

# Study of boundary conditions in the Iterative Filtering method for the decomposition of nonstationary signals

Antonio Cicone\*      Pietro Dell'Acqua†

February 14, 2019

## Abstract

Non-stationary and non-linear signals are ubiquitous in real life. Their decomposition and analysis is an important research topic of signal processing. Recently a new technique, called Iterative Filtering, has been developed with the goal of decomposing such signals into simple oscillatory components. Several papers have been devoted to the investigation of this technique from a mathematical point of view. All these works start with the assumption that each compactly supported signal is extended periodically outside the boundaries. In this work, we tackle the problem of studying the influence of different boundary conditions on the decompositions produced by the Iterative Filtering method. In particular, the choice of boundary conditions gives rise to different types of structured matrices. Thus, we describe their spectral properties and then convergence properties of Iterative Filtering algorithm (in which such matrices are involved). Numerical results provide an interesting overview on important aspects (such as accuracy and error propagation) of the techniques proposed and show the way of further promising developments.

## 1 Introduction

Given a real life non-stationary signal  $\mathbf{s}(x)$ ,  $x \in \mathbb{R}$ , we may be interested in decomposing it into simple components in order to identify features and quasi periodicities hidden in it. We can think, for instance, to a economic index, like the GDP of a nation, or a geophysical signal, like the sea level during a tsunami, as well as to an engineering measure, like the vibrations of a structure or machinery. Standard techniques, like Fourier or wavelet transform, cannot help, in general, to decompose meaningfully non-stationary signals. Whereas, following the idea proposed by Huang et al. in [8], we can iteratively decompose such signals into a finite sequence of simple components, defined Intrinsic Mode Functions (IMFs), which fulfill two properties: *i*) the number of extrema and the number of zero crossings must either equal or differ at most by one; *ii*) considering upper and lower envelopes connecting respectively all the local maxima and minima of the function, their mean has to be zero at any point.

The method originally proposed by Huang et al. in [8], called Empirical Mode Decomposition (EMD), proved to be unstable to small perturbations. For this reason an alternative algorithm, called Ensemble Empirical Mode Decomposition (EEMD), was proposed in [14] which is based on the idea of applying EMD to an ensemble of signals produced perturbing hundreds of times the given one with noise. The decomposition is then derived as the average decomposition of the ensemble. In [10] the authors proposed an alternative technique to the EMD, called Iterative Filtering (IF), which has the very same structure of EMD, but it is stable and convergent both in the continuous setting [4, 9] and in the discrete one [6, 7]. We point out that IF has been also generalized producing the so called Adaptive Local Iterative Filtering (ALIF) algorithm whose convergence and stability are under investigation [4, 3]. For further details on why standard techniques may

---

\*Istituto Nazionale di Alta Matematica, Città Universitaria, P.le Aldo Moro 5, 00185, Roma, Italy, and DISIM, Università degli Studi dell'Aquila, via Vetoio 1, 67100, L'Aquila, Italy, and Gran Sasso Science Institute, viale Francesco Crispi 7, 67100, L'Aquila, Italy ([antonio.cicone@univaq.it](mailto:antonio.cicone@univaq.it))

†Libera Università di Bolzano, Piazza Domenicani 3, 39100 Bolzano, Italy ([pietro.dellacqua@gmail.com](mailto:pietro.dellacqua@gmail.com))

fail in decomposing a non-stationary signal and the advantages of using EEMD or IF we refer the interested reader to [2].

In this work we consider the case of compactly supported and discrete signals. For simplicity we assume that each signal  $\mathbf{s} = [\mathbf{s}(x_j)]_{j=0}^{n-1}$  (with  $n \in \mathbb{N}$ ) is supported on  $[0, 1]$  and it is sampled at  $n$  points  $x_j = \frac{j}{n-1}$ , with  $j = 0, \dots, n-1$ . Without losing generality we can further assume that  $\|\mathbf{s}\|_2 = 1$ .

Any signal decomposition method which deals with a compactly supported signal requires assumptions on how the signal extends outside the boundaries, the so called Boundary Conditions (BCs). This aspect has a key role in many applications, for instance in image restoration [12], and usually the goal is to employ BCs able to guarantee good accuracy of the approximated solution computed by some numerical algorithm [11]. Regarding IF, in [6] the authors addressed its convergence when it is assumed that the signals extend periodically outside the boundaries. The questions which are still open are: does IF converge also for other kinds of BCs? Given a signal extended artificially outside the boundaries in a certain way, how do the errors introduced outside the boundaries effect the decomposition in the iterations? Given a compactly supported signal what is the best choice in terms of BCs?

The paper is organized as follows. In Section 2 we review the IF algorithm applied to discrete and compactly supported signals. Section 3 is devoted to a summary of BCs and their properties. In Section 4 we study the IF convergence when different BCs are chosen for the signal. In Section 5 we present an extended version of IF method, which is useful for addressing the question of how the errors propagate from outside the boundaries to the inside. This work ends with some numerical examples showing the impact of BCs on decomposition quality and error propagation in Section 6, and concluding remarks in Section 7.

## 2 Discrete Iterative Filtering

In this section we review the IF method focusing on the case of discrete and compactly supported signals  $\mathbf{s}$ . We start from the definition of filter

**Definition 1.** A non-negative vector  $\mathbf{w} = (0, \dots, 0, w_{-l}, \dots, w_{-1}, w_0, w_1, \dots, w_l, 0, \dots, 0) \in \mathbb{R}^n$  such that  $w_j > 0$  for  $j = -l, \dots, l$  and  $\sum_{j=-l}^l w_j = 1$  is called a **filter** of length  $l$ , with  $0 < l \leq \lfloor \frac{n-1}{2} \rfloor$ . If a filter is such that  $w_{-j} = w_j$  for  $j = 1, \dots, l$ , then  $\mathbf{w}$  is called **symmetric**. If a filter is such that for  $0 \leq i < j$ ,  $w_i \geq w_j$  and for  $j < i \leq 0$ ,  $w_i \geq w_j$ , then  $\mathbf{w}$  is called **decreasing**.

In this work we consider only symmetric filters. When we have a symmetric filter associated with some step  $m$ , we employ the following notation

$$\mathbf{w}_m = (0, \dots, 0, w_{l_m}^m, \dots, w_1^m, w_0^m, w_1^m, \dots, w_{l_m}^m, 0, \dots, 0), \quad (1)$$

with  $w_0^m + 2 \sum_{j=1}^{l_m} w_j^m = 1$ .

If we assume that some filter shape  $h : [-1, 1] \rightarrow \mathbb{R}$  (symmetric with respect to  $y$ -axis) has been selected a priori, like one of the Fokker-Planck filters described in [4], then the elements  $w_j^m$  can be computed, for  $j = 0, 1, \dots, l_m$ , by the linear scaling formula

$$w_j^m = h\left(\frac{j}{l_m}\right) \frac{1}{l_m}, \quad (2)$$

where  $l_m$  is the length that characterizes the filter. Assuming  $\mathbf{s}_1^m = \mathbf{s}$ , where the two indices will become

clear in the next paragraph, the main step of the IF method, for  $i = 0, \dots, n - 1$ , is

$$\begin{aligned}
\mathbf{s}_{k+1}^m(x_i) &= \mathbf{s}_k^m(x_i) - \int_{x_i - \frac{l_m}{n-1}}^{x_i + \frac{l_m}{n-1}} \mathbf{s}_k^m(y) h \left( \frac{(x_i - y)(n-1)}{l_m} \right) \frac{n-1}{l_m} dy \\
&\approx \mathbf{s}_k^m(x_i) - \sum_{x_j = x_i - \frac{l_m}{n-1}}^{x_i + \frac{l_m}{n-1}} \mathbf{s}_k^m(x_j) h \left( \frac{(x_i - x_j)(n-1)}{l_m} \right) \frac{1}{l_m} \\
&= \mathbf{s}_k^m(x_i) - \sum_{j=i-l_m}^{i+l_m} \mathbf{s}_k^m(x_j) h \left( \frac{i-j}{l_m} \right) \frac{1}{l_m} \\
&= \mathbf{s}_k^m(x_i) - \sum_{j=i-l_m}^{i+l_m} \mathbf{s}_k^m(x_j) w_{|i-j|}^m
\end{aligned}$$

Algorithm 1 provides the pseudocode of the Discrete Iterative Filtering (DIF) Algorithm. We observe that the first while loop is called Outer Loop, whereas the second one Inner Loop. In the notation  $\mathbf{s}_k^m$ ,  $m$  denotes the step relative to the Outer Loop, while  $k$  denotes the step relative to the Inner Loop. The idea

---

**Algorithm 1 Discrete Iterative Filtering** IMFs = DIF( $s, h$ )

---

```

 $m = 1$ 
 $\mathbf{s}_1^m = \mathbf{s}$ 
while the number of extrema of  $\mathbf{s}_1^m \geq 2$  do
   $k = 1$ 
  compute the filter length  $l_m$  for the signal  $\mathbf{s}_k^m$ 
  compute  $\mathbf{w}_m$  (having  $h$  and  $l_m$ )
  while the stopping criterion is not satisfied do
     $(\mathbf{s}_k^m)^{\text{BC}}(x_i) = \mathbf{s}_k^m(x_i)$ ,  $i = 0, \dots, n - 1$ 
    apply BCs for computing  $(\mathbf{s}_k^m)^{\text{BC}}(x_i)$ ,  $i = -l_m, \dots, -1$  and  $i = n, \dots, n - 1 + l_m$ 
     $\mathbf{s}_{k+1}^m(x_i) = \mathbf{s}_k^m(x_i) - \sum_{j=i-l_m}^{i+l_m} (\mathbf{s}_k^m)^{\text{BC}}(x_j) w_{|i-j|}^m$ ,  $i = 0, \dots, n - 1$ 
     $k = k + 1$ 
  end while
   $\mathbf{f}_m = \mathbf{s}_k^m$ 
   $\mathbf{s}_1^{m+1} = \mathbf{s}_1^m - \mathbf{f}_m$ 
   $m = m + 1$ 
end while
 $\mathbf{f}_m = \mathbf{s}_1^m$ 
IMFs =  $\{\mathbf{f}_1, \dots, \mathbf{f}_m\}$ 

```

---

is to suitably choose the filter length  $l_m$  in order to capture the desired frequencies in the IMF  $\mathbf{f}_m$ , and this is done by subtracting iteratively from the signal its moving average computed as convolution of the signal itself with the selected filter. In matrix form we have

$$\mathbf{s}_{k+1}^m = (I - W_m) \mathbf{s}_k^m, \quad (3)$$

where  $W_m$  is a structured matrix constructed from the filter  $\mathbf{w}_m$  and the BCs imposed. In particular  $W_m$  can be written as the sum of two matrices

$$W_m^{\text{BC}} = T_m + K_m^{\text{BC}}, \quad (4)$$

where the first one is a Toeplitz matrix, while the second one is a correction matrix which depends on BCs (for details, see Section 3).

From (3), it follows immediately that

$$\mathbf{s}_{k+1}^m = (I - W_m)^k \mathbf{s}_1^m, \quad (5)$$

so ideally the first IMF is given by

$$\mathbf{f}_1 = \lim_{k \rightarrow \infty} (I - W_1)^k \mathbf{s}. \quad (6)$$

However, in the implemented algorithm we do not let  $k$  to go to infinity, instead we use a stopping criterion. We can define, for instance, the following quantity

$$\Delta_k^m := \frac{\|\mathbf{s}_{k+1}^m - \mathbf{s}_k^m\|_2}{\|\mathbf{s}_k^m\|_2}, \quad (7)$$

so we can either stop the process when the value  $\Delta_k^m$  reaches a certain threshold or we can introduce a limit on the maximal number of iterations for all the Inner Loops. It is also possible to adopt different stopping criteria for different Inner Loops.

### 3 Boundary conditions and structured matrices

Boundary conditions deal with the problem of extending the signal outside the field of view in which the detection is made. So let  $\mathbf{s}$  be the signal inside the boundaries and let  $p$  be the parameter relative to the space outside the boundaries, we have that, for  $j = 1, \dots, p$ , Zero BCs are defined as

$$\mathbf{s}(x_{-j}) = 0, \quad \mathbf{s}(x_{n-1+j}) = 0, \quad (8)$$

Periodic BCs are defined as

$$\mathbf{s}(x_{-j}) = \mathbf{s}(x_{n-j}), \quad \mathbf{s}(x_{n-1+j}) = \mathbf{s}(x_{j-1}), \quad (9)$$

Reflective BCs are defined as

$$\mathbf{s}(x_{-j}) = \mathbf{s}(x_{j-1}), \quad \mathbf{s}(x_{n-1+j}) = \mathbf{s}_{n-j}, \quad (10)$$

Anti-Reflective BCs are defined as

$$\mathbf{s}(x_{-j}) = 2\mathbf{s}(x_0) - \mathbf{s}(x_j), \quad \mathbf{s}(x_{n-1+j}) = 2\mathbf{s}(x_{n-1}) - \mathbf{s}(x_{n-1-j}). \quad (11)$$

According to the BCs imposed, we have a different kind of structured matrix as  $W^{BC}$ , whose elements are defined from values of the filter  $\mathbf{w}$  of length  $l$ . As said, here we take into account symmetric filters, since this choice allows to have useful theoretical properties. In particular, symmetry is not necessary to get the algebra of Circulant matrices, associated with Periodic BCs, while it is necessary to get the algebra of Reflective matrices and the algebra of Anti-Reflective matrices. The symmetry property also allows in all these three cases to have a fast transform that can be employed for computing matrix-vectors products in an efficient way. In particular, we have Discrete Fourier Transform (DFT) for Circulant matrices, Discrete Cosine Transform of type III (DCT-III) for Reflective matrices and Anti-Reflective Transform (ART) – strongly linked to Discrete Sine Transform of type I (DST-I) – for Anti-Reflective matrices.





In the Anti-Reflective case, the structure of the matrix is more involved, namely

$$W^{\mathcal{AR}} = \begin{pmatrix} z_1 + w_0 & 0 & \dots & \dots & \dots & 0 & 0 \\ z_2 + w_1 & & & & & & \vdots \\ \vdots & & & & & & \vdots \\ z_l + w_{l-1} & & & & & & 0 \\ w_l & & \hat{W}^{\mathcal{AR}} & & & & w_l \\ 0 & & & & & & z_l + w_{l-1} \\ \vdots & & & & & & \vdots \\ \vdots & & & & & & z_2 + w_1 \\ 0 & 0 & \dots & \dots & \dots & 0 & z_1 + w_0 \end{pmatrix}_{n \times n}, \quad (20)$$

where  $z_j = 2 \sum_{k=j}^l w_k$  and  $\hat{W}^{\mathcal{AR}} = PW^{\mathcal{AR}}P^T$ , with

$$P = \begin{pmatrix} 0 & 1 & & & 0 \\ 0 & & 1 & & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & & & & 1 & 0 \\ 0 & & & & & 1 & 0 \end{pmatrix}_{(n-2) \times n}. \quad (21)$$

Moreover,  $\hat{W}^{\mathcal{AR}} = \hat{T} - \hat{H}^{\mathcal{AR}}$ , where  $\hat{T}$  is a Toeplitz matrix

$$\hat{T} = \begin{pmatrix} w_0 & w_1 & \dots & w_l \\ w_1 & w_0 & w_1 & \ddots & w_l \\ \vdots & w_1 & w_0 & w_1 & \ddots & \ddots \\ w_l & \ddots & w_1 & w_0 & \ddots & \ddots & w_l \\ & w_l & \ddots & \ddots & \ddots & w_1 & \ddots & w_l \\ & & \ddots & \ddots & w_1 & w_0 & w_1 & \vdots \\ & & & w_l & \ddots & w_1 & w_0 & w_1 \\ & & & & w_l & \dots & w_1 & w_0 \end{pmatrix}_{(n-2) \times (n-2)}, \quad (22)$$

while  $\hat{H}^{\mathcal{AR}}$  is a Hankel matrix

$$\hat{H}^{\mathcal{AR}} = \begin{pmatrix} w_2 & w_3 & \dots & w_l \\ w_3 & \ddots & \ddots & \\ \vdots & \ddots & & \\ w_l & & & \\ & & & w_l \\ & & & \vdots \\ & & & \ddots \\ & & \ddots & \ddots & w_3 \\ w_l & \dots & w_3 & w_2 \end{pmatrix}_{(n-2) \times (n-2)}. \quad (23)$$

$\hat{W}^{\mathcal{AR}}$  can be diagonalized by  $\hat{Q}^{\mathcal{AR}}$ , that is the  $(n-2)$ -dimensional DST-I, having entries

$$[\hat{Q}^{\mathcal{AR}}]_{i,j} = \sqrt{\frac{2}{n-1}} \sin\left(\frac{ij\pi}{n-1}\right), \quad i, j = 1, \dots, n-2. \quad (24)$$

Eigenvalues of  $\hat{W}^{\mathcal{AR}}$  can be obtained by the following formula

$$\hat{\lambda}_i^{\mathcal{AR}} = \frac{[\hat{Q}^{\mathcal{AR}}(\hat{W}^{\mathcal{AR}}\hat{\mathbf{e}}_1)]_i}{[\hat{Q}^{\mathcal{AR}}\hat{\mathbf{e}}_1]_i} \quad (25)$$

where  $\hat{\mathbf{e}}_1 = (1, 0, \dots, 0)^T \in \mathbb{R}^{n-2}$ . Therefore, using the variable  $t = \frac{i\pi}{n-1}$ , and exploiting

$$\sin(2t) = 2 \sin(t) \cos(t)$$

and

$$\cos(2t) = 1 - 2 \sin^2(t)$$

and

$$\begin{aligned} \sin((j+2)t) - \sin(jt) &= \sin(jt) \cos(2t) + \sin(2t) \cos(jt) - \sin(jt) \\ &= \sin(jt)(1 - 2 \sin^2(t)) + 2 \sin(t) \cos(t) \cos(jt) - \sin(jt) \\ &= -2 \sin(jt) \sin^2(t) + 2 \sin(t) \cos(t) \cos(jt) \end{aligned}$$

and

$$2 \sin(jt) \sin(t) = \cos(jt - t) - \cos(jt + t)$$

and

$$2 \cos(t) \cos(jt) = \cos(jt - t) - \cos(jt + t)$$

we get that eigenvalues of  $\hat{W}^{\mathcal{AR}}$  can be computed by this formula, for  $i = 1, \dots, n-2$ ,

$$\begin{aligned} \hat{\lambda}_i^{\mathcal{AR}} &= \sum_{j=1}^{n-2} [\hat{W}^{\mathcal{AR}}]_{1,j} \frac{\sin\left(\frac{ji\pi}{n-1}\right)}{\sin\left(\frac{i\pi}{n-1}\right)} \\ &= w_0 + w_1 \frac{\sin(2t)}{\sin(t)} + \sum_{j=1}^{l-1} w_{j+1} \frac{\sin((j+2)t)}{\sin(t)} - w_{j+1} \frac{\sin(jt)}{\sin(t)} \\ &= w_0 + 2w_1 \cos(t) + \sum_{j=1}^{l-1} w_{j+1} (2 \cos(t) \cos(jt) - 2 \sin(jt) \sin(t)) \\ &= w_0 + 2w_1 \cos(t) + 2 \sum_{j=1}^{l-1} w_{j+1} \cos((j+1)t) \\ &= w_0 + 2w_1 \cos(t) + 2 \sum_{j=2}^l w_j \cos(jt) \\ &= w_0 + 2 \sum_{j=1}^l w_j \cos\left(\frac{ji\pi}{n-1}\right) \end{aligned} \quad (26)$$

By Lemma 3.1 in [13], we know that the eigenvalues of  $W^{\mathcal{AR}}$  are given by 1 with multiplicity two and by the eigenvalues  $\{\hat{\lambda}_i^{\mathcal{AR}}\}_{i=1, \dots, n-2}$  of  $\hat{W}^{\mathcal{AR}}$ .

Looking at the structure of the matrices presented, it can be easily verified that the eigenvector  $u_1^{\mathcal{P}}$  associated with  $\lambda_1^{\mathcal{P}} = 1$  is  $(1, \dots, 1)^T$ , the eigenvector  $u_1^{\mathcal{R}}$  associated with  $\lambda_1^{\mathcal{R}} = 1$  is  $(1, \dots, 1)^T$ , and the eigenvectors associated with  $\lambda_1^{\mathcal{AR}} = \lambda_2^{\mathcal{AR}} = 1$  are  $u_1^{\mathcal{AR}} = (0, 1, 2, \dots, n-2, n-1)^T$  and  $u_2^{\mathcal{AR}} = (n-1, n-2, \dots, 2, 1, 0)^T$ .

We notice that  $Q^{\mathcal{P}}$ ,  $Q^{\mathcal{R}}$  and  $\hat{Q}^{\mathcal{AR}}$  are all unitary matrices. However, the last one is linked to  $\hat{W}^{\mathcal{AR}}$ , so we also need to know how to diagonalize the original matrix  $W^{\mathcal{AR}}$ . In other words, basing on DST-I, we

need to introduce ART. Unfortunately this transform is not unitary, since the matrices in Anti-Reflective algebra are in general not normal. Recalling theoretical results reported in [1, 12], we have that  $W^{AR}$  can be diagonalized by  $Q^{AR}$ , that is the  $n$ -dimensional ART, defined as follows

$$Q^{AR} = \begin{pmatrix} (n-1)\eta^{-1} & 0 & \dots & \dots & \dots & 0 & 0 \\ (n-2)\eta^{-1} & & & & & & \eta^{-1} \\ (n-3)\eta^{-1} & & & & & & 2\eta^{-1} \\ \vdots & & & & & & \vdots \\ \vdots & & & \hat{Q}^{AR} & & & \vdots \\ \vdots & & & & & & \vdots \\ 2\eta^{-1} & & & & & & (n-3)\eta^{-1} \\ \eta^{-1} & & & & & & (n-2)\eta^{-1} \\ 0 & 0 & \dots & \dots & \dots & 0 & (n-1)\eta^{-1} \end{pmatrix}_{n \times n}, \quad (27)$$

where  $\eta = \sqrt{\sum_{j=0}^{n-1} j^2}$  is introduced just to normalize the first and the last column, which come from eigenvectors  $u_1^{AR}$  and  $u_2^{AR}$ .

Summarizing, in this Section we have introduced BCs, analyzing the properties of matrices associated with their choice, in particular providing descriptions of matrix structures, formulas for computation of eigenvalues and diagonalization results (associated with discrete transforms). All this will be very useful in Section 4, in which convergence properties of DIF will be investigated.

## 4 Spectral properties and convergence results

At this stage, we consider some BCs (Periodic or Reflective or Anti-Reflective). By exploiting results presented in Section 3, we are able to prove spectral properties of  $W^{BC}$  and convergence results for DIF algorithm.

**Lemma 1.** *Let  $W^{BC} \in \mathbb{R}^{n \times n}$  be the structured matrix constructed from a symmetric decreasing filter  $\mathbf{w}$  of length  $0 < l \leq \lfloor \frac{n-1}{2} \rfloor$ , imposing some BCs (Periodic or Reflective or Anti-Reflective). Then  $\sigma(W^{BC}) \subseteq [-1, 1]$ .*

*Proof.* By definition  $W^{BC}$  is a symmetric matrix, so it has a real spectrum. By considering the following estimate relative to the spectral radius

$$\rho(W^{BC}) \leq \|W^{BC}\|_{\infty} = \max_i \sum_{j=1}^n |[W^{BC}]_{i,j}| = w_0 + 2 \sum_{k=1}^l w_k = 1,$$

we can conclude that all eigenvalues lie in the interval  $[-1, 1]$ .  $\square$

**Theorem 1.** *Let  $\mathbf{s} \in \mathbb{R}^n$  the signal that has to be decomposed. Let  $\mathbf{v}$  be a symmetric decreasing filter of length  $0 < l' \leq \lfloor \frac{n-1}{4} \rfloor$  and  $\mathbf{w} = \mathbf{v} * \mathbf{v}$  be another symmetric decreasing filter of length  $0 < l \leq \lfloor \frac{n-1}{2} \rfloor$  defined by making the convolution of  $\mathbf{v}$  with itself. Then for the matrix  $W^{BC} \in \mathbb{R}^{n \times n}$  constructed from  $\mathbf{w}$  and some BCs (Periodic or Reflective or Anti-Reflective), it holds that  $\sigma(W^{BC}) \subseteq [0, 1]$ . Moreover,  $\lambda_1^P = 1$  and  $\{\lambda_i^P\}_{i=2, \dots, n} \subseteq [0, 1)$ ,  $\lambda_1^R = 1$  and  $\{\lambda_i^R\}_{i=2, \dots, n} \subseteq [0, 1)$ ,  $\lambda_1^{AR} = \lambda_2^{AR} = 1$  and  $\{\lambda_i^{AR}\}_{i=3, \dots, n} \subseteq [0, 1)$ .*

*Proof.* As already said, in case of Periodic or Reflective or Anti-Reflective BCs we are in a matrix algebra; this means that the matrix product gives rise to a matrix which can still be interpreted as a matrix constructed from a filter and the same BCs. In particular, thanks to the fact that  $l' \leq \lfloor \frac{n-1}{4} \rfloor$ , such filter can be computed by means of convolution. Thus, if we denote by  $V^{BC}$  the structured matrix associated with filter  $\mathbf{v}$ , we have that  $W^{BC} = (V^{BC})^2$ . By Lemma 1,  $\sigma(V^{BC}) \subseteq [-1, 1]$ , therefore  $\sigma(W^{BC}) \subseteq [0, 1]$ .





## 5 Extended Iterative Filtering

Now, once defined the matrix

$$R = ( O_{n \times p} \ I_{n \times n} \ O_{n \times p} )_{n \times (n+2p)}, \quad (29)$$

where  $O$  is matrix of all zeros and  $I$  is the identity matrix, in Algorithm 1 we present the pseudocode of Extended Iterative Filtering (EIF) algorithm. Differently from the DIF method, it is not based (as DIF algorithm) on a structured matrix  $W_m^{BC} \in \mathbb{R}^{n \times n}$  (whose structure depends on BCs), associated with the original signal  $\mathbf{s} \in \mathbb{R}$ , but on a Circulant matrix  $\check{W}_m^P \in \mathbb{R}^{(n+2p) \times (n+2p)}$  (whose structure does not depend on BCs), associated with a signal  $\mathbf{s}^{BC} \in \mathbb{R}^{(n+2p)}$  extended by means of any BCs. Hence, since in this framework the BCs do not affect the matrix structure, we have more freedom in their choice. Any method that extends the original signal in a suitable way outside the boundaries can be employed now. Moreover, we notice that in Algorithm 2 there is only one step involving the use of BCs, while in Algorithm 1 the BCs are employed at every step of the Inner Loop. Considering that every time we make use of BCs we are adding errors to the decomposition method, we think that the proposed approach may improve the standard one. Furthermore, we remark that employing Periodic BCs allows to reduce the computational burden of IF. In fact the algorithm can be rewritten using Fast Fourier Transform (FFT) to become what is known as Fast Iterative Filtering (FIF) [6].

---

### Algorithm 2 Extended Iterative Filtering IMF<sub>s</sub> = EIF( $s, h$ )

---

```

 $\mathbf{s}^{BC}(x_i) = \mathbf{s}(x_i), i = 0, \dots, n - 1$ 
apply BCs to compute  $\mathbf{s}^{BC}(x_i), i = -p, \dots, -1$  and  $i = n, \dots, n - 1 + p$ 
 $m = 1$ 
 $\mathbf{s}_1^m = \mathbf{s}^{BC}$ 
while the number of extrema of  $\mathbf{s}_1^m \geq 2$  do
   $k = 1$ 
  compute the filter length  $l_m$  for the signal  $\mathbf{s}_k^m$ 
  compute  $\mathbf{w}_m$  (having  $h$  and  $l_m$ )
  while the stopping criterion is not satisfied do
     $\mathbf{s}_{k+1}^m = (I - \check{W}_m^P)\mathbf{s}_k^m$ 
     $k = k + 1$ 
  end while
   $\mathbf{f}_m = \mathbf{s}_k^m$ 
   $\mathbf{s}_1^{m+1} = \mathbf{s}_1^m - \mathbf{f}_m$ 
   $m = m + 1$ 
end while
 $\mathbf{f}_m = \mathbf{s}_1^m$ 
IMFs =  $\{R\mathbf{f}_1, \dots, R\mathbf{f}_m\}$ 

```

---

We denote by  $\mathbf{f}_1 \in \mathbb{R}^n$  the first approximated IMF computed by the EIF algorithm.  $\bar{\mathbf{f}}_1 \in \mathbb{R}^n$  represents the first exact IMF, whereas  $\mathbf{s}^{BC} \in \mathbb{R}^{n+2p}$  the signal extended according to some a priori chosen boundary conditions and  $\check{\mathbf{s}} \in \mathbb{R}^{n+2p}$  the unknown real signal extended also outside the boundaries. Then we can split these last two vectors considering the portion of signal inside and outside the field of view identified by the boundaries, and get

$$\mathbf{s}^{BC} = \mathbf{s}_{in}^{BC} + \mathbf{s}_{out}^{BC} \quad (30)$$

and

$$\check{\mathbf{s}} = \check{\mathbf{s}}_{in} + \check{\mathbf{s}}_{out}. \quad (31)$$

Clearly  $\mathbf{s}_{in}^{BC} = \check{\mathbf{s}}_{in}$ .

We have that

$$\begin{aligned}
\mathbf{f}_1 &= R(I - \ddot{W}_1^{\mathcal{P}})^k \mathbf{s}^{\mathcal{BC}} \\
&= R(I - \ddot{W}_1^{\mathcal{P}})^k (\mathbf{s}_{\text{in}}^{\mathcal{BC}} + \mathbf{s}_{\text{out}}^{\mathcal{BC}} + \ddot{\mathbf{s}}_{\text{out}} - \ddot{\mathbf{s}}_{\text{out}}) \\
&= R(I - \ddot{W}_1^{\mathcal{P}})^k \ddot{\mathbf{s}} + R(I - \ddot{W}_1^{\mathcal{P}})^k (\mathbf{s}_{\text{out}}^{\mathcal{BC}} - \ddot{\mathbf{s}}_{\text{out}})
\end{aligned}$$

where the first term tends to  $\bar{\mathbf{f}}_1$ , while the second term it propagates itself from the boundaries inside the field of view as  $k$  grows.

In general it is not possible to estimate the difference  $\mathbf{s}_{\text{out}}^{\mathcal{BC}} - \ddot{\mathbf{s}}_{\text{out}}$ . However it is reasonable to assume that the approximation of the real signal outside the boundaries  $\mathbf{s}_{\text{out}}^{\mathcal{BC}}$  produces an error  $\mathbf{s}_{\text{out}}^{\mathcal{BC}} - \ddot{\mathbf{s}}_{\text{out}}$  which is a constant function. In order to estimate such error function we can assume that its value equals  $\chi = \max(|\mathbf{s}_{\text{in}}^{\mathcal{BC}}|) = \max(|\ddot{\mathbf{s}}_{\text{in}}|)$ . This is clearly in most cases an overestimation.

Now the question is how the error  $R(I - \ddot{W}_1^{\mathcal{P}})^k (\mathbf{s}_{\text{out}}^{\mathcal{BC}} - \ddot{\mathbf{s}}_{\text{out}})$  propagates inside the field of view during the iterations. We can estimate an upper bound for the error inside the field of view at step  $k$  of the iterations using the following formula

$$\text{err}_k = R(I - \ddot{W}_1^{\mathcal{P}})^k \mathbf{u} = R \left( I - \binom{k}{1} \ddot{W}_1^{\mathcal{P}} + \binom{k}{2} \left( \ddot{W}_1^{\mathcal{P}} \right)^2 + \dots + (-1)^k \left( \ddot{W}_1^{\mathcal{P}} \right)^k \right) \mathbf{u}, \quad (32)$$

where  $\mathbf{u} \in \mathbb{R}^{n+2p}$  is a vector equal to  $\chi$  outside the field of view and 0 inside.

A potential upper bound of the error for each point  $x_i$  inside the field of view can be computed as

$$\text{ub}_k(x_i) = \max_{j \leq k} (|\text{err}_j(x_i)|), \quad (33)$$

where  $k$  is the number of iterations. This is just an estimate because in general the actual error introduced by the extension of the signal outside the boundaries is unknown. If we deal with artificial examples, and we consider DIF algorithm employing some BCs, we can compute the actual error using the formula

$$\text{err}_k^{\mathcal{BC}}(x_i) = |\mathbf{f}_1(x_i) - \bar{\mathbf{f}}_1(x_i)|, \quad (34)$$

where  $k$  is the number of iterations required to produce the approximated IMF  $\mathbf{f}_1$ , while  $\bar{\mathbf{f}}_1$  is the first exact IMF.

## 6 Numerical results

In order to show how DIF method works when we employ different BCs, we present four simple examples in which the signal is artificially created through one IMF plus a trend. In particular, we analyze how the errors introduced outside the boundaries propagate inside the field of view, evaluating the performance of formula (32) for the a priori estimation of the error when there is no knowledge on the behavior of the signal outside the boundaries.

### 6.1 Example 1

We consider a first signal, shown on the left of Figure 1, which can safely extended periodically and anti-reflectively outside the boundaries. On the right of Figure 1 we can look at decomposition obtained after 3 steps of DIF method with Reflective BCs. If we apply the DIF algorithm using Periodic, Reflective and Anti-Reflective BCs, we can extract a first IMF and measure the difference, after a fixed number of steps, between the exact IMF and the computed IMF measured in absolute. In Figure 2 we plot the errors introduced by each extension, formula (34), and we compare them with the a priori estimate of the error computed using formula (33). Due to a symmetry in the signal, it is sufficient to plot the errors for the left half of the field of view. As expected, we produce a minimal error in the decomposition if we extended the signal periodically or anti-reflectively. Furthermore, the a priori upper bound estimate of the error proves to be correct.

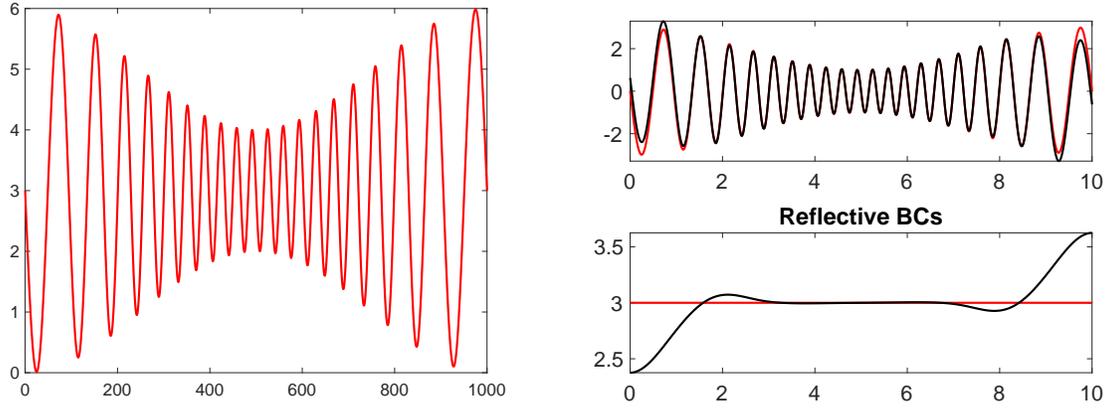


Figure 1: (Example 1) Left panel: signal. Right panel: decomposition computed by DIF method with Reflective BCs (black line) compared with exact one (red line).

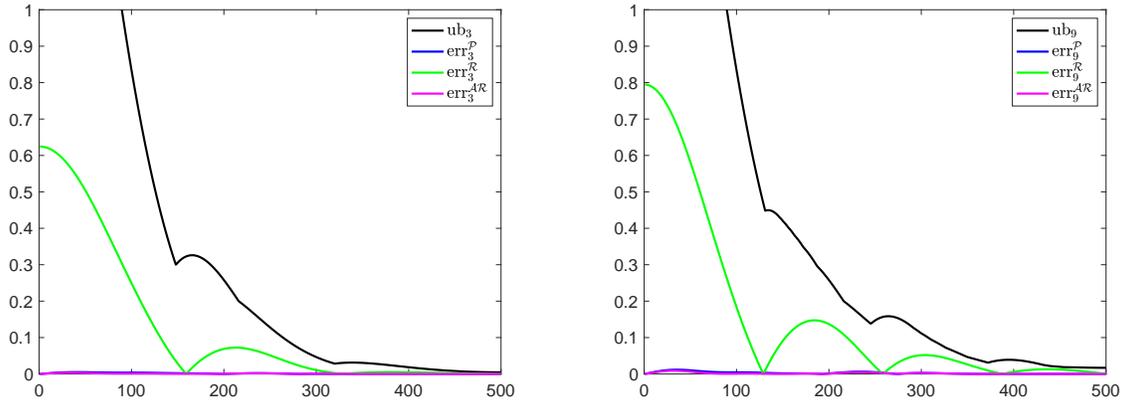


Figure 2: (Example 1) Upper bound and errors measured in absolute value for Periodic, Reflective and Anti-Reflective BCs relative to 3 iterations (on the left) and 9 iterations (on the right) of DIF method.

## 6.2 Example 2

In this second example we consider the signal plotted in the left panel of Figure 3, while in the right panel we can look at decomposition obtained after 3 steps of DIF method with Anti-Reflective BCs. After extending the signal using Periodic, Reflective and Anti-Reflective BCs, we extract a first IMF using the DIF method. In Figure 4 we compare the absolute values of the errors introduced by the different BCs, formula (34), and we compute the upper bound by means of (33). Again, due to a symmetry in the curves, we report only the first half of the error plot. In this case the reflective extension works best in minimizing the error in the decomposition. Furthermore, the upper bound on the error allows to correctly estimate the error everywhere inside the field of view.

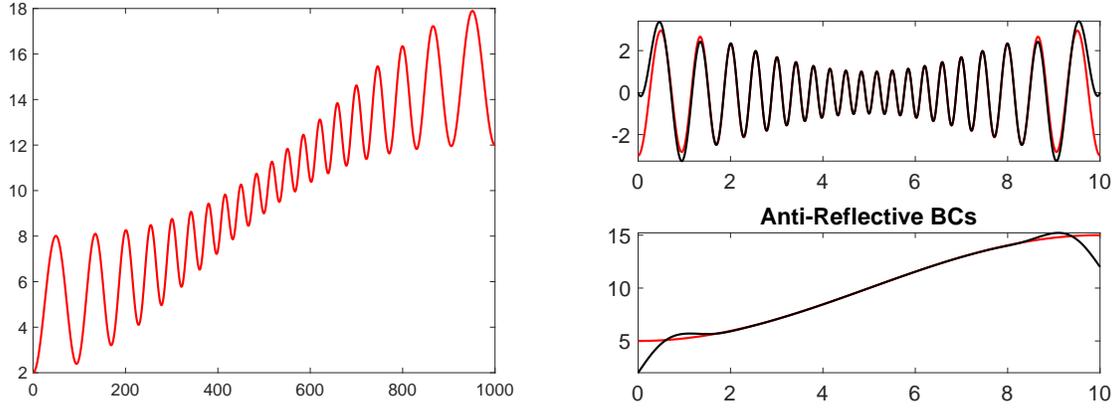


Figure 3: (Example 2) Left panel: signal. Right panel: decomposition computed by DIF method with Anti-Reflective BCs (black line) compared with exact one (red line).

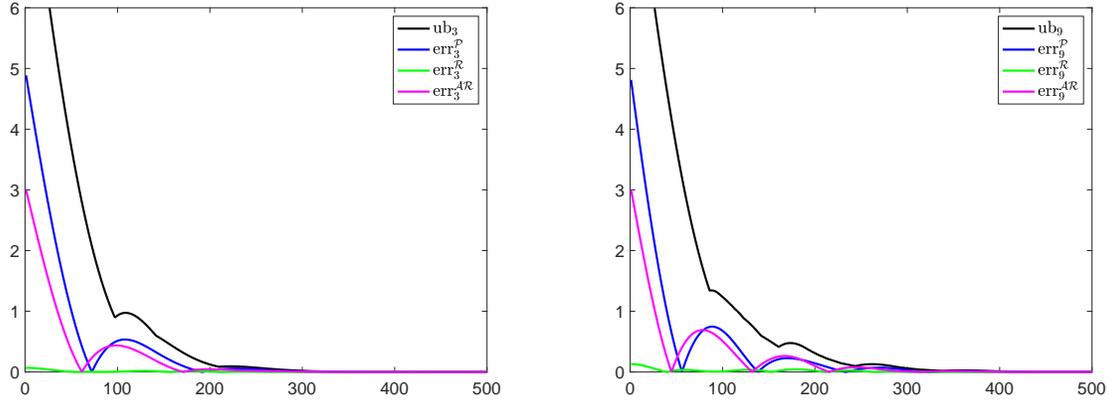


Figure 4: (Example 2) Upper bound and errors measured in absolute value for Periodic, Reflective and Anti-Reflective BCs relative to 3 iterations (on the left) and 9 iterations (on the right) of DIF method.

### 6.3 Example 3

In this case the signal, shown in the left panel of Figure 5, is best extended outside the boundaries using Anti-Reflective BCs. This fact is confirmed by the error curves (34) plotted in Figure 6. Due to symmetry properties of the signal under study, we report only the first half of the error curves. Also in this example the upper bound allows to give a priori estimate of the behavior relative to the error propagation inside the field of view.

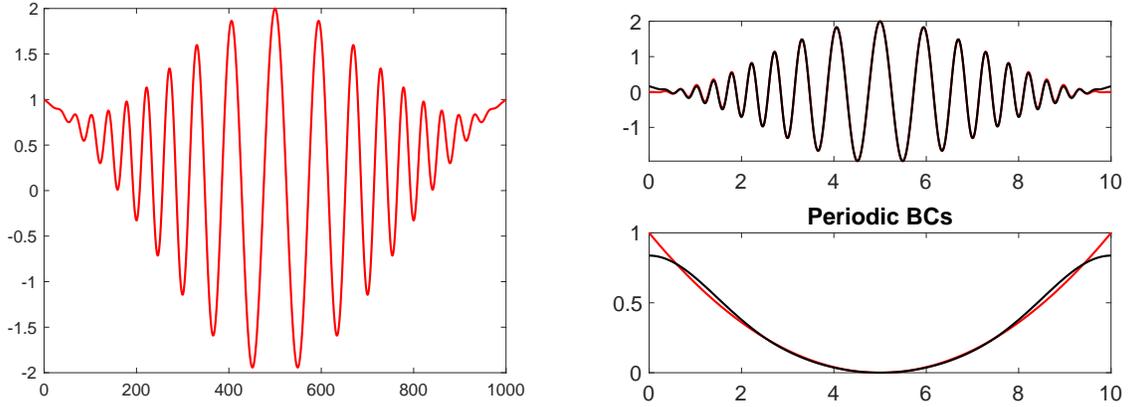


Figure 5: (Example 3)Left panel: signal. Right panel: decomposition computed by DIF method with Periodic BCs (black line) compared with exact one (red line).

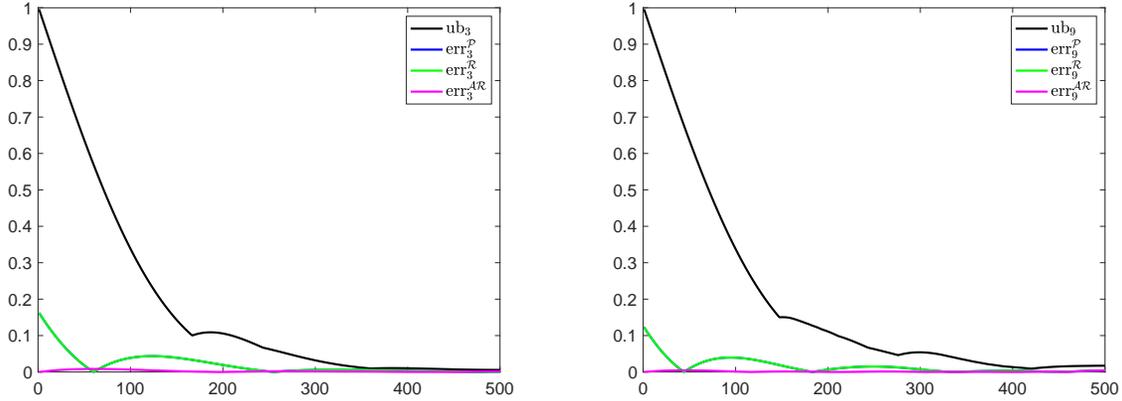


Figure 6: (Example 3)Upper bound and errors measured in absolute value for Periodic, Reflective and Anti-Reflective BCs relative to 3 iterations (on the left) and 9 iterations (on the right) of DIF method.

## 6.4 Example 4

In this example we want to study how the errors induced by the extension outside the boundaries depends on the phase of the signal at the boundary. We consider a simple signal, plotted in the left panel of Figure 7, which is a superposition of a constant trend and a plain sine. The signal support is originally given by the interval  $[-333, 0]$  which is then extended up to 300 step by step of  $\Delta t = 0.01$ . Each time we enlarge the support of a  $\Delta t$  we recompute the newly extended signal using DIF with different kind of BCs. For each BCs we compute the relative error

$$\text{err}_{\text{rel}}^{\text{BC}} = \frac{\|\mathbf{f}_1 - \bar{\mathbf{f}}_1\|_{\infty}}{\|\bar{\mathbf{f}}_1\|_{\infty}} \quad (35)$$

and the relative error upper bound

$$\text{ub}_{\text{rel}} = \frac{\|\text{ub}_k\|_{\infty}}{\|\bar{\mathbf{f}}_1\|_{\infty}} \quad (36)$$

where  $\bar{f}_1$  represents the first exact IMF and  $ub_k$  is the error upper bound computed using (33) for a fixed number of iterations  $k$ . We plot the relative errors in the right panel of Figure 7. As expected such curves

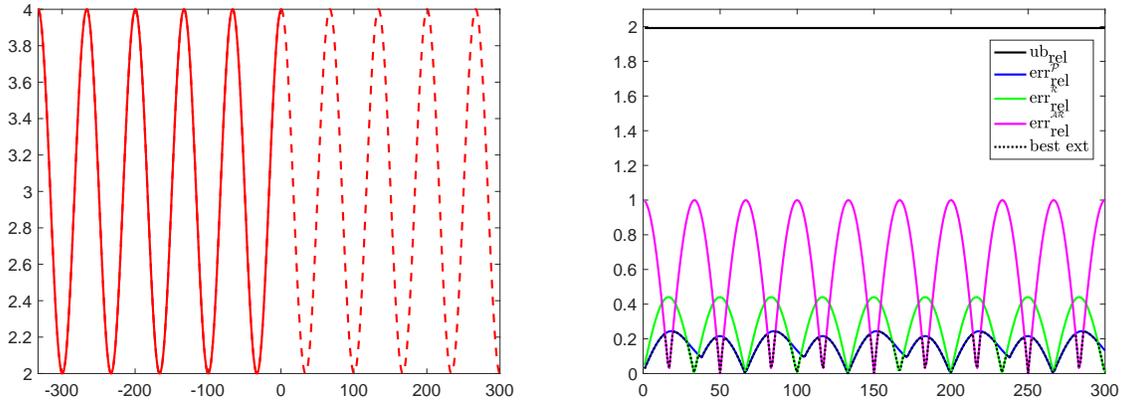


Figure 7: (Example 4) Left panel: signal. Right panel: upper bound, relative errors for different BCs and best extension.

have the same periodicity as the given signal. Furthermore the relative error upper bound a priori estimate proves to be valid also in this example. Finally this study allows to identify which kind of extension is performing best from a relative error point of view for each value of the phase of the signal at the boundary, dashed curve the right panel of Figure 7.

## 7 Conclusions

We considered the decomposition of non-stationary signals, which is a problem of great interest from the theoretical point of view and has important applications in many different fields. For instance, it occurs in the identification of hidden periodicities and trends in time series relative to natural phenomena (like average troposphere temperature) and economic dynamics (like financial indices). Since standard techniques like Fourier or Wavelet Transform are unable to properly capture non-stationary phenomena, in the last years several ad hoc methods have been proposed in the literature. Such techniques provide iterative procedures for decomposing a signal into a finite number of simple components.

In this paper we focused on investigating IF algorithm employing different BCs (Periodic, Reflective and Anti-Reflective BCs), which give rise to different matrix structures. We analyzed spectral properties of these matrices and convergence properties of IF method. We also presented an extended version of IF, in which any BCs can be employed; this allows to estimate the error propagation from the boundary towards the internal part of the signal. Numerical experiments show that a suitable choice of BCs is able to improve in a meaningful way the quality of signal decomposition in IMFs computed by IF method.

We think that this paper can open the way for further interesting developments, for instance the study of adaptive choice of BCs, basing on the signal at hand, and the proposal of new accurate BCs, as done in [11] for image restoration problem.

## Acknowledgments

This work was supported by the Istituto Nazionale di Alta Matematica (INdAM) “INdAM Fellowships in Mathematics and/or Applications cofunded by Marie Curie Actions”, FP7-PEOPLE-2012-COFUND, Grant agreement n. PCOFUND-GA-2012-600198.

## References

- [1] A. ARICÒ, M. DONATELLI, J. NAGY, AND S. SERRA-CAPIZZANO, *The Anti-Reflective Transform and Regularization by Filtering*, Numerical Linear Algebra in Signals, Systems and Control (2011), pp. 1–21.
- [2] A. CICONE, *Nonstationary signal decomposition for dummies*, preprint.
- [3] A. CICONE, C. GARONI, AND S. SERRA-CAPIZZANO, *Spectral and convergence analysis of the Discrete ALIF method*, submitted.
- [4] A. CICONE, J. LIU, AND H. ZHOU, *Adaptive local iterative filtering for signal decomposition and instantaneous frequency analysis*, Appl. Comput. Harmon. Anal., 41 (2016), pp. 384–411.
- [5] A. CICONE, J. LIU, AND H. ZHOU, *Hyperspectral chemical plume detection algorithms based on multi-dimensional iterative filtering decomposition*, Phil. Trans. R. Soc. A: Math. Phys. Eng. Sci., 374 (2016), pp. 20150196.
- [6] A. CICONE AND H. ZHOU, *Iterative Filtering algorithm numerical analysis with new efficient implementations based on FFT*, preprint.
- [7] A. CICONE AND H. ZHOU, *Multidimensional iterative filtering method for the decomposition of high-dimensional non-stationary signals*, Numer. Math. Theory Methods Appl., 10 (2017), pp. 278–298.
- [8] N.E. HUANG, Z. SHEN, S.R. LONG, M.C. WU, H.H. SHIH, Q. ZHENG, N.C. YEN, C.C. TUNG, AND H.H. LIU, *The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis*, Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 454 (1998), pp. 903.
- [9] C. HUANG, L. YANG, AND Y. WANG, *Convergence of a convolution-filtering-based algorithm for empirical mode decomposition*, Advances in Adaptive Data Analysis, 1(2009), pp. 561–571.
- [10] L. LIN, Y. WANG, AND H. ZHOU, *Iterative filtering as an alternative algorithm for empirical mode decomposition*, Advances in Adaptive Data Analysis, 1(2009), pp. 543–560.
- [11] P. DELL’ACQUA, *A note on Taylor boundary conditions for accurate image restoration*, Adv. Comput. Math., 43 (2017), pp. 1283–1304.
- [12] P. DELL’ACQUA, M. DONATELLI, S. SERRA-CAPIZZANO, D. SESANA, C. TABLINO-POSSIO, *Optimal preconditioning for image deblurring with Anti-Reflective boundary conditions*, Linear Algebra and its Applications, 502 (2016), pp. 159–185.
- [13] S. SERRA CAPIZZANO, *A note on antireflective boundary conditions and fast deblurring models*, SIAM J. Sci. Comput., 25 (2003), pp. 1307–1325.
- [14] Z. WU AND N. E. HUANG, *Ensemble empirical mode decomposition: a noise-assisted data analysis method*, Adv. Adapt. Data Anal., 1 (2009), pp. 1–41.