subsignals are shown in Fig. 4. The resulting MSEs of different methods are given in Table II. Note that the results are given in "mean $\pm$ standard deviation" format. The TBATS extension method is not implemented, as its excessive computing time makes it impossible to implement it on 6000 segments within a reasonable period of time. Note that the MSE of SigExt is in average higher than the MSE of EDMD or the symmetric extension, with a huge standard deviation. This variability is caused by the presence of few segments contaminated by artifacts so that it is unpredictable via a too simple dynamical model like (4). The left of Fig. 5 illustrates one of these outliers, where SigExt fails to catch the fast varying dynamic of the instantaneous amplitude to satisfactorily forecast the signal. The EDMD and symmetric extensions are more robust to those situations, as shown in this example, and in Table II. Nevertheless, SigExt provides a sufficiently relevant extension to give TF representations sparingly affected by boundary effects. On the right of Fig. 5, we

TABLE II
RESPIRATORY SIGNAL: PERFORMANCE OF THE EXTENSION METHODS
AND THE ASSOCIATED BOUNDARY-FREE TF REPRESENTATIONS

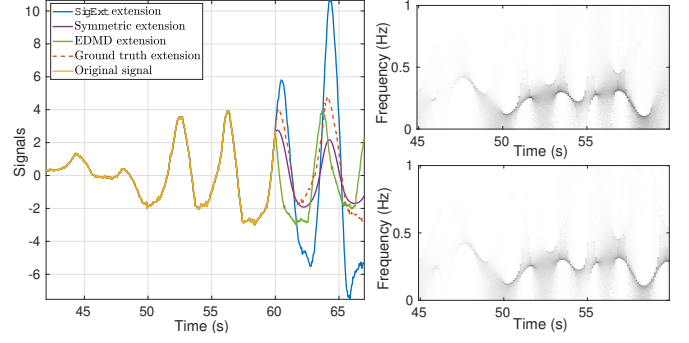| Extension method	MSE	Performance index D (mean $\pm$ SD)			
		STFT	SST	RS	ConceFT
SigExt	0.292 $\pm$4.438	0.370 $\pm$0.623	0.408 $\pm$0.436	0.866 $\pm$0.879	0.423 $\pm$0.344
Symmetric	0.044 $\pm$0.111	1.162 $\pm$0.893	1.173 $\pm$0.886	1.022 $\pm$0.281	1.144 $\pm$0.579
EDMD	0.026 $\pm$0.112	0.359 $\pm$0.266	0.422 $\pm$0.282	0.828 $\pm$0.248	0.449 $\pm$0.296
GPR	0.331 $\pm$4.858	0.391 $\pm$0.853	0.411 $\pm$0.406	0.897 $\pm$1.140	0.430 $\pm$0.364

Fig. 5. Extensions of a segment of the respiratory signal (left) where SigExt is outperformed by the EDMD and Symmetric extensions. Corresponding SST (top-right) and boundary-free SST obtained with SigExt (bottom-right).
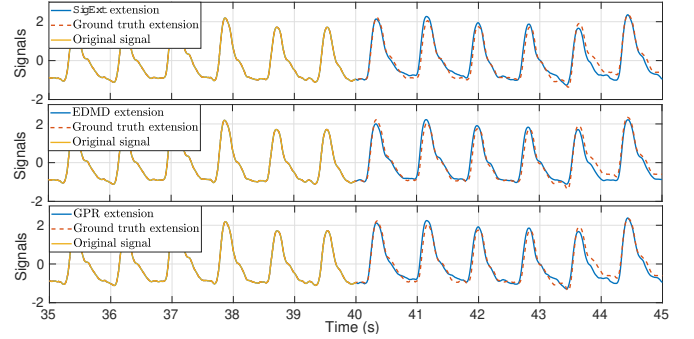

Fig. 6. Extended PPG signal (blue) obtained by the SigExt forecasting (top), the EDMD forecasting (middle), and the GPR forecasting (bottom), superimposed with the ground truth signal (red dash).

display a comparison between the right boundary of SST of the same segment of signal (top-right), and its boundary-free SST obtained after the SigExt forecasting (bottom-right). The extension of the instantaneous frequency visible on the right side of the image illustrates the reduction of boundary effects produced despite an inaccurate signal forecasting.

We then apply BoundEffRed for diverse TF representations: STFT, SST, RS, as well as *Concentration of Frequency and Time* (ConceFT), a generalized multitaper SST-based representation introduced in [32]. We mention that unlike other representations, the standard RS is not causal and rigorously requires knowing the STFT on the whole time-frequency plane before making any reassignment. To enable real-time implementation of RS, we use here a real-time version of RS proposed in [34].

We set the extension length L accordingly to the window length used by the TF analysis tool. For instance, the window length used to evaluate the STFT is of 1400 samples. To prevent the STFT from being sensitive to the boundaries, we set $L = 700$. In this way, the evaluation of the spectral content of the signal near its boundary is not limited by a lack of information all along the window support. From now on, all results are given for L equal to the half of the width of the window used in the TF transform. In Table II, we give the averaged performance index (26) by evaluating the whole TF representations (including boundaries). Even though SigExt performs moderately well in the sense of MSE, the boundary effects are dramatically reduced on the TF representations, and the averaged performance is in the same order of that given by EDMD or GPR. A t-test is performed to compare the performance index of SigExt with those of other methods. Under the null hypothesis that the means are equal, with a 5% significance level, the t-tests show no statistical significant difference between SigExt and EDMD or GPR, regardless of the representation considered.

2) Photoplethysmogram: We perform a study similar to the previous one on a 640-second-long photoplethysmogram (PPG) signal extracted from the Physionet dataset [35], [36], sampled at $f_s = 125$ Hz. A portion of this signal is displayed in Fig. 6. The estimated 5 seconds extensions of this segment obtained by SigExt, EDMD, and GPR forecastings are superimposed to the ground-

truth extension in Fig. 6.

We divide the signal into 32-second-long segments and apply BoundEffRed to each of them. Table III shows the performance index D of the boundary-free TF representations, averaged over the signals. For all the TF representations considered, the results show that SigExt reduces boundary effects about as efficiently as the extensions given by EDMD or GPR. As for the respiratory signal, t-tests show no statistical significant difference between SigExt and EDMD or GPR, regardless of the representation considered. For a visual inspection, the TF representation of SST resulting from the BoundEffRed strategy is shown in the bottom-right panel of Fig. 1, where SST is applied to the portion of PPG displayed in Fig. 6. It produces a significant improvement in the quality of the SST near boundaries. Indeed, the blurring visible when zooming on the right boundary of the SST has almost vanished. Real-time tracking of the instantaneous frequencies contained in the measured signal is therefore greatly facilitated.

To further evaluate the influence of the noise level on the performance of BoundEffRed, we artificially add Gaussian noise to the measured PPG signal. It is thus an additional noise to the measurement noise actually contained in the signal. Fig. 7 shows the averaged performance index of BoundEffRed for different values of the Signal-to-Noise Ratio (SNR). Whatever the represen-

TABLE III
PPG Signal: Performance of the Boundary-Free TF
Representations According to the Extension Method

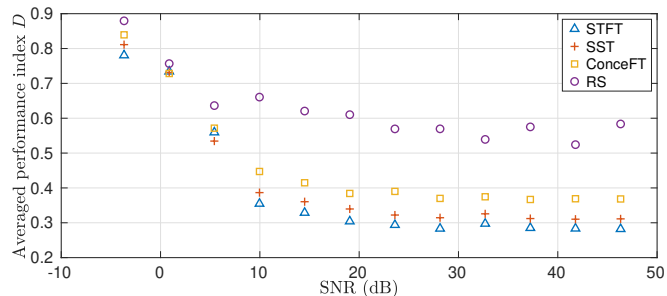| Extension method	Performance index D (mean $\pm$ SD)			
	STFT	SST	ConceFT	RS
SigExt	0.280 $\pm$0.107	0.309 $\pm$0.112	0.367 $\pm$0.183	0.534 $\pm$0.160
Symmetric	1.168 $\pm$0.390	1.209 $\pm$0.340	1.310 $\pm$0.140	0.983 $\pm$0.304
EDMD	0.289 $\pm$0.126	0.319 $\pm$0.134	0.375 $\pm$0.163	0.503 $\pm$0.163
GPR	0.276 $\pm$0.106	0.303 $\pm$0.110	0.361 $\pm$0.165	0.544 $\pm$0.157

Fig. 7. Performance index of BoundEffRed applied on a PPG, in function of the SNR.

tation considered, BoundEffRed is relatively robust to noise, as long as the SNR remains above 10 dB. Below this level, the reduction of boundary effects gradually deteriorates.

E. Real-Time Implementation

We simulate the real-time implementation of the SST from BoundEffRed applied to the PPG described in section IV-D2, subsampled by a factor of 2. The test is performed on a 2-Core Intel Core i5 CPU running at 1.7 GHz and 7.7 GB of RAM.

For this signal, sampled at $f_{\mathsf{s}} = 65.5$ Hz, the suitable window is 8 seconds long. Therefore, the extension must be over 4 seconds, i.e. $L = 250$. The forecasting step in each iteration of BoundEffRed then takes no more than $t_{\text{forecast}} = 46$ ms. Denote by $H \geq 1$ the hop size in samples, that is the number of samples between successive columns of the SST. Besides, since the support of the analysis window covers $2L$ samples, the update of the SST requires the calculation of $\lceil 2L/H \rceil$ new columns of the SST. In this example, the frequency dimension of the SST being 512, the computational time of one column is $t_{\text{SST}} = 2.08$ ms, on average. A general rule to determine the acceptable values of H for real-time implementation is verifying that the computational time to update the boundary-free TF representation is smaller than the lag between H samples; that is,

$$ t_{\text{forecast}} + \left\lceil \frac{L}{H} \right\rceil t_{\text{SST}} < \frac{H}{f_{\mathsf{s}}} . $$

In our example, taking $H \geq 8$ samples is sufficient to ensure the feasibility of real-time implementation. This

value is reasonable because it allows a maximum overlap of 98.4% of the window length. SST variations are therefore perceived gradually. Combined with the low latency of the algorithm, this allows for smooth monitoring of the real-time SST. A video clip illustrating this is available on GitHub, at https://github.com/AdMeynard/BoundaryEffectsReduction/tree/master/Animations.

V. Conclusion

In this paper, we propose an algorithm, named BoundEffRed, for the real-time reduction of boundary effects in TF representations. This method is based on an extension of the signal obtained by a simple-minded and numerically efficient forecasting. We have shown theoretically that the chosen dynamic model is sufficient to extend signals formed by a sum of sine waves. Moreover, the low computational time allows us to switch to a real-time implementation of BoundEffRed, unlike other existing forecasting methods. The numerical results also confirmed the robustness to noise of BoundEffRed, as well as its ability to be applied to many TF representations. An additional application to ECG is included into the supplementary material.

Various improvements can be considered to make the algorithm more robust. In particular, we have noticed (see Fig. 5) that when the regular oscillations of the observed signal break, the forecasting step is no longer relevant, and only slows down the calculation of the TF representation. A preliminary step should then be added to the algorithm to detect signal activity, and disable the forecasting step when possible. A more fundamental improvement would be to perform the forecasting step directly in the TF plane, to specifically predict the evolution of each instantaneous frequency. This leads to seeing a TF representation as a multivariate time series, with each frequency band containing a univariate time series. These avenues will be explored in our future work.

References

[1] D. Stowell, *Computational Analysis of Sound Scenes and Events*. Springer, 2018, ch. Computational Bioacoustic Scene Analysis, pp. 303–333.

[2] M. Müller, D. P. W. Ellis, A. Klapuri, and G. Richard, "Signal processing for music analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1088–1110, 2011.

[3] Z. Peng, F. Chu, and Y. He, "Vibration signal analysis and feature extraction based on reassigned wavelet scalogram," *Journal of Sound and Vibration*, vol. 253, no. 5, pp. 1087 – 1100, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022460X01940854

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

[5] P. Stoica and R. Moses, *Spectral analysis of signals*. Upper Saddle River, N.J.: Pearson/Prentice Hall, 2005.

[6] G. Matz, F. Hlawatsch, and W. Kozek, "Generalized evolutionary spectral analysis and the Weyl spectrum of nonstationary random processes," *IEEE Transactions on Signal Processing*, vol. 45, no. 6, pp. 1520–1534, Jun. 1997.

[7] K. Gröchenig, *Foundations of time-frequency analysis*, ser. Applied and Numerical Harmonic Analysis. Boston, MA: Birkhäuser Inc., 2001.

[8] P. Flandrin, *Time-frequency/time-scale analysis*, ser. Wavelet Analysis and its Applications. San Diego: Academic Press Inc., 1999, vol. 10.

[9] I. Daubechies, *Ten lectures on wavelets*, ser. CBMS-NSF Regional Conference Series in Applied Mathematics. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 1992, vol. 61.

[10] I. Daubechies, J. Lu, and H.-T. Wu, "Synchrosqueezed wavelet transforms: An empirical mode decomposition-like tool," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 243 – 261, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520310001016

[11] 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.

[12] N. Delprat, B. Escudié, P. Guillemain, R. Kronland-Martinet, P. Tchamitchian, and B. Torrésani, "Asymptotic wavelet and Gabor analysis: extraction of instantaneous frequencies," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 644–664, Mar. 1992.

[13] C. K. Chui and E. Quak, *Numerical Methods in Approximation Theory*, ser. ISNM 105: International Series of Numerical Mathematics. Basel: Birkäuser, 1992, vol. 9, ch. Wavelets on a Bounded Interval, pp. 53–75.

[14] U. Depczynski, K. Jetter, K. Molt, and A. Niemöller, "The fast wavelet transform on compact intervals as a tool in chemometrics: II. Boundary effects, denoising and compression," *Chemometrics and Intelligent Laboratory Systems*, vol. 49, no. 2, pp. 151 – 161, 1999. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169743999000374

[15] 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.

[16] L. Chen, T. Q. Nguyen, and K.-P. Chan, "Symmetric extension methods for m-channel linear-phase perfect-reconstruction filter banks," *IEEE Transactions on Signal Processing*, vol. 43, no. 11, pp. 2505–2511, 1995.

[17] J. R. Williams and K. Amaratunga, "A discrete wavelet transform without edge effects using wavelet extrapolation," *The Journal of Fourier Analysis and Applications*, vol. 3, pp. 435–449, 1997.

[18] 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.

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

[20] S. Roberts, M. Osborne, M. Ebden, S. Reece, N. Gibson, and S. Aigrain, "Gaussian processes for time-series modelling," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 371, no. 1984, p. 20110550, 2013. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2011.0550

[21] 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

[22] M. West and J. Harrison, *Bayesian forecasting and dynamic models*. Springer Science & Business Media, 2006.

[23] J. Fan and I. Gijbels, *Local polynomial modelling and its applications: monographs on statistics and applied probability 66*. CRC Press, 1996, vol. 66.

[24] P. Hall and J. D. Opsomer, "Theory for penalised spline regression," *Biometrika*, vol. 92, no. 1, pp. 105–118, 2005.

[25] Y.-W. Chang, C.-J. Hsieh, K.-W. Chang, M. Ringgaard, and C.-J. Lin, "Training and testing low-degree polynomial data mappings via linear svm." *Journal of Machine Learning Research*, vol. 11, no. 4, 2010.

[26] J. S. Marron, S. Adak, I. Johnstone, M. Neumann, and P. Patil, "Exact risk analysis of wavelet regression," *Journal of Computational and Graphical Statistics*, vol. 7, no. 3, pp. 278–309, 1998.

[27] P. R. Vlachas, W. Byeon, Z. Y. Wan, T. P. Sapsis, and P. Koumoutsakos, "Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 474, no. 2213, p. 20170844, 2018.

[28] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. OTexts, 2018.

[29] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, vol. 656, pp. 5–28, 2010.

[30] F. Takens, "Detecting strange attractors in turbulence," in *Dynamical Systems and Turbulence*, ser. Lecture Notes in Mathematics, D. Rand and L.-S. Young, Eds. Springer Berlin Heidelberg, 1981, vol. 898, pp. 366–381.

[31] Y.-C. Chen, M.-Y. Cheng, and H.-T. Wu, "Non-parametric and adaptive modelling of dynamic periodicity and trend with heteroscedastic and dependent errors," *Journal of the Royal Statistical Society Series B*, vol. 76, no. 3, pp. 651–682, June 2014. [Online]. Available: https://ideas.repec.org/a/bla/jorssb/v76y2014i3p651-682.html

[32] 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

[33] 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

[34] Y.-T. Lin and H.-T. Wu, "ConceFT for Time-Varying Heart Rate Variability Analysis as a Measure of Noxious Stimulation During General Anesthesia," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 1, pp. 145–154, 2017.

[35] M. A. F. Pimentel, A. E. W. Johnson, P. H. Charlton, D. Birrenkott, P. J. Watkinson, L. Tarassenko, and D. A. Clifton, "Toward a robust estimation of respiratory rate from pulse oximeters," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 8, pp. 1914–1923, 2017.

[36] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, M. R. G., M. J. E., M. G. B., P. C. K., and S. H. E., "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. circulation [online]. 101 (23),," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.