



**HAL**  
open science

# Forward-backward Filtering and Penalized Least-Squares Optimization: A Unified Framework

Arman Kheirati Roonizi, Christian Jutten

► **To cite this version:**

Arman Kheirati Roonizi, Christian Jutten. Forward-backward Filtering and Penalized Least-Squares Optimization: A Unified Framework. *Signal Processing*, 2021, 178, pp.107796. 10.1016/j.sigpro.2020.107796 . hal-03019950

**HAL Id: hal-03019950**

**<https://hal.science/hal-03019950>**

Submitted on 23 Nov 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Forward-backward Filtering and Penalized Least-Squares Optimization: A Unified Framework

Arman Kheirati Roonizi<sup>a,\*</sup>, Christian Jutten<sup>b</sup>

<sup>a</sup>Department of Computer Science, Faculty of Science, Fasa University, Fasa, Iran

<sup>b</sup>Gipsa-Lab, CNRS, Univ. Grenoble Alpes, France

---

## ARTICLE INFO

### Keywords:

Forward-backward filtering  
Penalized least squares optimization  
Zero-phase filtering  
Butterworth  
Chebyshev  
Quadratic variation regularization

## ABSTRACT

The paper proposes a framework for unification of the penalized least-squares optimization (PLSO) and forward-backward filtering scheme. It provides a mathematical proof that forward-backward filtering (zero-phase IIR filters) can be presented as instances of PLSO. On the basis of this result, the paper then represents a unifying approach to the design and implementation of forward-backward filtering and PLSO algorithms in the time and frequency domain. A new block-wise matrix formulation is also presented for implementing the PLSO and forward-backward filtering algorithms. The approach presented in this paper is particularly suited for understanding the task of zero-phase filters in the time domain and analyzing PLSO algorithms in the frequency domain. In this paper, we show that the task of a zero-phase digital Butterworth filter in the time domain is to fit the signal with impulse train and penalties on the derivatives of the fitted model. For a zero-phase digital Chebyshev filter, a linear combination of derivatives of the model is used in the penalty term.

---

## 1. Introduction

Filtering/smoothing is a major branch of signal processing, based on prior assumptions on the signal and noise structure, with the objectives of attenuating the noise and recovering the underlying signals. Therefore, signal filtering and smoothing have been the subject of many research works. In this paper, we present a unifying framework for a category of zero-phase filters and show that they can be presented as instances of penalized least-squares optimization (PLSO).

PLSO offers a methodological approach for solving problems in a wide variety of application domains where the problem is described as the minimization of a cost function, consisting of the sum of a data fit term and penalty terms. It has been widely used in application areas like control [5, 11], economics and finance [10], communications [49], signal and image processing [7, 20, 21, 17], compressed sensing [14], statistics [44], machine learning and computer vision [3, 48, 50], to name just a few. In functional data analysis, PLSO is widely used for data smoothing, where the penalty terms are connected to fitting data by a linear differential equation [32]. Total variation (TV) regularization denoising is an example of the PLSO algorithm where the penalty term is defined by the simplest differential equation.

---

\*Corresponding author

 ebad.kheirati.roonizi@gmail.com ( Arman Kheirati Roonizi)

ORCID(s): 0000-0003-2696-920X ( Arman Kheirati Roonizi)

<sup>1</sup>This is the author's version of the manuscript accepted for publication in Elsevier Signal Processing, August 2020. The paper can be found in its final format at: <https://doi.org/10.1016/j.sigpro.2020.107796>

Introduced first by Rudin, Osher, and Fatemi for imaging problems [33], TV has been widely modified and used in many applications such as clustering and segmentation [7, 20, 21, 9]. An improved TV algorithm denoted as higher degree TV (HDTV) introduced in [20], has been extended in [21] to higher dimensions and to a wider class of penalties based on higher degree differential operators. HDTV has received significant attention by the researchers in the field of signal and image processing in the past two decades [20, 17, 9, 8, 1, 30] due to its properties in recovered signal, especially preserving fast transitions and discontinuities in the signal while preventing oscillations. In [42], Unser et al. proposed algorithms for designing recursive regularized filters with first and second difference operations. In [43], the authors presented a mathematical framework to justify the use of splines in signal processing applications and studied the relation between smoothing splines and Wiener filter. Recently, we have proposed an improved smoothing filter (i.e., smoothness priors or quadratic variation) using bilinear transform [23].

Recursive digital filters are the simplest usual method for signal denoising. They are mainly used to extract a signal within a predetermined frequency band. Infinite impulse response (IIR) digital filters can be designed from the classical analog filters (e.g., Butterworth, Chebyshev, elliptic and etc.) [28, 29, 39, 40]. IIR digital filters, however, have nonlinear phase, which implies time-shifts varying according to the frequency. Hence, at the filter output, the summation of the frequency components with different time-shifts generates signal distortions. This issue is particularly important in biomedical signal processing where the signal patterns are often considered by clinicians and biomedical engineers as indexes of some underlying pathologies. For example, the ST-segment in an electrocardiogram (ECG) signal can be analyzed for detecting ischemia and intrapartum fetal monitoring, and the peak-and-wave pattern on an EEG tracing can be considered as a marker of epilepsy. Therefore, the filtering procedure should not introduce any form of morphological changes that can mimic pathological changes. To avoid this risk, in biomedical applications, the designed filters should have linear phase frequency response at least in the signal frequency range. A common way to bypass problems related to a nonlinear phase response is to filter the noisy data with one causal filter in the forward time direction, then time-reversing the output of that process and filtering it again with the same filter and time reversing it again [38, 16]. The procedure is known as forward-backward filtering. It has applications in zero-phase filtering and the implementation of non-causal filters.

In this paper, we propose a new framework for implementing the zero-phase filters. The contribution of this paper is to present a framework for the unification of forward-backward filtering and PLSO algorithms. We especially prove that, in the time domain, a zero-phase filter can be represented as a least-squares data smoothing with a delta impulse train and a penalty on the derivatives of the fitted model. The proposed framework can find various applications in signal processing. As an example, we use it for appropriate cutoff frequency selection of zero-phase filters and frequency analysis of PLSO algorithms. The paper also shows that zero-phase filters and PLSO are special cases of the optimal Wiener filter: they can be formed as an optimal Wiener filter for a random process obtained from an autoregressive

(AR) model driven by input white (innovation) noise in presence of white observation noise. Zero-phase filters and PLSO algorithms such as smoothness priors and quadratic variation denoising<sup>2</sup> algorithms, Butterworth and Chebyshev are optimal only when the signal is a random process obtained from a specific AR model driven by input white noise in presence of white observation noise. In other cases, when the noises are not white (colored spectral noises), the paper suggests that the regularization factor or cutoff frequency should be a function of the power spectral density (PSD) of noises.

## 2. Background

Let us consider a noisy observation  $y(t)$ , of an original signal  $x(t)$ , modeled as

$$y(t) = x(t) + v(t), \quad (1)$$

where  $v(t)$  is the additive noise which can be either white or colored random process signal. We assume that the noise  $v(t)$  and signal  $x(t)$  are uncorrelated. Further, we consider  $x(t)$  as a signal with a decaying Fourier transform  $X(\omega)$ , i.e.,

$$\int_{-\infty}^{\infty} |X(\omega)|(1 + |\omega|^n)d\omega < \infty$$

which implies that  $x(t)$  is globally  $n$  times continuously differentiable (see [25], Theorem 2.5). In order to digitally process the signal, one needs to consider  $y_k = y(kT_s)$ , the discrete-time samples of  $y(t)$ ,  $x_k$  the sampled desired signal and  $v_k$  the sampled observation noise:

$$y_k = x_k + v_k, \quad k = 1, \dots, L. \quad (2)$$

We particularly consider the reconstruction procedures that recover the signal by one of the following approaches: 1) using zero-phase digital filters (obtained by forward-backward filtering, instead). 2) by solving a PLSO problem. The IIR digital filters can be approximated using well-formulated and well-understood classical analog filters [41, 40]. The most widely used classes of frequency-selective continuous-time filters are those referred to as Butterworth, Chebyshev, and elliptic filters [29, 40].

Let us consider a frequency-selective continuous-time (CT) filter defined with magnitude-squared characteristics,

---

<sup>2</sup>Note that quadratic variation regularization is not equivalent to total variation regularization as the former is a linear smoothing filter and the latter is a non-linear smoothing filter.

$|H_n(\omega)|^2$  [40, chapter 2]:

$$|H_n(\omega)|^2 = \frac{1}{1 + \epsilon^2 F_n(\omega) F_n^*(\omega)}, \quad (3)$$

where  $n$  is the order of the filter,  $*$  denotes the conjugate operator and  $F_n(\omega)$  is a polynomial function in  $\omega$ :

$$F_n(\omega) = \sum_{i=0}^n \alpha_i \omega^i, \quad (4)$$

where  $\alpha_i$  are some constants. For Butterworth filter,  $F_n(\omega)$  is monomial:

$$F_n^B(\omega) = \omega^n,$$

where the superscript  $B$  stands for Butterworth. For Chebyshev filter, it is as [40]

$$\begin{cases} F_0^C(\omega) = 1 \\ F_1^C(\omega) = \omega \\ F_{n+1}^C(\omega) = 2\omega F_n^C(\omega) - F_{n-1}^C(\omega) \end{cases} \quad (5)$$

Considering (3) as the base equation for the filter derivation (i.e.,  $G(\omega) = |H_n(\omega)|^2$ ), the frequency response is real so the phase is zero. In the Fourier domain, the output of the zero-phase filter  $G(\omega)$  is expressed as:

$$\hat{X}(\omega) = G(\omega)Y(\omega) = H_n(\omega)H_n^*(\omega)Y(\omega) \quad (6)$$

Under suitable conditions (Dirichlet condition),  $x$  is determined by  $\hat{x}$  via the inverse Fourier transform of (6):

$$\hat{x}(t) = h_n(t) * h_n(-t) * y(t) \quad (7)$$

where  $*$  denotes the convolution operator. Following (7), the impulse response of the filter is  $h_n(t) * h_n(-t)$ . Since the impulse response is even (i.e., the transfer function is real), it is a zero-phase non-causal filter. Therefore, the standard approach to design (7) is obtained by the following procedure [28, 29]: a) polynomial root finding and identification of the impulse response of the causal continuous-time filter,  $h_n(t)$ , b) discretizing the impulse response using a transform method (e.g. bilinear or impulse invariance transform), c) applying forward filtering backward smoothing to avoid nonlinear phase response. The forward filtering is suitable for online filtering, however, as discussed in the introduction, it is not linear-phase which limits its application to special applications where phase distortions are of

less significance. In other applications which demand no phase distortion, backward smoothing is applied after forward filtering. In the following sections, we provide a mathematical proof that zero-phase forward-backward filtering can be represented as instances of PLSO problem. Consequently, the optimization problem can be applied directly in the time domain. As a result, there is no need for identifying the causal continuous-time impulse response in the frequency domain.

### 3. Zero-phase filtering: a PLSO problem

In the previous section, the standard approach for zero-phase digital filter design was reviewed. However, it needs to resort to the frequency domain followed by a forward-backward filtering scheme. In this section, we show that many classical filter design schemes have a constrained least-squares interpretation. Therefore, the zero-phase filter can be directly designed in the time domain without resorting to the frequency domain. We first state the following theorem.

**Theorem 1.** *Any zero-phase filter with frequency response defined by (3) can be presented as an instance of the following PLSO problem:*

$$x_{opt}(t) = \underset{x(t)}{\operatorname{argmin}} \|y(t) - x(t)\|^2 + \lambda \|f_n(t) * x(t)\|^2, \quad (8)$$

where  $\lambda$  is the regularization factor,  $\|\cdot\|^2$  denotes the Euclidean norm and  $f_n(t)$  is the inverse Fourier transform of (4):

$$f_n(t) = \sum_{i=0}^n \alpha_i D^i \delta(t) \quad (9)$$

The differentiation operator  $D^i$ , which represents  $d^i/dt^i$  is the  $i$ -th derivative and  $\delta(t)$  is the Dirac delta function.

To prove the theorem, first we state the following lemma.

**Lemma 2.** *If  $\mathcal{K} = \|f_n(t) * x(t)\|^2$ , then  $\frac{\partial \mathcal{K}}{\partial x(t)} = 2f_n(-t) * f_n(t) * x(t)$ .*

#### Proof

$\mathcal{K}$  can be expressed as

$$\mathcal{K} = \int_{-\infty}^{\infty} \left[ \int_{-\infty}^{\infty} f_n(u - \tau) x(\tau) d\tau \right]^2 du$$

The derivative of  $\mathcal{K}$  with respect to  $x$  is equal to

$$\frac{\partial \mathcal{K}}{\partial x(t)} = 2 \int_{-\infty}^{\infty} f_n(u - t) \underbrace{\int_{-\infty}^{\infty} f_n(u - \tau) x(\tau) d\tau}_{f_n(u) * x(u)} du \quad (10)$$

Defining  $\psi(t) = f_n(t) * x(t)$ , (10) is expressed as

$$\frac{\partial \mathcal{K}}{\partial x(t)} = 2 \underbrace{\int_{-\infty}^{\infty} f_n(u-t)\psi(u)du}_{f_n(-t)*\psi(t)} = 2f_n(-t) * \psi(t)$$

Substituting  $\psi(t)$  with  $f_n(t) * x(t)$  we find

$$\frac{\partial \mathcal{K}}{\partial x(t)} = 2f_n(-t) * f_n(t) * x(t),$$

which proves the Lemma.

The optimal solution of (8) is found by setting its derivative with respect to  $x$  to zero, which leads to

$$-2 [y(t) - x_{opt}(t)] + 2\lambda f_n(-t) * f_n(t) * x_{opt}(t) = 0,$$

where  $x_{opt}$  denotes the optimal solution of (8). After some simplification, we find

$$x_{opt}(t) + \lambda f_n(t) * f_n(-t) * x_{opt}(t) = y(t) \quad (11)$$

Taking the Fourier transform of (11), we find

$$X_{opt}(\omega) = \frac{1}{1 + \lambda F_n(\omega)F_n^*(\omega)} Y(\omega) \quad (12)$$

The frequency response of (12) (i.e., the solution of (8)) is

$$G(\omega) = \frac{1}{1 + \lambda F_n(\omega)F_n^*(\omega)} \quad (13)$$

Comparing (13) and (3), we find that they are equivalent if  $\lambda = \epsilon^2$ . Hence, we conclude that the zero-phase filter transfer function is equivalent to the transfer function related to the solution of the PLSO problem defined by (8). It means that designing a zero-phase filter can be viewed as solving a PLSO problem.

On the other hand, by substituting (9) in (8), any zero-phase filter can be represented by the following optimization problem:

$$x_{opt}(t) = \underset{x(t)}{\operatorname{argmin}} \|y(t) - x(t)\|^2 + \epsilon^2 \left\| \sum_{i=0}^n \alpha_i D^i x(t) \right\|^2. \quad (14)$$

Denoting  $p(t) = \sum_{m=-\infty}^{\infty} \delta(t - mT_s)$ , the impulse train, with the sampling period  $T_s$ , the discrete time signal  $x_p(t)$ , sampled at sampling period  $T_s$ , is deduced of  $x(t)$  as [28]:

$$x_p(t) = x(t)p(t) = \sum_{m=-\infty}^{\infty} x[mT_s]\delta(t - mT_s)$$

Accordingly (1) is expressed in discrete time (DT) domain as

$$y_p(t) = x_p(t) + v_p(t)$$

$x_p(t)$  can also be represented as a linear combination of these basis functions:

$$x_p(t) = \sum_{m=-\infty}^{\infty} c_m \phi_m(t),$$

where  $c_m = x[mT_s]$  and  $\phi_m(t) = \delta(t - mT_s)$ . Therefore, the smoothing problem defined in (14), changes to the estimate of the parameters  $c_m$ :

$$c_{opt,m} = \underset{c_m}{\operatorname{argmin}} \left\| y_p(t) - \sum_{m=-\infty}^{\infty} c_m \phi_m(t) \right\|^2 + \epsilon^2 \left\| \sum_{i=0}^n \alpha_i D^i \sum_{m=-\infty}^{\infty} c_m \phi_m(t) \right\|^2 \quad (15)$$

where  $c_{opt,m} = x_{opt}[mT_s]$ . It means that the task of a zero-phase filter in the time domain is to smooth the sampled signal with a penalty on the derivatives of the model. It can be rewritten as

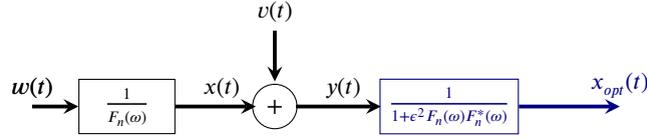
$$x_{opt}(t) = \underset{x_p(t)}{\operatorname{argmin}} \left\| y_p(t) - x_p(t) \right\|^2 + \epsilon^2 \left\| \sum_{i=0}^n \alpha_i D^i x_p(t) \right\|^2$$

#### 4. An equivalent Wiener smoother for zero-phase filters

In the previous section, we showed that the zero-phase filter can be viewed as a PLSO problem. In this section, we show that the zero-phase filter and PLSO are both special cases of the optimal Wiener filter. To this purpose, we consider the linear state-space model corresponding to (8) which can be represented as [24]

$$\begin{aligned} f_n(t) * x(t) &= w(t) \\ y(t) &= x(t) + v(t) \end{aligned} \quad (16)$$

where  $v(t)$  is known as the observation noise,  $w(t)$  is an additive zero-mean random term and known as the process (model) noise. The model is depicted in Fig. 1 with black color. Consider the problem of optimal (Wiener) filtering in



**Figure 1:** An equivalent Wiener filter structure for zero-phase filters defined by (3).

which the aim is to find a filter  $g_W(t)$  such that its output,  $x_{opt}(t)$  matches  $x(t)$  "as best as possible" in the least-squares sense:

$$x_{opt}(t) = g_W(t) * y(t) \approx x(t).$$

The filter structure is shown in Fig. 1 via blue color. The optimum Wiener filter is [47]

$$G_W(\omega) = \frac{S_{ww}(\omega)}{F_n(\omega)S_{vv}(\omega)F_n^*(\omega) + S_{ww}(\omega)} \quad (17)$$

where  $S_{ww}(\omega)$  and  $S_{vv}(\omega)$  are the PSD of  $w$  and  $v$ , respectively. Equation (17) can be written as

$$G_W(\omega) = \frac{1}{1 + \frac{S_{vv}(\omega)}{S_{ww}(\omega)} F_n(\omega)F_n^*(\omega)}. \quad (18)$$

Consider a specific case where the observation and process noises are Gaussian white noises with variances  $\sigma_v^2$  and  $\sigma_w^2$ , respectively. In this specific case, (18) can be expressed as

$$G_W(\omega) = \frac{1}{1 + \frac{\sigma_v^2}{\sigma_w^2} F_n(\omega)F_n^*(\omega)}. \quad (19)$$

Comparing (19) with (3) and (13), we conclude that the zero-phase filters and PLSO are special cases of Wiener filter when the parameters are chosen as

$$\lambda = \epsilon^2 = \frac{\sigma_v^2}{\sigma_w^2}. \quad (20)$$

It means that the zero-phase filters and PLSO algorithms are optimal when the signal is a random process obtained from a specific AR model driven by input white noise in the presence of white observation noise. In other cases, when the noises are not white (colored spectral noises), (18) suggests that the regularization factor should be a function of

the PSD of noises:

$$\lambda = \frac{S_{vv}(\omega)}{S_{ww}(\omega)}. \quad (21)$$

## 5. A block-wise matrix formulation for zero-phase filter

As discussed above, the standard solution for designing zero-phase filter needs to resort to frequency domain. In section 3, we showed that the zero-phase filter can be viewed as a PLSO problem. In this section, we present a block-wise matrix formulation for designing zero-phase filters directly in the time domain. Two transform methods, namely step invariance (or ZOH-) transform [26, 35] and bilinear transform are employed to the design of the procedure. In DT, (16) is expressed as [24]

$$\begin{aligned} f_{n,k} * x_k &= w_k \\ y_k &= x_k + v_k \end{aligned} \quad (22)$$

The desired signal  $x_{opt,k}$  can be obtained by solving the following PLSO problem:

$$x_{opt,k} = \underset{x_k}{\operatorname{argmin}} \sum_{j=1}^L (y_j - x_j)^2 + \lambda \sum_{j=2}^L \left( f_{n,j} * x_j \right)^2 \quad (23)$$

### 5.1. Step invariance transform based matrix formulation

The step invariance technique is an extension of the impulse invariance transform method, in which the response of the analog system is obtained by assuming that the input signal is approximated by a sequence of steps.

**Lemma 3.** For any function  $f_n(t)$  defined by (9), a discrete form of  $f_n(t)$  using step invariance is

$$f_{n,k} = \sum_{i=0}^n \beta_i \delta_{k-i}, \quad (24)$$

where  $\beta_i = (-1)^i \sum_{j=i}^n \alpha_j \binom{j}{i}$  and  $\delta_{k-i}$  is the shifted delta function.

*Proof of Lemma 3:* see Appendix 8.1.

Now, putting (24) in (23) yields

$$x_{opt,k} = \underset{x_k}{\operatorname{argmin}} \sum_{j=1}^L (y_j - x_j)^2 + \lambda \sum_{j=2}^L \left( \sum_{i=0}^n \beta_i x_{j-i} \right)^2$$

Denoting  $\mathbf{x} = [x_1, \dots, x_L]^T$ ,  $\mathbf{y} = [y_1, \dots, y_L]^T$ , the previous equation becomes:

$$\mathbf{x}_{opt} = \underset{\mathbf{x}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{x}\|^2 + \lambda \|\Lambda_n \mathbf{x}\|^2, \quad (25)$$

where  $\Lambda_n$  is

$$\Lambda_n = \begin{pmatrix} \beta_n & \dots & \beta_1 & \beta_0 & 0 & \dots & 0 \\ 0 & \beta_n & \dots & \beta_1 & \beta_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \beta_0 & 0 \\ 0 & \dots & 0 & \beta_n & \dots & \beta_1 & \beta_0 \end{pmatrix}.$$

By setting the derivative of (25) with respect to  $x$  equal to zero, we get the following matrix equation:

$$(I + \lambda \Lambda_n^T \Lambda_n) \mathbf{x}_{opt} = \mathbf{y},$$

where  $\Lambda_n^T$  denotes the transpose of the matrix  $\Lambda_n$ . If  $(I + \lambda \Lambda_n^T \Lambda_n)$  is invertible, the optimal solution is

$$\mathbf{x}_{opt} = (I + \lambda \Lambda_n^T \Lambda_n)^{-1} \mathbf{y} \quad (26)$$

We refer to  $(I + \lambda \Lambda_n^T \Lambda_n)^{-1}$  as the *smoothing matrix* using step invariance.

## 5.2. Bilinear transform based matrix formulation

In this section, the bilinear transform [12] is employed to implement zero-phase filters directly in the DT domain.

We first make the following observation:

**Lemma 4.** *A discrete form of  $f_n(t)$  using bilinear transform is described with the following linear time invariant (LTI) system:*

$$\sum_{i=0}^n \binom{n}{i} f_{n,k-i} = \sum_{i=0}^n \gamma_i \delta_{k-i}, \quad (27)$$

where the parameters  $\gamma_i$  are determined from the following system equation:

$$\sum_{i=0}^n \gamma_i z^{-i} = \sum_{i=0}^n \alpha_i \left( \frac{2}{T_s} \right)^i \binom{n}{i} (1 - z^{-1})^i (1 + z^{-1})^{n-i}$$

**Proof.** See Appendix 8.2.

Equation (27) can be expressed as

$$\mathcal{A}(z) f_{n,k} = \Gamma(z) \delta_k, \quad (28)$$

where  $z^{-i}f_{n,k} = f_{n,k-i}$  and

$$\mathcal{A}(z) = \sum_{i=0}^n \binom{n}{i} z^{-i} \quad \text{and} \quad \Gamma(z) = \sum_{i=0}^n \gamma_i z^{-i}$$

If (28) is causal and invertible, then it can be represented by the following AR model [31, Chapter 2]:

$$f_{n,k} = \frac{\Gamma(z)}{\mathcal{A}(z)} \delta_k \quad (29)$$

Convolving both sides of (29) with  $x_k$  yields

$$f_{n,k} * x_k = \frac{\Gamma(z)}{\mathcal{A}(z)} x_k \quad (30)$$

Putting (30) in (23), we have

$$x_{opt,k} = \underset{x_k}{\operatorname{argmin}} \sum_{j=1}^L (y_j - x_j)^2 + \lambda \sum_{j=1}^L \left( \frac{\Gamma(z)}{\mathcal{A}(z)} x_j \right)^2, \quad (31)$$

which can equivalently be represented in the following optimization problem [22]:

$$\mathbf{x}_{opt} = \underset{\mathbf{x}}{\operatorname{argmin}} \|\Upsilon_n(\mathbf{y} - \mathbf{x})\|^2 + \lambda \|\Gamma_n \mathbf{x}\|^2 \quad (32)$$

The optimal solution of (32) is

$$\mathbf{x}_{opt} = (\Upsilon_n^T \Upsilon_n + \lambda \Gamma_n^T \Gamma_n)^{-1} \Upsilon_n^T \Upsilon_n \mathbf{y}, \quad (33)$$

where  $\Gamma_n$  is

$$\Gamma_n = \begin{pmatrix} \gamma_n & \cdots & \gamma_1 & \gamma_0 & 0 & \cdots & 0 \\ 0 & \gamma_n & \cdots & \gamma_1 & \gamma_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \gamma_0 & 0 \\ 0 & \cdots & 0 & \gamma_n & \cdots & \gamma_1 & \gamma_0 \end{pmatrix}$$

and Pascal's triangle gives us the elements of  $\Upsilon_n$  for the expanded binomial of the form  $(1 + z^{-1})^n$ :

$$\Upsilon_n = \begin{pmatrix} \binom{n}{0} & \binom{n}{1} & \dots & \binom{n}{n} & 0 & \dots & 0 \\ 0 & \binom{n}{0} & \binom{n}{1} & \dots & \binom{n}{n} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \binom{n}{n} & 0 \\ 0 & \dots & 0 & \binom{n}{0} & \binom{n}{1} & \dots & \binom{n}{n} \end{pmatrix}$$

$\Upsilon_n$  can also be defined as the Toeplitz matrix form of  $\mathbf{d}_n$  where  $\mathbf{d}_n$  is defined by the following recursion:

$$\begin{cases} \mathbf{d}_1 \triangleq (1, 1) & n = 1 \\ \mathbf{d}_n = \mathbf{d}_{n-1} * \mathbf{d}_1 & n > 1 \end{cases}$$

We refer to  $(\Upsilon_n^T \Upsilon_n + \lambda \Gamma_n^T \Gamma_n)^{-1} \Upsilon_n^T \Upsilon_n$ , as the *smoothing matrix* using bilinear transform. In the next section, several applications of the proposed framework are presented.

### 5.3. An FIR filter

The smoothing matrices  $(\Upsilon_n^T \Upsilon_n + \lambda \Gamma_n^T \Gamma_n)^{-1} \Upsilon_n^T \Upsilon_n$  and  $(I + \lambda \Lambda_n^T \Lambda_n)^{-1}$  are symmetric, positive definite. Using Cholesky decomposition, they can be decomposed as the product  $LL^T$  where L is a lower triangular matrix. Of course,  $LL^T$  is symmetric too, i.e.,  $LL^T = L^T L$ . Consequently, (26) and (33) can be written as

$$\mathbf{x}_{opt} = L^T L \mathbf{y} \quad (34)$$

Substituting  $L^T$  with JLJ, (34) can be expressed as

$$\mathbf{x}_{opt} = JLJ \mathbf{y} \quad (35)$$

where J is the exchange matrix, a ‘‘row or column-reversed’’ version of the identity matrix. The proposed smoothing filters can be implemented as a combination of a forward filtering the input signal  $\mathbf{y}$  with the FIR filter L (as L is a lower triangular matrix) and backward filtering with the same filter. The forward filtering output at time  $k$  is obtained as

$$x_{opt,k}^f = \sum_{i=0}^{k-1} L(k, k-i) y_{k-i}, \quad (36)$$

where superscript  $f$  stands for forward filtering. The proposed FIR filter has the advantage to be inherently stable.

## 6. Applications

The proposed framework can find various applications in engineering science, such as appropriate cutoff frequency selection for zero-phase filters, frequency analysis of PLSO algorithms, band-pass smoothing filter, and zero-phase Chebyshev filter design.

### 6.1. Precise cutoff frequency selection

In many applications of signal processing such as ECG denoising, LTI filters are still the standard choice for ECG system front-ends, as they impose weak assumptions on the signals [46]. The performance of these filters depends on the choice of cutoff frequency. However, in actual applications, the cutoff frequency is unknown. Therefore, an efficient algorithm for cutoff frequency selection remains an important and challenging issue. In this section, we consider a zero-phase Butterworth filter and present an efficient algorithm for selecting its cutoff frequency. Although the proposed technique is presented for Butterworth cutoff frequency selection, its principles are easy to extend to other zero-phase filters. Recall that the original Butterworth filter of order  $n$  is defined by [28]

$$|H_n^B(\omega)|^2 = \frac{1}{1 + (\frac{\omega}{\omega_c})^{2n}}. \quad (37)$$

According to Theorem 1, by setting  $\lambda = 1/\omega_c^{2n}$ ,  $F_n^B(\omega) = \omega^n$  and  $f_n^B(t) = D^n \delta(t)$ , the zero-phase Butterworth filter can be designed as

$$x_{opt}(t) = \underset{x(t)}{\operatorname{argmin}} \int [y(\tau) - x(\tau)]^2 d\tau + \lambda \int [f_n^B(\tau) * x(\tau)]^2 d\tau. \quad (38)$$

Therefore, the appropriate choice of cutoff frequency is related to the optimal value of regularization factor  $\lambda$ . The discrete Butterworth filter is also the solution of the following PLSO

$$x_{opt,k} = \underset{x_k}{\operatorname{argmin}} \sum_{j=1}^L (y_j - x_j)^2 + \lambda \sum_{j=1}^L [f_{n,j}^B * x_j]^2, \quad (39)$$

where,  $f_{n,j}^B$  can be obtained either with step invariance or bilinear transform. Note that  $F_n^B(\omega) = \omega^n$  and its Laplace transform is  $F_n^B(s) = s^n$ . If we use bilinear transform, then we have

$$F_n^B(s) = s^n \implies F_n^B(z) = \frac{\Gamma(z)}{\mathcal{A}(z)} = \left( \frac{1 - z^{-1}}{1 + z^{-1}} \right)^n$$

According to (28)-(32), we can express (39) in the following matrix notation

$$\mathbf{x}_{opt} = \underset{\mathbf{x}}{\operatorname{argmin}} \left\| \Upsilon_n(\mathbf{y} - \mathbf{x}) \right\|^2 + \lambda \left\| \Gamma_{B,n} \mathbf{x} \right\|^2$$

The optimal solution is

$$\mathbf{x}_{opt} = (\Upsilon_n^T \Upsilon_n + \lambda \Gamma_{B,n}^T \Gamma_{B,n})^{-1} \Upsilon_n^T \Upsilon_n \mathbf{y}, \quad (40)$$

where the operator  $\Upsilon_n$  was defined before and  $\Gamma_{B,n}$  is the matrix:

$$\Gamma_{B,n} = \begin{pmatrix} \beta_n^B & \dots & \beta_1^B & \beta_0^B & 0 & \dots & 0 \\ 0 & \beta_n^B & \dots & \beta_1^B & \beta_0^B & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \beta_0^B & 0 \\ 0 & \dots & 0 & \beta_n^B & \dots & \beta_1^B & \beta_0^B \end{pmatrix}$$

where the entries  $\beta_i^B$  are the coefficients of the binomial expansion of  $(1 - z^{-1})^n$ :

$$\beta_i^B = (-1)^i \binom{n}{i}.$$

$\Gamma_{B,n}$  can also be defined as the Toeplitz matrix form of  $\mathbf{b}_n$  where  $\mathbf{b}_n$  is defined by the following recursion:

$$\begin{cases} \mathbf{b}_1 \triangleq (1, -1) & n = 1 \\ \mathbf{b}_n = \mathbf{b}_{n-1} * \mathbf{b}_1 & n > 1 \end{cases}$$

We refer to  $(\Upsilon_n^T \Upsilon_n + \lambda \Gamma_{B,n}^T \Gamma_{B,n})^{-1} \Upsilon_n^T \Upsilon_n$ , as the  $n$ -th order Butterworth *smoothing matrix* using bilinear transform<sup>3</sup>.

## 6.2. Computation of the hyperparameter $\lambda$

The bi-criterion problem discussed in [4, Ch. 6] can be used to find the optimal regularization factor,  $\lambda_{opt}$ . The bi-criterion problem is

$$\operatorname{minimize} \text{ (w.r.t } R_+^2) \left( \left\| \Upsilon_n(\mathbf{y} - \mathbf{x}_{opt}) \right\|^2, \left\| \Gamma_{B,n} \mathbf{x}_{opt} \right\|^2 \right) \quad (41)$$

The optimal regularization factor,  $\lambda_{opt}$ , is found using L-curve which is obtained by plotting  $\left\| \Upsilon_n(\mathbf{y} - \mathbf{x}_{opt}) \right\|$  (the norm of the residual term) versus  $\left\| \Gamma_{B,n} \mathbf{x}_{opt} \right\|$  (the norm of the regularized term) as  $\lambda$  varies over  $(0, \infty)$ . Note that  $\lambda$  is used

<sup>3</sup>When step invariance is used to implement the discrete Butterworth filter,  $\mathcal{A}(z) = 1$  and consequently  $\Upsilon_n = I$ .

to compute  $\mathbf{x}_{opt}$ . As a primary example, we use the procedure for a specific case (record 119m from the PhysioNet Computing in Cardiology Challenge 2014 [2]). Fig. 2 shows the original ECG (blue color) and its noisy signal (red color) with SNR = 5 dB. The L-curve corresponds to the noisy ECG plotted with green color. The  $\lambda$  corresponding to the corner of the curve (i.e., the point at which  $\|\Upsilon_n(\mathbf{y} - \mathbf{x}_{opt})\| + \|\Gamma_{B,n}\mathbf{x}_{opt}\|$  is minimized) can be chosen as the optimal regularization factor. We denote this value with  $\lambda_{opt}$ . A motivation for choosing  $\lambda = \lambda_{opt}$  is that it seeks to balance the regularized term and the residual error in the estimated signal  $\mathbf{x}_{opt}$  due to noises in the measurement  $\mathbf{y}$ . In other cases when  $\lambda$  is small, then the estimated signal  $\mathbf{x}_{opt}$  of (40) is of huge norm and is likely to be contaminated by the propagated error that stems from noises in  $\mathbf{y}$ . Conversely, when  $\lambda$  is large, the estimated signal is a poor approximation of a solution of  $\mathbf{x}_{opt} = \mathbf{y}$  and the associated residual error term is of large norm. We refer the interested readers to [18, 19]. Now, the optimal regularization factor can be used to find the appropriate cutoff frequency in DT. To this purpose, we need to find the relation between cutoff frequency and regularization factor. In order to find the relation between them, we write (40) in the following form [22]:

$$x_{opt,k} = (h_{n,-k} * h_{n,k} + \lambda b_{n,-k} * b_{n,k})^{-1} * h_{n,-k} * h_{n,k} * y_k \quad (42)$$

The impulse response of (42) is obtained by substituting  $y_k$  with  $\delta_k$  [22]. The frequency response is obtained by taking the  $Z$ -transform of impulse response. Hence, the impulse and frequency responses are<sup>4</sup>

$$\begin{aligned} g_{n,k} &= (h_{n,-k} * h_{n,k} + \lambda b_{n,-k} * b_{n,k})^{-1} * h_{n,-k} * h_{n,k} \\ G_n^{BT}(z) &= \frac{(1 + z^{-1})^n (1 + z)^n}{(1 + z^{-1})^n (1 + z)^n + \lambda (1 - z^{-1})^n (1 - z)^n} \end{aligned} \quad (43)$$

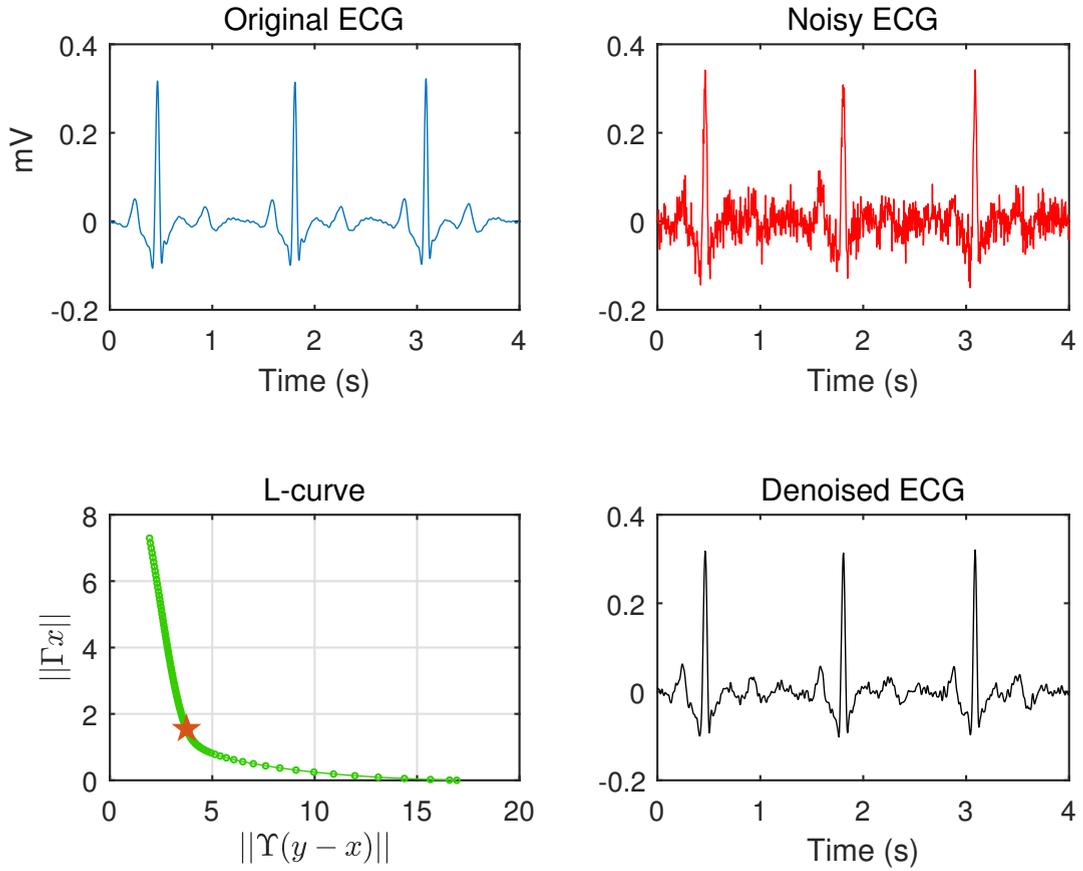
where the “ $BT$ ” superscript stands for “Bilinear Transformation”. Eq. (43) becomes in the Fourier domain

$$G_n^{BT}(e^{j\omega}) = \frac{1}{1 + \lambda \tan^{2n} \frac{\omega}{2}} \quad (44)$$

The optimal value of regularization factor corresponding to  $-6$  dB cutoff frequency  $f_c$  is found by setting (44) equal to  $1/2$ , which leads to

$$\lambda_{opt} = \frac{1}{\tan^{2n} \frac{\omega_c}{2}}. \quad (45)$$

<sup>4</sup>Please note that  $H_n(z) = (1 + z^{-1})^n$  and  $b_n(z) = (1 - z^{-1})^n$ .



**Figure 2:** A segment of original ECG, record 119m from MIT-BIH Atrial Fibrillation Database [2] (blue), noisy ECG (red), the optimal trade-off L-curve using the proposed approach (green) and denoised ECG using zero-phase Butterworth filter (black).

Equation (45) shows that the optimal regularization factor is related to cutoff frequency. Finally the appropriate cutoff frequency is found by

$$f_c = \frac{f_s}{\pi} \arctan\left(\frac{1}{2^q/\lambda_{opt}}\right). \quad (46)$$

where  $f_s = 1/T_s$ . The denoised ECG using zero-phase Butterworth filter with the optimal cutoff frequency is plotted in Fig. 2 with black color. In the following the Butterworth filter is employed to ECG signal denoising and the proposed approach is used to find the optimal cutoff frequency. We show that the L-curve can be used for precise cutoff frequency selection.

### 6.3. Experimental result

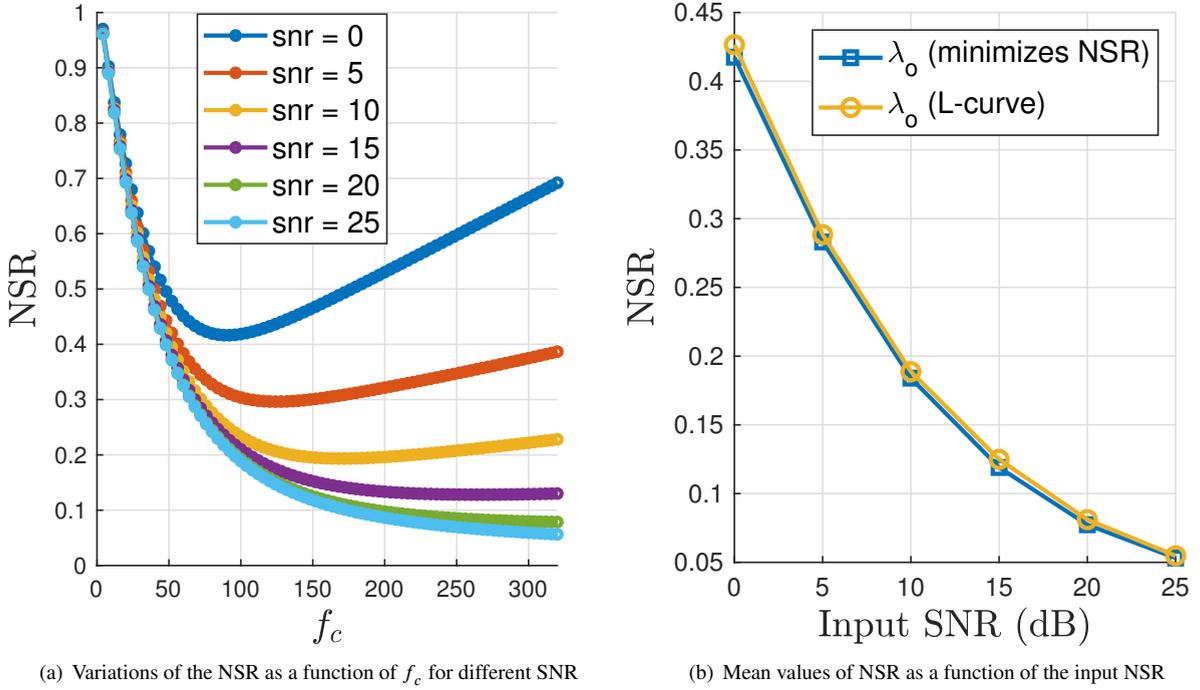
In order to evaluate the performance of the proposed method, we tested the approach over the PhysioNet PTB Diagnostic ECG Database [2]. The database contains 549 records from 290 subjects. Each record consists of twelve conventional ECG leads plus the three Frank's ones, sampled at 1kHz with 16-bit resolution. The noise was added to the original signals with varying SNR (from 0 to 25 dB). Then the ECG signals were reconstructed using Butterworth filter. We compared the Butterworth filter on this problem in two cases. First when bilinear transform is used to Butterworth filter design. In this case, we compare the optimal regularization factor obtained using L-curve and the one that minimizes the reconstruction error. Secondly, we compare bilinear transform based approach and step invariance based approach for Butterworth filter design. In this case, the L-curve is used to find the optimal regularization factor for both approaches. To quantify the performance of the methods, we employed noise to signal ratio (NSR), i.e., the ratio between the power of the reconstruction error and the power of the original signal, given by [22]

$$\text{NSR} = \sqrt{\frac{\sum_k (x_k - x_{opt,k})^2}{\sum_k x_k^2}}.$$

In the first experiment, we employ Butterworth filter (estimated using bilinear transform) to ECG denoising with two methods for cutoff frequency selection. First we choose the cutoff frequency that minimizes the NSR as optimal cutoff frequency. As an example, in Fig. 3(a), we plot the NSR variation against the choice of the cutoff frequency for different SNR. The curves are similar to the L-curve. Second, we use L-curve to find the optimal cutoff frequency. The mean of NSR versus different input SNRs achieved over the ECG segments are plotted in Fig. 3(b). Compared to the one that minimizes the NSR, the results are close when the cutoff frequency is approximated using L-curve.

Finally, we compare step invariance and bilinear transform for designing Butterworth filter for ECG denoising. For both methods, we employed the L-curve to find the optimal regularization factor, i.e., the cutoff frequency. In Fig. 4, we plot the mean values of NSR as a function of the input NSR for Butterworth filter designed using these methods. Clearly, the two design methods give very good and similar results, even if the bilinear transform provides a slightly better signal reconstruction, especially for high input SNR than the step invariance method.

In the last part of our experiments, we employ the proposed FIR filter (34) presented in Section 5.3 for the design and implementation of an FIR Butterworth filter. We compare our proposed FIR Butterworth filter with the traditional implementation of IIR Butterworth filter (using the Matlab command: *filter*). The NSRs obtained using both methods are reported in Figure 5(a). Figure 5(a) shows that the obtained NSR for the proposed FIR Butterworth is less than the obtained NSR for the traditional IIR Butterworth. Since the causal filters induces delay, we also compared them using cross-correlation, where a good reconstruction is related to a cross-correlation close to 1. The results (Fig. 5(b)) show that the proposed FIR Butterworth filter outperforms the traditional IIR Butterworth filter.



**Figure 3:** Comparing the optimal regularization factor obtained using L-curve and the one that minimizes the reconstruction error. a) The variation of NSR as a function of  $f_c$ . b) Mean values of NSR for signal reconstruction by Butterworth, using the cutoff frequency that minimizes the NSR (blue color) and the one that is obtained using L-curve (yellow color).

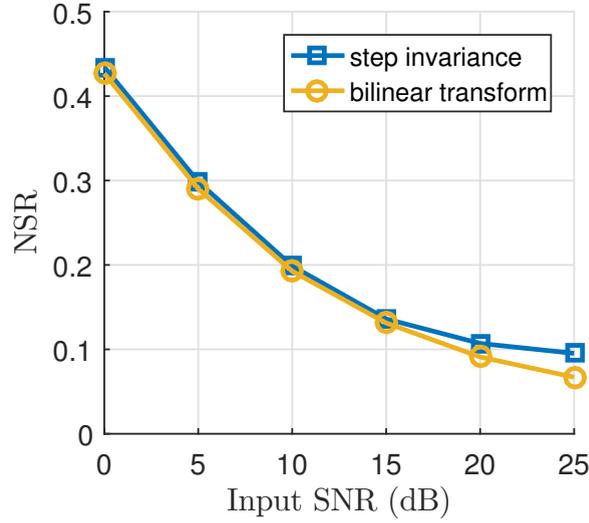
#### 6.4. Frequency analysis of PLSO algorithms

Let us consider the PLSO problem defined by (8). Its solution depends on the regularization factor which can be found using L-curve function. However, in some cases, we need to extract a signal within a predetermined frequency band. In this specific case, a closed-form expression can be found for the regularization factor using the frequency response of the PLSO as discussed in the following. According to (13), we have

$$G(\omega) = \frac{1}{1 + \lambda F_n(\omega) F_n^*(\omega)} \quad (47)$$

The optimal value of the regularization factor, corresponding to  $-6$  dB cutoff frequency  $\omega_c$ , is found by setting (47) equal to  $1/2$ , which leads to

$$\lambda_{opt} = \frac{1}{F_n(\omega_c) F_n^*(\omega_c)}. \quad (48)$$



**Figure 4:** Mean values of NSR for signal reconstruction by Butterworth using bilinear transform and stepinvariance, as a function of the input SNR.

It shows that the regularization factor is related to the cutoff frequency. On the other hand, (47) can be expressed in the Laplace domain

$$G(s) = \frac{1}{1 + \lambda F_n(s)F_n(-s)} \quad (49)$$

The filter (49) is non-causal but can be factorized as the cascade of a minimum-phase system (considering the causal poles and zeros) and a maximum-phase system (considering the non-causal poles and zeros):

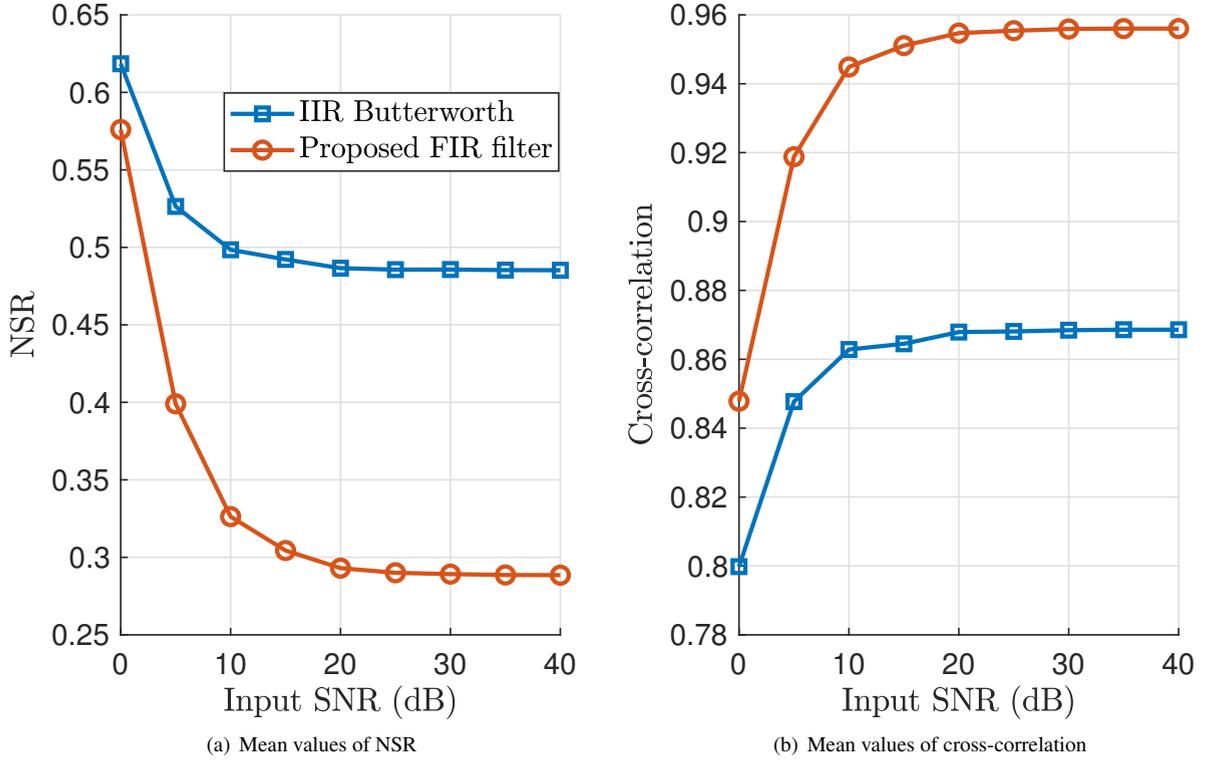
$$G(s) = H^{\text{causal}}(s)H^{\text{non-causal}}(s) = H(s)H(-s) \quad (50)$$

The minimum phase system can be implemented using causal filter and the whole procedure can be implemented using forward-backward filtering.

As an example, in signal denoising/smoothing [37, 45, 13, 34], PLSO is used to smooth the signals by considering the following penalty:

$$\text{PEN}_1(x) = \int \left[ \frac{d}{d\tau} x(\tau) \right]^2 d\tau = \|Dx(t)\|^2 \quad (51)$$

The algorithm is known as smoothness priors or quadratic variation (QV) denoising. For the  $n$ -th degree QV, the



**Figure 5:** Mean values of NSR and cross-correlation for signal reconstruction by Butterworth using Matlab command and the proposed FIR filter (34), as a function of the input SNR.

following PLSO is considered in one dimension [21]:

$$x_{opt}(t) = \underset{x(t)}{\operatorname{argmin}} \int [y(\tau) - x(\tau)]^2 d\tau + \lambda \int \left[ \frac{d^n}{d^n \tau} x(\tau) \right]^2 d\tau \quad (52)$$

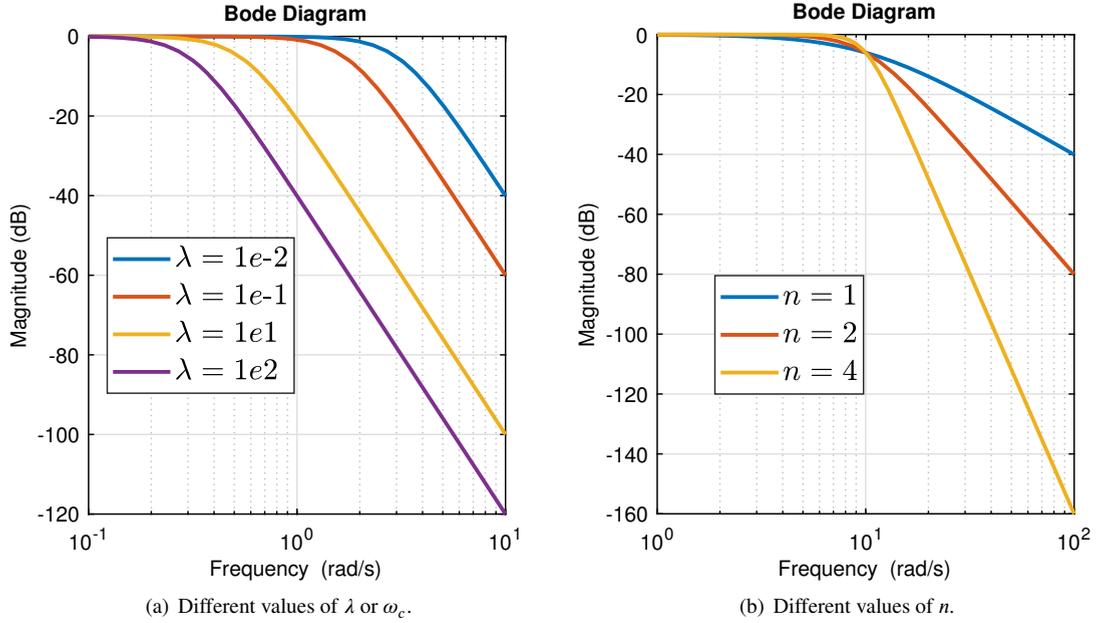
which is known as high degree QV (HDQV). Using the idea presented in previous sections, the solution can be expressed as

$$x_{opt}(t) = [\delta(t) + \lambda f_n^{QV}(-t) * f_n^{QV}(t)]^{-1} * y(t) \quad (53)$$

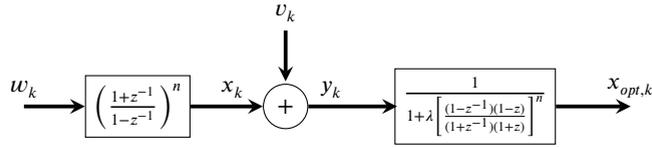
Taking the Laplace transform of (53), we obtain

$$\hat{X}_{opt}^{QV}(s) = \frac{1}{1 + \lambda F_n^{QV}(s) F_n^{QV}(-s)} Y(s), \quad (54)$$

where  $F_n^{QV}(s) = s^n$ . The interesting result is that  $F_n^{QV}(s) = F_n^B(s)$ . Hence, we conclude that the QV algorithm and



**Figure 6:** Amplitude response of the the HDQV, a) for different values of  $\lambda$  (or cutoff frequency) b) for different values of  $n$ .



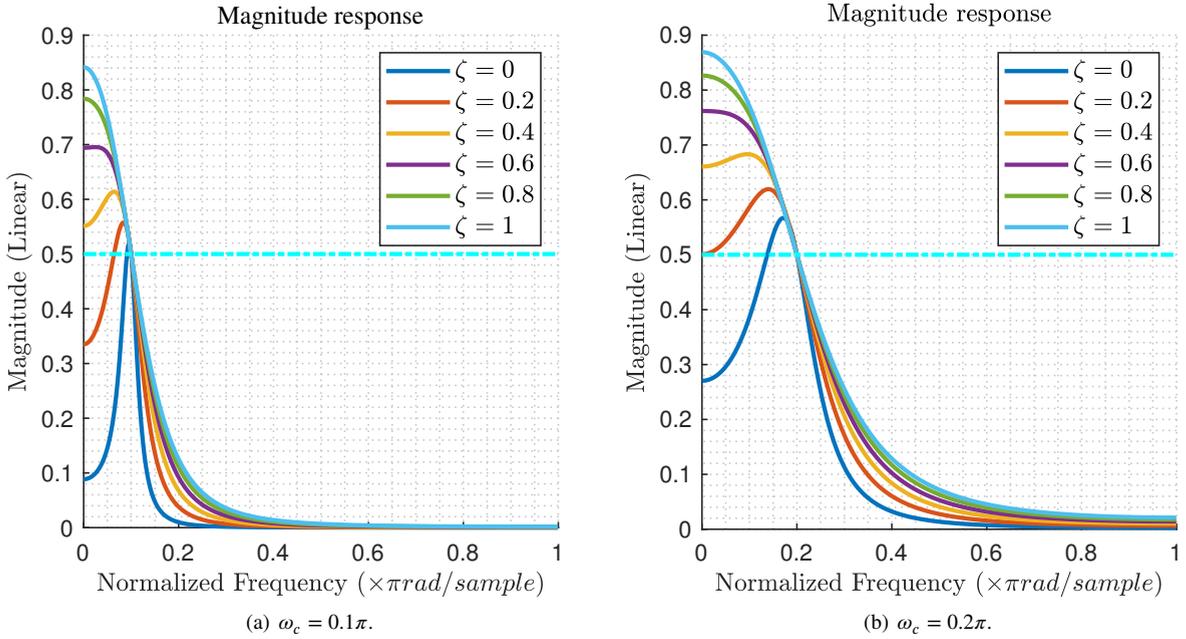
**Figure 7:** An equivalent Wiener smoother for zero-phase Butterworth filter or HDQV algorithm when bilinear transform is used for their DT implementation.

the Butterworth filter introduced in 1930 [6] leads to the same filter. The frequency response of (54) is

$$|H_n^{QV}(s)|^2 = \frac{1}{1 + \lambda(-1)^n s^{2n}}. \quad (55)$$

The frequency response of the first order HDQV for different values of  $\lambda$  is shown in Fig. 6(a).  $\lambda$  changes from 0.01 to 100. The result shows that it acts as a low-pass filter. As  $\lambda$  increases the cut-off frequency decreases. In Fig. 6(b), we plot the frequency response for  $\omega_0 = 10$  and different values of  $n$  (higher degree QV).

Finally, using the idea presented in this paper, Butterworth filter or HDQV algorithm can be viewed as an optimal Wiener filter. An equivalent Wiener filter scheme for the Butterworth filter or HDQV algorithm (when bilinear transform is used for implementation) is depicted in Fig. 7.



**Figure 8:** Amplitude response of the the PLSO, for different values of  $\zeta$ .

### 6.5. Zero-phase Band-pass filter

For analyzing periodic data, it would be more natural to use the harmonic acceleration operator [36] in the penalty term [32, chapter 5]:

$$\text{PEN}(x) = \left\| (D^2 + 2\zeta\omega_0 D + \omega_0^2)x(t) \right\|^2 \quad (56)$$

In the following, we analyze the PLSO problem with the penalty term (56), in both CT and DT domains. In CT domain, the frequency response of the PLSO with penalty term (56) is

$$G(s) = \frac{1}{1 + \lambda F(s)F(-s)}$$

$$F(s) = s^2 + 2\zeta\omega_0 s + \omega_0^2$$

In DT domain, both step invariance and bilinear transform can be used for discretizing (56). If the step invariance is used then (56) is converted to

$$\text{PEN}_n(x) = \left\| (a + bz^{-1} + z^{-2})x_k \right\|^2$$

where

$$a = 1 + \omega_0^2 + 2\zeta\omega_0$$

$$b = -2(1 + \zeta\omega_0)$$

In this case, its frequency response is expressed as

$$G^{SI}(z) = H^{SI}(z)H^{SI}\left(\frac{1}{z}\right) = \frac{1}{1 + \lambda F(z)F\left(\frac{1}{z}\right)} \quad (57)$$

$$F(z) = a + bz^{-1} + z^{-2}$$

where ‘‘SI’’ stands for step invariance. The product in (57) is equal to the cascade of a minimum phase system,  $H^{SI}(z)$ , considering the causal poles (the poles inside unit circle) and maximum phase system,  $H^{SI}\left(\frac{1}{z}\right)$ , considering the non-causal poles (the poles outside unit circle). The poles of (57) can be recognized simply by looking at the roots of the polynomial:

$$P(z) = 1 + \lambda F(z)F\left(\frac{1}{z}\right) = a\lambda z^4 + \lambda b(a+1)z^3 + [1 + \lambda(1+a^2+b^2)]z^2 + \lambda b(a+1)z + a\lambda \quad (58)$$

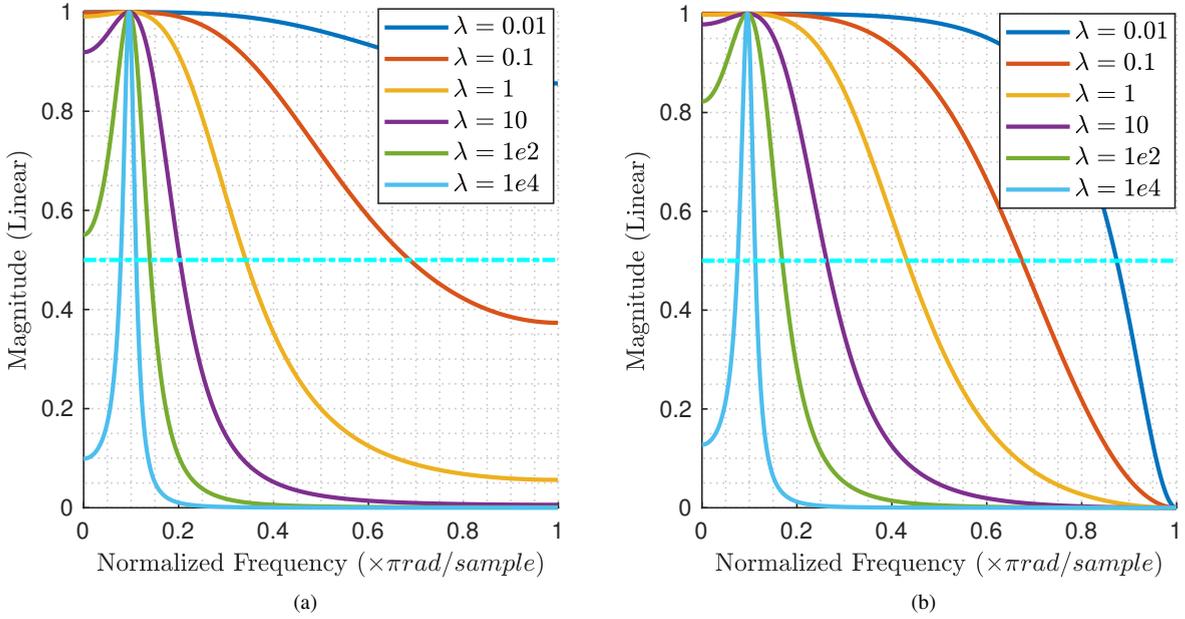
The filter transfer function is determined from the poles which are located inside the unit circle, and thus provide a stable system,  $H^{SI}(z)$ . In the Fourier domain, (57) is expressed as

$$G(e^{j\omega}) = \frac{1}{1 + \lambda(1+a^2+b^2+2b(a+1)\cos(\omega_0) + 2ac\cos(2\omega_0))} \quad (59)$$

The frequency response of the PLSO,  $G^{SI}(z)$ , for two specific values of  $\omega_0$  as a function of  $\zeta$  is depicted in Fig. 8. Note that we set the regularization factor such that  $|G^{SI}(e^{j\omega})| = \frac{1}{2}$ , which leads to the following regularization factor

$$\lambda = \frac{1}{1 + a^2 + b^2 + 2b(a+1)\cos(\omega_0) + 2ac\cos(2\omega_0)}$$

The quality factor  $Q$  of a band-pass smoothing filter is the ratio of the smoothing filter’s center frequency over its bandwidth. The value  $Q$  is a measure of the sharpness of a bandpass smoothing filter. A variable-Q filter is obtained by changing the regularization factor. The Q-factor is inversely related to the regularization factor. As  $\lambda$  increases the Q-factor decreases. The frequency response of the variable-Q filter using step invariance is depicted in Fig. 9(a).



**Figure 9:** Amplitude response of the the band-pass PLSO, for  $\omega_0 = 0.1$  and  $\zeta = 0$  for different values of  $\lambda$  a) using step invariance b) using bilinear transform.

**Table 1**

Penalty terms for different orders of the zero-phase Butterworth and Chebyshev filters.

Filter	$n$				
	1	2	3	4	5
Butterworth	$Dx(t)$	$D^2x(t)$	$D^3x(t)$	$D^4x(t)$	$D^5x(t)$
Chebyshev	$Dx(t)$	$(2D^2 - 1)x(t)$	$(4D^3 - 3D)x(t)$	$(8D^4 - 8D^2 + 1)x(t)$	$(16D^5 - 20D^3 + 4D)x(t)$

Finally, if we employ the bilinear transform, the frequency response of the PLSO is as

$$G^{BT}(z) = \frac{(1 + z^{-1})(1 + z)}{(1 + z^{-1})(1 + z) + \lambda F(z)F(\frac{1}{z})} \quad (60)$$

The frequency response of the variable-Q filter using bilinear transform is depicted in Fig. 9(b).

We note that although the optimal choice of regularization factor in sections 6.4 and 6.5 is linked to bandwidth, the factor can also be computed similar to those in section 6.1 using L-curve function.

## 6.6. Zero-phase Chebyshev Filter

Chebyshev filter is another example that can be studied in the hereby presented framework. Using the idea presented in previous sections, a zero-phase Chebyshev filter is the solution of (8) with the following penalty:

$$\text{PEN}_n(x) = \left\| f_n^C(t) * x(t) \right\|^2$$

where  $f_n^C(t)$  is found by taking the inverse of (5):

$$f_n^C(t) = 2Df_{n-1}^C(t) - f_{n-2}^C(t) \quad n \geq 2$$

$$f_0^C(t) = \delta(t)$$

$$f_1^C(t) = D\delta(t)$$

We refer to  $f_n^C(t)$  as the Chebyshev *smoothing operator*. The Chebyshev *discrete smoothing operator* is found by discretizing  $f_n^C(t)$ . In table (1), we compare the penalty term for different orders of Butterworth and Chebyshev in CT. If we employ step invariance, then the Chebyshev *discrete smoothing operator* is computed by the following recursion:

$$f_{n,k}^C = 2(1 - z^{-1})f_{n-1,k}^C - f_{n-2,k}^C \quad n \geq 2$$

$$f_{0,k}^C = \delta_k$$

$$f_{1,k}^C = (1 - z^{-1})\delta_k$$

Therefore, the following PLSO problem is obtained for the  $n$ -th order Chebyshev filter:

$$x_{opt,k} = \underset{x_k}{\operatorname{argmin}} \left\| y_k - x_k \right\|^2 + \epsilon^2 \left\| f_{n,k}^C * x_k \right\|^2$$

The optimal solution is

$$\mathbf{x}_{opt} = (I + \epsilon^2 \Gamma_{C,n}^T \Gamma_{C,n})^{-1} \mathbf{y} \quad (61)$$

where  $\Gamma_{C,n}$  is defined as the Toeplitz matrix form of  $c_n$  and  $c_n$  is defined by the following recursion:

$$\begin{cases} c_1 \triangleq (1, -1) & n = 1 \\ c_2 \triangleq (1, -4, 2) & n = 2 \\ c_n = 2c_1 * c_{n-1} - c_{n-2} & n > 2 \end{cases}$$

The transfer function of (61) is

$$G_n^C(z) = \frac{1}{1 + \epsilon^2 c_n(\frac{1}{z}) c_n(z)}$$

The zero-phase Chebyshev filter design using bilinear transform is straightforward.

## 7. Conclusion

This paper presented a framework for the unification of the PLSO and zero-phase filters. We showed that a zero-phase filter can be viewed as a least-squares data smoothing with impulse train and a penalty term on the derivatives of the fitted model. We proposed an FIR filter for implementing the zero-phase filters which has the advantage of being inherently stable. Based on the results of this paper, a unified framework was proposed for PLSO algorithm and zero-phase filter design in both time and frequency domain. Especially, we show that each type of filter is related to a particular penalty (regularization) term, and that the balance parameter allows us to choose accurately the cutoff frequency for low-pass or high-pass filters and the  $Q$ -factor for a band-pass filter. The computational complexity of the traditional implementation of a zero-phase IIR digital filter is  $O(n)$ . Our solutions (26) and (33) can be solved with complexity  $O(n)$  [15] as the smoothing matrices  $(Y_n^T Y_n + \lambda \Gamma_n^T \Gamma_n)^{-1} Y_n^T Y_n$  and  $(I + \lambda \Lambda_n^T \Lambda_n)^{-1}$  are symmetric, positive definite. However, our method enjoys of being inherently stable as it can be implemented as a combination of a forward filtering the input signal with an FIR filter and backward filtering with the same FIR filter.

## 8. Appendix

### 8.1. Proof of Lemma (3)

Taking the Laplace transform of (9), we obtain

$$F_n(s) = \sum_{i=0}^n \alpha_i s^i \quad (62)$$

The transform is defined by [27]

$$F_n(z) = \text{ZOH}\{F_n(s)\} = (1 - z^{-1}) \mathcal{Z}\{\text{Sampling}\{\mathcal{L}^{-1}\{\frac{F_n(s)}{s}\}\}\}$$

where  $\mathcal{L}^{-1}$  and  $\mathcal{Z}\{\cdot\}$  denote the Laplace inverse and  $\mathcal{Z}$ -transform, respectively.

In order to derive the discrete form of  $f_n(t)$ , we use the step invariance by substituting  $s$  with  $1 - z^{-1}$ :

$$F_n(z) = \sum_{i=0}^n \alpha_i (1 - z^{-1})^i \quad (63)$$

Expanding (63) using binomial expansion, we get

$$F_n(z) = \sum_{i=0}^n \alpha_i \sum_{j=0}^i \binom{i}{j} (-z^{-1})^{i-j}$$

which is equal to

$$F_n(z) = \sum_{i=0}^n \underbrace{\sum_{j=i}^n \alpha_j (-1)^i \binom{j}{i}}_{\beta_i} z^{-i}$$

Substituting  $(-1)^i \sum_{j=i}^n \alpha_j \binom{j}{i}$  with  $\beta_i$ , we obtain

$$F_n(z) = \sum_{i=0}^n \beta_i z^{-i} \quad (64)$$

Taking the inverse transform of (64), we find

$$f_{n,k} = \sum_{i=0}^n \beta_i \delta_{k-i} \quad (65)$$

which proves the Lemma.

## 8.2. Proof of Lemma (4)

In order to derive the discrete form of  $f_n(t)$  using bilinear transform, we substitute  $s$  with  $\frac{2}{T_s} \frac{1-z^{-1}}{1+z^{-1}}$  [28]. Hence in the  $Z$ -domain, (62) is expressed as

$$F_n(z) = \sum_{i=0}^n \alpha_i \left( \frac{2}{T_s} \right)^i \left( \frac{1-z^{-1}}{1+z^{-1}} \right)^i \quad (66)$$

Expanding (66) using binomial expansion, we get

$$F_n(z) = \frac{1}{(1+z^{-1})^n} \sum_{i=0}^n \alpha_i \left( \frac{2}{T_s} \right)^i (1-z^{-1})^i (1+z^{-1})^{n-i}$$

which is equal to

$$(1+z^{-1})^n F_n(z) = \sum_{i=0}^n \alpha_i \left( \frac{2}{T_s} \right)^i (1-z^{-1})^i (1+z^{-1})^{n-i}$$

It can be written as

$$\sum_{i=0}^n \binom{n}{i} F_n(z) z^{-i} = \sum_{i=0}^n \gamma_i z^{-i} \quad (67)$$

where the parameters  $\gamma_i$  are obtained from

$$\sum_{i=0}^n \gamma_i z^{-i} = \sum_{i=0}^n \alpha_i \left( \frac{2}{T_s} \right)^i \binom{n}{i} (1 - z^{-1})^i (1 + z^{-1})^{n-i}$$

Taking the inverse transform of (67), we find

$$\sum_{i=0}^n \binom{n}{i} f_{n,k-i} = \sum_{i=0}^n \gamma_i \delta_{k-i}$$

## References

- [1] Afonso, M.V., Sanches, J.M.R., 2015. Blind inpainting using  $\ell_0$  and total variation regularization. *IEEE Transactions on Image Processing* 24, 2239–2253. doi:10.1109/TIP.2015.2417505.
- [2] Goldberger et al., A.L., 2003. Physiobank, Physiokit, and Physionet: Components of a new research resource for complex physiologic signals. *Circulation* 101, e215–e220.
- [3] Bao, C., Ji, H., Quan, Y., Shen, Z., 2016. Dictionary learning for sparse coding: Algorithms and convergence analysis. *IEEE Trans. Pattern Anal. Mach. Intell* 38, 1356–1369.
- [4] Boyd, S., Vandenberghe, L., 2004. *Convex Optimization*. Cambridge University Press, New York, NY, USA.
- [5] Boyd, S.P., Barratt, C.H., 1991. *Linear Controller Design: Limits of Performance*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- [6] Butterworth, S., 1930. On the Theory of Filter Amplifiers. *Wireless Engineer* 7, 536–541.
- [7] Chan, T., Marquina, A., Mulet, P., 2000. High-order total variation-based image restoration. *SIAM J. Sci. Comput.* 22, 503–516.
- [8] Condat, L., 2013. A Direct Algorithm for 1-D Total Variation Denoising. *IEEE Signal Process. Lett.* 20, 1054–1057.
- [9] Condat, L., 2017. Discrete total variation: New definition and minimization. *SIAM Journal on Imaging Sciences* 10, 1258–1290.
- [10] Cornuéjols, G., Tutuncu, R., 2007. *Optimization Methods in Finance*. doi:10.1017/CBO9780511753886.
- [11] Dahleh, M.A., Diaz-Bobillo, I.J., 1995. *Control of Uncertain Systems: A Linear Programming Approach*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- [12] Dyer, S.A., Dyer, J.S., 2000. The bilinear transformation. *IEEE Instrumentation Measurement Magazine* 3, 30–34.
- [13] Fasano, A., Villani, V., 2014. Baselinewander removal for bioelectrical signals by quadratic variation reduction. *Signal Processing* 99, 48–57.
- [14] Goldstein, T., Osher, S., 2009. The Split Bregman Method for L1-Regularized Problems. *SIAM Journal on Imaging Sciences* 2, 323–343.
- [15] Golub, G.H., Van Loan, C.F., 1996. *Matrix Computations* (3rd Ed.). Johns Hopkins University Press, Baltimore, MD, USA.
- [16] Gustafsson, F., 1996. Determining the initial states in forward-backward filtering. *IEEE Trans. Signal Process* 44, 988–992.
- [17] Hait, E., Gilboa, G., 2019. Spectral total-variation local scale signatures for image manipulation and fusion. *IEEE Transactions on Image Processing* 28, 880–895. doi:10.1109/TIP.2018.2872630.
- [18] Hansen, P.C., 1992. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Review* 34, 561–580.

- [19] Hansen, P.C., O’Leary, D.P., 1993. The Use of the L-Curve in the Regularization of Discrete Ill-Posed Problems. *SIAM Journal on Scientific Computing* 14, 1487–1503.
- [20] Hu, Y., Jacob, M., 2012. Higher degree total variation (hdtv) regularization for image recovery. *IEEE Trans. Image Process* 21, 2559–2571.
- [21] Hu, Y., Ongie, G., Ramani, S., Jacob, M., 2014. Generalized higher degree total variation (hdtv) regularization. *IEEE Trans. Image Process* 23, 2423–2435.
- [22] Kheirati Roonizi, A., 2019. A New Approach to ARMAX Signals Smoothing: Application to Variable-Q ARMA Filter Design. *IEEE Transactions on Signal Processing* 67, 4535–4544.
- [23] Kheirati Roonizi, A., Jutten, C., 2020. Improved smoothness priors using bilinear transform. *Signal Processing* 169, 107381. doi:<https://doi.org/10.1016/j.sigpro.2019.107381>.
- [24] Kitagawa, G., Gersch, W., 1984. A smoothness priors-state space modeling of time series with trend and seasonality. *Journal of the American Statistical Association* 79, 378–389.
- [25] Mallat, S., 2008. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. 3rd ed., Academic Press, Inc., Orlando, FL, USA.
- [26] Middleton, R.H., Goodwin, G.C., 1990. *Digital Control and Estimation: A Unified Approach*. Prentice Hall Professional Technical Reference.
- [27] Ogata, K., 1987. *Discrete-time Control Systems*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- [28] Oppenheim, A., Schaffer, R., 2011. *Discrete-Time Signal Processing*. Pearson Education.
- [29] Paarmann, L.D., 2001. *Design and Analysis of Analog Filters; A Signal Processing Perspective*. Springer US.
- [30] Perrone, D., Favaro, P., 2016. A clearer picture of total variation blind deconvolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38, 1041–1055. doi:[10.1109/TPAMI.2015.2477819](https://doi.org/10.1109/TPAMI.2015.2477819).
- [31] Prado, R., West, M., 2010. *Time Series: Modeling, Computation, and Inference*. 1st ed., Chapman & Hall/CRC.
- [32] Ramsay, J., Silverman, B.W., 2005. *Functional Data Analysis*. 2nd ed., New York: Springer-Verlag.
- [33] Rudin, L.I., Osher, S., Fatemi, E., 1992. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena* 60, 259 – 268.
- [34] Sameni, R., 2017. Online filtering using piecewise smoothness priors: Application to normal and abnormal electrocardiogram denoising. *Signal Processing* 133, 52 – 63.
- [35] Schoukens, J., Pintelon, R., Hamme, H.V., 1994. Identification of linear dynamic systems using piecewise constant excitations: Use, misuse and alternatives. *Automatica* 30, 1153 – 1169.
- [36] Senjanović, I., 1984. Harmonic acceleration method for dynamic structural analysis. *Computers and Structures* 18, 71 – 80.
- [37] Shiller, R.J., 1973. A distributed lag estimator derived from smoothness priors. *Econometrica* 41, 775–788.
- [38] Smith, J., 2006. *Introduction to Digital Filters: with Audio Applications*. volume 2. Booksurge Llc.
- [39] Smith, S.W., 1997. *The Scientist and Engineer’s Guide to Digital Signal Processing*. California Technical Publishing.
- [40] Su, K., 2002. *Analog Filters*. Springer-Verlag New York Inc.
- [41] Theede, L., 2004. *Practical analog and digital filter design*. Artech House, Norwood.
- [42] Unser, M., Aldroubi, A., Eden, M., 1991. Recursive regularization filters: design, properties, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13, 272–277.
- [43] Unser, M., Blu, T., 2005. Generalized smoothing splines and the optimal discretization of the wiener filter. *IEEE Transactions on Signal Processing* 53, 2146–2159.
- [44] Wang, Z., Liu, H., Zhang, T., 2014. Optimal computational and statistical rates of convergence for sparse nonconvex learning problems. *Annals of statistics* 42 6, 2164–2201.

- [45] Whittaker, E.T., 1923. On a new method of graduation. Proc. Edinburgh Math. Soc 41, 63–75.
- [46] Widmann, A., Schröger, E., Maess, B., 2015. Digital filter design for electrophysiological data – a practical approach. Journal of Neuroscience Methods 250, 34 – 46.
- [47] Wiener, N., 1949. Extrapolation, interpolation, and smoothing of stationary time series : with engineering applications. Technology Press of the Massachusetts Institute of Technology, London.
- [48] Wright, J., Ma, Y., Mairal, J., Sapiro, G., Huang, T.S., Yan, S., 2010. Sparse representation for computer vision and pattern recognition. Proceedings of the IEEE 98, 1031–1044.
- [49] Yukawa, M., Amari, S., 2016.  $\ell_p$ -regularized least squares ( $0 < p < 1$ ) and critical path. IEEE Transactions on Information Theory 62, 488–502.
- [50] Zhang, Z., Xu, Y., Yang, J., Li, X., Zhang, D., 2015. A survey of sparse representation: Algorithms and applications. IEEE Access 3, 490–530.