Denoising Using Low-Pass Filtering Combined With Total Variation Filtering

Ayman Yousef Manasra

Submitted to the
Institute of Graduate Studies and Research
in partial fulfilment of the requirements for the degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
February 2016
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Cem Tanova
Acting Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Prof. Dr. Hasan Demirel
Chair, Department of Electrical
and Electronic Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Prof. Dr. Osman Kükrer
Supervisor

Examining Committee

1. Prof. Dr. Osman Kükrer

2. Prof. Dr. Hüseyin Özkaramanlı

3. Asst. Prof. Dr. Hasan Abou Rajab

ABSTRACT

Generally LTI filters are appropriate to denoise a signal that have low-frequency band. On the other hand, total variation denoising is appropriate to filter a signal having sparse representation. Some signals cannot be classified as having specific frequency band, or having sparse representation, such as the signal comprised in biomedical applications (near infrared spectroscopic imaging and nano-particle biosensing). This thesis introduces a new approach for denoising signals based on low-pass filtering combined with total variation denoising, assuming that the noisy observation is near infrared spectroscopic time series measurement, which can be modelled as a sum of two components, one of them low frequency and the other sparse or sparse derivative. The problem is formulated in terms of an optimization problem, and the cost function of the optimization problem is convex. As a consequence, two iterative algorithms are presented; the first one is derived using the majorization-minimization technique, and models the signals as consisted of low frequency and sparse derivative components. On the other hand, the second algorithm is derived using alternative direction method of multipliers, and models the signals as consisted of low frequency, sparse and sparse derivative components. In view of the above, simulation algorithms based on existing noisy observations are developed for validation and verification of the proposed approach. The simulation results show that the proposed approach for denoising signals recovers the signals well. Furthermore, it was found that the proposed approach is better in terms of run time.

Keywords: NIRS, low-pass filter, total variation denoising, sparse derivative.

ÖZ

LTI filtereler genellikle düşük frekans bandında olan sinyallerin gürültüden temizlenmesi için uygundur. Diğer yandan, toplam değişim gürültü giderme, seyrek temsiliyeti olan sinyalleri filtreleme için uygundur. Bazı sinyaller, yalnız belirli bir frekans bandına sahip, veya yalnız seyrek temsiliyeti olan sinyal diye tanımlanamaz. Buna örnek, biomedikal uygulamalarda karşılaşılan (yakın kızılötesi spektroskopik görüntüleme ve nano-parçacık bio-algılama) sinyallerdir. Bu tez, düşük-frekans geçirgen filtereleme ve toplam değişim gürültü gidermenin, sinyallerin gürültü giderilmesi için birarada kullanıldığı bir çalışmaya dayanmaktadır. Gürültülü verinin yakın kızılötesi spektroskopik zaman-dizisi ölçümlerinden elde edildiği varsayılmaktadır. Bu veri, biri düşük frekans içerikli, diğeri ise seyrek veya seyrek türevi olan iki sinyalin toplamı olarak modellenebilir. Problem, bir eniyileştirme problemi olarak düzenlenip, maliyet işlevi konvekstir. Problemin çözümü için iki tane algoritma incelenmiştir. Birincisi büyükleme-en aza indirgeme yöntemine dayanıp sinyali, düşük frekans içerikli ve seyrek türevli bileşenlerden oluşan sinyal olarak modellemektedir. İkinci algoritma ise sinyali düşük frekans içerikli ve seyrek ek olarak seyrek bir bileşenden oluşan sinyal olarak modeller, ve.Bu algoritmalar, önerilen yaklaşımı doğrulamak üzere, gerçek gözlemlerle elde edilmiş gürültülü sinyal uygulanmıştır. Benzetim sonuçları algoritmaların, sinyallerin gürültüden temizlenmesinde başarılı olduklarını göstermiştir. Buna ek olarak, önerilen yöntemin hesaplama zamanı olarak diğer yöntemlere göre daha iyi olduğu bulunmuştur.

Anahtar Kelimeler: NIRS, alçak geçiş filtresi, toplam varyasyon filtreler, seyrek türevi.

To My Family

ACKNOWLEDGEMENT

First and foremost, I would like to sincerely thank my supervisor, Prof. Dr. Osman Kükrer, for his guidance, patience and motivation. His guidance helped me in all the time of research and writing of thesis. I would like to thank Prof. Dr. Hüseyin Özkaramanlı and Asst. Prof. Dr. Hasan Abou Rajab, who also served in the defense committee, for their enlightening advice and comments on my thesis. I would also like to acknowledge Prof. Dr. Hasan Demirel for being cooperative and helpful as chairman of Electrical and Electronic Engineering Department.

I have to thank my parents for their love, endless prayers and support throughout my life. Thank you for giving me strength to reach the stars and chase my Dreams. My sisters and brothers, my wholehearted, thanks as well.

To all my friends, thank you for your understanding and encouragement in my many moments of crisis. Our friendship makes my life a wonderful experience. I cannot list all the names here, but you are always on my mind. Thank you, Allah, for always being there for me. This thesis is only a beginning of my journey. Last but not least, deepest thanks go to all people who took part in making this thesis real.

TABLE OF CONTENTS

ABSTRACT iii

ÖZ iv

DEDICATIONS v

ACKNOWLEDGEMENT vi

LIST OF TABLES ix

LIST OF FIGURES x

LIST OF ABBREVIATIONS xii

1 INTRODUCTION 1

1.1 Linear Time-Invariant Filters and Sparse Representation 1

1.2 Research Aims and Objectives 2

1.3 Structure of the Thesis 2

2 LITERATURE REVIEW 4

3 PRELIMINARIES 7

3.1 Near Infrared Spectroscopy 7

3.2 Notation 7

3.3 Total Variation Denoising 8

3.4 Majorization-Minimization Procedure 9

4 PROBLEM FOURMLATION 11

5 ALGORITHMS AND DESIGN OF LTI FILTERS 14

5.1 LPF/TVD Algorithm 14

5.1.1 Optimality Condition 18

5.1.2 Setting the Regularization Parameter 18

5.2 LPF/CSD Algorithm 19

5.3 Design LTI Filters 22
5.3.1 Zero-Phase Filters 23
5.3.2 Zero-Phase Higher-Order High-Pass Filter 28
5.3.3 Low-Pass Filter 32
6 SIMULATION AND RESULTS 35
6.1 Example 1 of LPF/TVD Problem 35
6.2 Example 2 of LPF/TVD Problem 39
6.3 Example 3 of LPF/CSD Problem 41
6.4 Example 4 of LPF/TVD Problem 43
6.5 Example 5 of LPF/CSD Problem 44
6.6 Comparison between LPF/TVD, LPF/CSD, and K-VSD Algorithms 44
7 CONCLUSION AND FUTURE WORK 45
REFERENCES 46

LIST OF TABLES

Table 1: LPF/TVD algorithm. 17

Table 2: LPF/CSD algorithm. 21

Table 3: Comparison between algorithms 44

LIST OF FIGURES

Figure 1: Second order high-pass filter (a) HPF frequency response (b) HPF impulse response (c) HPF pole-zero diagram. 28

Figure 2: Fourth-order non-causal high-pass filter (52) with d=2 (a) frequency response of the filter (b) impulse response (c) pole-zero diagram. 31

Figure 3: Fourth-order non-causal low-pass filter (52) with d=2 (a) frequency response of the filter (b) impulse response (c) pole-zero diagram. 34

Figure 4: Results of example 1 (a) Noisy data (b) sparse-derivative signal x and low-pass signal f (c) sum of two component (d) cost function history (e) the scatter is satisfied. Algorithm parameter d=2, $\omega_c = 0.044\pi$, $\lambda = 0.8$ 37

Figure 5: Results of example 1 (a) sparse-derivative signal x and low-pass signal f (b) sum of two component. Algorithm parameter d=2, $\omega_c = 0.044\pi$, $\lambda = 0.4$ 38

Figure 6: Results of example 2 (a) NIRS time series (b) TVD component (c) Data with TVD component subtracted (d) cost function history (e) optimality condition. Algorithm parameter d=1, $\omega_c = 0.04\pi$, $\lambda = 1.2$ 40

Figure 7: Results of example 3 (a) NIRS time series (b) TVD component for LPF/CSD. 41

Figure 8: Result of comparison between LPF/TVD and LPF/CSD algorithms (a) TVD component for LPF/TVD (b) TVD component for LPF/CSD (c) convergence behaver for algorithms. 43

Figure 9: Denoising using LPF/TVD algorithm. Algorithm parameter d=1, $\lambda = 1.5$ 43

Figure 10: Denoising using LPF/CSD algorithm. Algorithm parameter d=1, $\lambda = 1.8$ 44

LIST OF ABBREVIATIONS

LTI	Linear Time-Invariant
NIRS	Near Infrared Spectroscopy
LPF	Low-Pass Filtering
FLSA	Fused Lasso Signal Approximator
MM	Majorization-Minimization
ADMM	Alternative Direction Method of Multipliers
TVD	Total Variation Denoising
CSD	Compound Sparse Denoising

Chapter 1

INTRODUCTION

This chapter presents a brief overview of linear time-invariant (LTI) filters and sparse representation, the objective and the organization of the thesis.

1.1 Linear Time-Invariant Filters and sparse representation

Linear time-invariant (LTI) filters are a particularly important class of filters, which are vastly applied in signal processing and telecommunication systems in many applications, where they can be used in biomedical signal processing, radar, noise reduction, video processing and audio processing. The output of LTI filters can be written as linear combination of the input signal. Also the coefficients of LTI filters do not change with time. Most important, the principle of superposition is used to define the linearity property, which states that the filter response is closed under additivity property and scaling property. Also the time invariance property leads to the coefficients and frequency response of the filters being fixed. The LTI filters can be described by difference equations with unique impulse responses. Also the frequency response of LTI filters can be obtained from the impulse response. Consequently filter response specification can be described as low-pass filter, which passes low frequencies, high-pass filter which passes high frequencies, band-pass filter and band-stop filter. The Nyquist sampling theorem is used in order to avoid overlapping, and when the signal is perfectly bandlimited. Otherwise there will be aliasing [1].

Lately, the concept of sparsity and sparse representation is an active research area in signal denoising and reconstruction. It enhances the possibility of describing a signal

of interest with a restricted number of non-zero parameters or components, and numerous algorithms have been derived in order to solve convex optimization problems to estimate sparse signals [2]. The ℓ_1-norm is special in sparsity, because it has a convex proxy for it, which promotes sparsity in many sparse signal estimation and reconstruction problems [3]. Total variation denoising is unlike conventional low-pass filtering, which leads to an optimization problem, and the cost function of the optimization problem is convex and has unique minimizer.

Finally, signals may arise in many applications that cannot be categorized as having known frequency band or known transform. Consequently, linear time-invariant filters are used to estimate and filter signals when the frequency band is known. On the other hand, sparsity-based denoising is used when the signal of interest has a known transform. Interestingly, it is convenient to jointly use LTI filters and total variation denoising to filter a wider class of signals arising in many applications [4].

1.2 Research Aims and Objectives

The aim of this research study is to analyze the denoising problem introduced in [5]. The thesis discusses the methods of low-pass filtering combined with total variation denoising. In achieving this aim, the major objectives of the research can be stated as: investigating the theoretical and mathematical formulation, implementing the algorithms, verifying the optimality condition of setting the regularization parameter and observing the effect on the convergence behavior of the algorithms, making comparison between two algorithms and check the uses of each one.

1.3 Structure of the Thesis

The remaining chapters of the thesis are organized as follows. Chapter 2 illustrates the related work about linear time-invariant filters and TV denoising. Chapter 3 reviews

notation and background about TV denoising, FLSA, and the MM procedure. The problem formulations are presented in chapter 4. Chapter 5 presents the solution for the optimization problem, implementation of the algorithms, and design of the discrete-time filters. Chapter 6 illustrates the experiment results with synthetic and real data. Finally, chapter 7 concludes the thesis and future work.

Chapter 2

LITERATURE REVIEW

In this chapter, related research about linear time-invariant filters and sparsity-based denoising are presented.

The authors in [6] worked on least-squares polynomial approximation combined with total variation denoising, in order to estimate a time-series signal and an approximately piecewise constant signal individually. Least-squares polynomial approximation was used for smoothing signals that are well approximated by a polynomial, while total variation denoising was used to estimate piecewise constant signals. Consequently, they presented an algorithm based on alternative direction method of multipliers, which need the user to specify three parameters: the degree of the polynomial, the size of the window, and the regularization parameter. The new approach [5] described in this thesis replaced the least-squares polynomial approximation with linear time-invariant filters. Interestingly, the advantages of the new approach described in this thesis can be noticed as we do not have to specify a lot of parameters. In addition, the new approach can be applied to more general signals that have low frequency component, and sparse or sparse-derivative component. On the other hand, the proposed algorithms described in this thesis converge faster than the algorithms described in the cited references. Furthermore, the propped methods described in this thesis provide the optimality condition in order to set the regularization parameter.

The research in [7] is related to this work, where they proposed an approach for recovering a signal assumed to be a piecewise smooth signal. They combined Tikhonov ℓ_2 regularization and total variation denoising. The authors addressed the problem of reconstruction of a signal that arises in many applications, and which can be modeled as comprising two components, one of which is a piecewise constant signal, and the other a smooth signal. They proposed an algorithms based on Tikhonov method, which provide an optimal solution with desirable properties. As a consequence, they used total variation denoising for the piecewise constant component, and ℓ_2-norm for smooth component. In addition, the proposed methods are applicable for 1D signals and images. A notable difference is using linear time-invariant filters instead of Tikhonov regularization, which gives a suitable way to specify the frequency response. Also the new approach described in [5] used compound regularization in order to promote specific properties of the signals. Furthermore, the new approach presented fast algorithms for computational efficiency.

Many papers in image processing model the image as having two components. In [8], [9] the authors conducted various experiments in image decomposition problem, where they modeled the image as composed of structural and textural parts. They applied sparse representation approaches. However, the proposed approach described in this work utilizes sparsity for only one component.

Many problems of image denoising have been addressed in [10], [11] which are based on wavelet methods. Typically, analyzed signal has various scales by using scaled version wavelets. Clearly, there are several distinct differences between the proposed approach and other wavelet based approaches. First, the low-pass filter in that approach

is interlay decoupled from the sparse signal description. In contrast, the determination of the low-pass sub-band in wavelet domain denoising is based on the specific wavelet transform used. According to the proposed approach the properties of the low-pass component in the signal model determines how the low-pass filter design is to be. In addition, the complications that couple the appointing of a proper wavelet could be avoided through the proposed method. Furthermore, another thing that could be avoided by the proposed approach is the 'pseudo-Gibbs' phenomenon, since this approach is based on total variation denoising.

The problem of estimating a noisy data that can be modeled as comprising two components and separated by discontinuities, has attracted many researchers [12], [13]. Typically, in [14] the authors address the problem of recovered an image from a blurred and noisy image. Due to this model, and in order to prohibit edges from blurring, they avoid filtering with discontinuities. In contrast with the new approach, they model the low-pass component with two sided discontinuities.

Chapter 3

PRELIMINARIES

This chapter presents the NIRS data, notation, total variation denoising, FLSA, and the majorization-minimization procedure.

3.1 Near Infrared Spectroscopy

NIRS is a technique that uses near infrared light in order to measure the changes in the blood flow to the brain [15]. It is assumed here that the noisy data can be modeled as

$$y(n) = f(n) + x(n) + w(n) \tag{1}$$

where f is a low-pass signal, x is a sparse and/or sparse derivative signal, and w is instrumental noise (approximately stationary white Gaussian noise). To filter the signals which are modeled in (1), an optimization approach is presented that combines LTI and sparsity-based denoising to extract a low-pass and a sparse signal from a single noisy additive mixture. Here, first the sparse or sparse-derivative component x is recovered by using two algorithms. If x were a sparse-derivative signal it is extracted using LPF/TVD algorithm which uses the majorization-minimization (MM) principle. If x were a sparse or sparse-derivative signal it is derived using the LPF/CSD algorithm which uses the alternate direction method of multipliers (ADMM).

3.2 Notation

Vectors are represented by lower case bold (e.g., **x**), while matrices are represented by upper case bold (e.g., **H**). The N-point signal x is represented by the vector

$$x = \left[x(0),\ \ldots,\ x(N-1)\right]^T$$

The first order difference matrix D is denoted as

$$D := \begin{bmatrix} -1 & 1 & & \\ & -1 & 1 & \\ & & \ddots & \ddots \\ & & & -1 \quad 1 \end{bmatrix} \tag{2}$$

where $\mathbf{D}\,\mathbf{x}$ is the first-order difference of an N-point signal, and D is of size $(N-1)\times N$.

The soft-threshold function is defined as

$$\text{soft}\,(x,\mathrm{T}) := \begin{cases} x - T\left(x/|x|\right) & |x| > T \\ 0 & |x| \le T \end{cases}$$

this is the usual soft-threshold function on the real line, generalized here to the complex plane, the notation $\text{soft}\,(x,\mathrm{T})$ refers to the soft-threshold function operated element-wise to x for T > 0.

3.3 Total Variation Denoising

TVD is an approach to recover a signal **x** from a noisy data **y**, where **x** is a sparse or sparse derivative signal vector. Consequently, it is more efficient to formulate the problem in terms of ℓ_1 norm in order to promote sparsity. This leads to the constrained optimization problem

$$\begin{aligned} &\arg\min_{\mathbf{x}} \quad \|\mathbf{D}\mathbf{x}\|_1 \\ &\text{such that} \quad \|\mathbf{y}-\mathbf{x}\|_2^2 \le N\sigma^2. \end{aligned} \tag{3}$$

problem (3) is equivalent, for suitable λ, to the unconstrained optimization problem

$$\arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|_2^2 + \lambda \|\mathbf{D}\mathbf{x}\|_1 \right\} \tag{4}$$

the solution to the unconstrained optimization problem (4) is denoted as

$$\text{tvd}\,(\mathbf{y},\lambda) := \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|_2^2 + \lambda \|\mathbf{D}\mathbf{x}\|_1 \right\} \tag{5}$$

there is no explicit solution to (4), but a fast algorithm to compute the exact solution has been developed. Increasing the parameter λ has the effect of making the solution x more nearly piecewise constant. Instead of the first order difference, other approximation of derivative can be used for sparse derivative denoising.

3.4 Fused Lasso Signal Approximator

FLSA is used when the signal x and the derivative of x are sparse. Then the denoising problem is more appropriately formulated as

$$\arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|_2^2 + \lambda_0 \|\mathbf{x}\|_1 + \lambda_1 \|\mathbf{D}\mathbf{x}\|_1 \right\} \tag{6}$$

specifically, in equation (6) there are two regularizers in order to promote sparsity in the coefficients and in their difference. The solution to (6) is given by

$$\mathbf{x} = \text{soft}\,(\text{tvd}\,(\mathbf{y},\lambda_1),\lambda_0) \tag{7}$$

hence, it is not necessary to have a separate algorithm for (6). It suffices to have an algorithm for TVD problem (5).

3.5 Majorization-Minimization Procedure

The MM approach is used to solve optimization minimization problems that are difficult to minimize. It involves obtaining a sequence $\mathbf{x}_{k+1}$ by minimizing $G_k(\mathbf{x})$ such that

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}} G_k(\mathbf{x}) \tag{8}$$

The majorizer $G_k(\mathbf{x})$ is convex, and it should satisfy $G_k(\mathbf{x}) \geq F(\mathrm{x})$ and $G_k(\mathbf{x}_k) = F(\mathbf{x}_k)$. It is suitable to use a quadratic majorizer for the ℓ_1-norm because it is easy to minimize. Consequently, the majorizer can be written as

$$\begin{aligned} &\frac{1}{2}\mathbf{x}^T \Lambda_k^{-1}\mathbf{x} + \frac{1}{2}\|\mathbf{x}_k\|_1 \geq \|\mathbf{x}\|_1 \\ &\Lambda_k = diag\left(|\mathbf{x}_k|\right) \end{aligned} \tag{9}$$

with equality when $\mathbf{x} = \mathbf{x}_k$. Therefore, the left-hand-side of (9) is a majorizer of $\|\mathbf{x}\|_1$ and we will use it as $G_k(\mathbf{x})$ in the MM procedure. Equation (9) is a direct consequence of $(|x| - |x_k|)^2 \geq 0$.

Chapter 4

PROBLEM FORMULATION

The problem considered is that of filtering noisy data $y(n)$ that arises from a noisy additive mixture. Most important, this class of noisy data has significant signal processing applications. Consequently, the noisy data $y(n)$ can be modeled as a mathematical equation

$$y = f + x + w, \tag{10}$$

filtering noisy data $y(n)$ means obtaining the following estimates

$$\begin{aligned} \hat{x} &\approx x. \\ \hat{f} &= f. \end{aligned} \tag{11}$$

suppose that estimate $\hat{x}$ of x is given, then it is easy to estimate f as

$$\hat{f} := \mathrm{LPF}(y - \hat{x}) \tag{12}$$

it is assumed that LPF is a specified low-pass filter. Thus, the problem will be to find the estimate $\hat{x}$. Using (12) in (11) gives

$$\mathrm{LPF}(y - \hat{x}) \approx f. \tag{13}$$

using (10) in (13) gives

$$\mathrm{LPF}(y - \hat{x}) \approx y - x - w. \tag{14}$$

in order to get an equation which contains just noisy data $\hat{x}$, and w, using (11) in (14) gives

$$\mathrm{LPF}(y - \hat{x}) \approx y - \hat{x} - w. \tag{15}$$

rewrite (15) as

$$(y - \hat{x}) - \mathrm{LPF}(y - \hat{x}) \approx w. \tag{16}$$

it is clear in equation (16) that a high-pass filter is applied to the term $(y - \hat{x})$. Consequently, by using that assumption makes the frequency response zero phase. Hence, the proposed approach will be effective and efficient. As is well known, the high-pass filter can be formulated as $\mathrm{HPF} := 1 - \mathrm{LPF}$. Also, we use this form to update equation (16) as

$$\mathrm{HPF}(y - \hat{x}) \approx \mathrm{w}. \tag{17}$$

in the previous equation, note that it does not contain the unknown signals f and x, but it has $\hat{x}$. Therefore, it can be used to find the estimate $\hat{x}$. However, to make this equation useful we define the high-pass filter matrix by bold-face **H**; hence, updating equation (17) to be $\mathbf{H}(\mathbf{y} - \hat{\mathbf{x}}) \approx \mathbf{w}$.

Specifically, the total variation denoising will provide the exact solution to x; hence, x has a sparse or approximately sparse derivative. Furthermore, in order to formulate the total variation filter as minimizing a particular cost function, the ℓ_1 norm can be used in an effective way with first-order difference to give the sparse solution. More generally, this is the basis of total variation denoising. As a result, the output of the total variation filter will provide the solution **x** to the constrained optimization problem

$$\begin{aligned} &\arg \min_{\mathbf{x}} \quad \|\mathbf{Dx}\|_1 \\ &\text{such that} \quad \|\mathbf{H}(\mathbf{y} - \mathbf{x})\|_2^2 \le N\sigma^2. \end{aligned} \tag{18}$$

however, for a suitable regularization parameter λ, the unconstrained optimization problem is equivalent to the constrained optimization problem in (18)

$$\arg \min_{\mathbf{x}} \left\{ \frac{1}{2} \|\mathbf{H}(\mathbf{y} - \mathbf{x})\|_2^2 + \lambda \|\mathbf{Dx}\|_1 \right\}. \tag{19}$$

Specifically, the unconstrained optimization problem is easier to solve than the constrained optimization problem. Recently, a fast algorithm has been developed to solve this type of optimization problem. In chapter 5 section 1, we derive an algorithm using the majorization-minimization approach in order to solve the unconstrained optimization problem (19). Also we describe an approach to choose a suitable regularization parameter λ.

However, the high-pass filter matrix **H** will be defined as

$$\mathbf{H} = \mathbf{A}^{-1}\,\mathbf{B} \tag{20}$$

where matrix **A** and matrix **B** are banded; hence, the fact that these matrices are banded leads to computational efficiency of the algorithm. On the other hand, the high-pass filter matrix **H** is not banded because the inverse of **A** is not banded. In chapter 5 section 2, we present the design of the high-pass filter matrix **H**. In addition, in order to estimate f in (12) the low-pass filter matrix $\mathbf{L} = \mathbf{I} - \mathbf{A}^{-1}\,\mathbf{B}$ is used.

Chapter 5

ALGORITHMS AND DESIGN OF LTI FILTERS

This chapter firstly implements the LPF/TVD and LPF/CSD algorithms and describes the design of different types of LTI filters.

5.1 LPF/TVD Algorithm

The optimization problem (19) is used to find the solution to total variation denoising. However, the cost function of the optimization problem is convex and non-differentiable. Many algorithms are developed for this type of problem (19) such as in [16]. Consequently, in this section an algorithm is derived to solve (19) by applying the majorization-minimization approach [17].

Typically, in order to make the solution to (19) unique, a constant should be added. Also the variables should be changed to facilitate use of majorization-minimization as follows

$$\mathbf{x} = \mathbf{S}\mathbf{u} \tag{21}$$

where **S** is a matrix having a size of $N \times (N-1)$. Note that the form of **S** is

$$\mathbf{S} \coloneqq \begin{bmatrix} 0 & & & & \\ 1 & 0 & & & \\ 1 & 1 & 0 & & \\ \vdots & & \ddots & \ddots & \\ 1 & 1 & \cdots & 1 & 0 \\ 1 & 1 & \cdots & 1 & 1 \end{bmatrix} \tag{22}$$

Also, note that **S** represents a cumulative sum. Hence, if the size of $\mathbf{D}$ is $(N\text{-}1)\times N$ and the size of **S** is $N\times(N-1)$, then

$$\mathbf{DS = I} \tag{23}$$

where **I** represents an identity matrix of size $(N\text{-}1)\times(N\text{-}1)$. Therefore, matrix **S** can be defined as the discrete anti-derivative. Note that,

$$\mathbf{Dx = DSu = u} \tag{24}$$

the matrix **B** which is given in chapter 3 can be formulated as

$$\mathbf{B = B_1 D} \tag{25}$$

where $\mathbf{B}_1$ is a banded matrix to make the algorithm effective and computationally efficient.

now, in order to minimize the cost function with respect to **u** instead of **x**, by using (21) in (19) gives

$$\arg\min_{\mathbf{u}} \left\{ F(\mathbf{u}) = \frac{1}{2}\left\|\mathbf{H(y - Su)}\right\|_2^2 + \lambda\left\|\mathbf{u}\right\|_1 \right\}. \tag{26}$$

in this case, the output of the optimization problem (26) is the optimal solution **u**; then the solution to equation (19) is unique and can be obtained by (21), by applying the majorization-minimization in order to minimize (26). Consequently, the cost function $F(\mathbf{u})$ in (26) needs a majorizer $G_k(\mathbf{u})$ to solve problem (26) directly. However, the idea is that we can use equation (9) to find a majorizer to $F(\mathbf{u})$ as

$$G_k(\mathbf{u}) = \frac{1}{2}\left\|\mathbf{H(y - Su)}\right\|_2^2 + \frac{\lambda}{2}\mathbf{u}^T\mathbf{\Lambda}_k^{-1}\mathbf{u} + \frac{\lambda}{2}\left\|\mathbf{u}_k\right\|_1$$

where $\mathbf{\Lambda}_k$ is a diagonal matrix and can be defined as the following

$$\mathbf{\Lambda}_k = \begin{bmatrix} |\mathbf{u}_k(1)| & & & \\ & |\mathbf{u}_k(2)| & & \\ & & \ddots & \\ & & & |\mathbf{u}_k(N)| \end{bmatrix} = diag(|\mathbf{u}_k(N)|).$$

note that, by using (20), (23), and (25), $\mathbf{HS}$ can be written as

$$\mathbf{HS} = \mathbf{A}^{-1}\mathbf{BS} = \mathbf{A}^{-1}\mathbf{B}_1\mathbf{DS} = \mathbf{A}^{-1}\mathbf{B}_1$$

hence, the majorizer $G_k(\mathbf{u})$ can be simplified as

$$G_k(\mathbf{u}) = \frac{1}{2}\left\|\mathbf{A}^{-1}\mathbf{By} - \mathbf{A}^{-1}\mathbf{B}_1\mathbf{u}\right\|_2^2 + \frac{\lambda}{2}\mathbf{u}^T\mathbf{\Lambda}_k^{-1}\mathbf{u} + C$$

where $\mathbf{C}$ is a constant which represents the term $\|\mathbf{u}_k\|_1$. Therefore, the majorization-minimization approach produces a sequence of $\mathbf{u}_{k+1}$ according to

$$\mathbf{u}_{k+1} = \arg\min_{\mathbf{u}} G_k(\mathbf{u}) \tag{27}$$

specifically, the solution to (27) can be given as

$$\mathbf{u}_{k+1} = (\mathbf{B}_1^T(\mathbf{AA}^T)^{-1}\mathbf{B}_1 + \lambda\mathbf{\Lambda}_k^{-1})^{-1}\mathbf{B}_1^T(\mathbf{AA}^T)^{-1}\mathbf{By}. \tag{28}$$

note that if the value of $\mathbf{u}_k$ goes to zero, then the entries of $\mathbf{\Lambda}_k^{-1}$ will go to infinity. In other words, there is a numerical problem. Consequently, to avoid this issue it is useful to use the matrix inverse lemma, which is described in [18]. Therefore, using this lemma to rewrite equation (28), we get

$$(\mathbf{B}_1^T(\mathbf{AA}^T)^{-1}\mathbf{B}_1 + \lambda\mathbf{\Lambda}_k^{-1})^{-1} = \frac{1}{\lambda}\mathbf{\Lambda}_k - \frac{1}{\lambda}\mathbf{\Lambda}_k\mathbf{B}_1^T\left(\underbrace{\lambda\mathbf{AA}^T + \mathbf{B}_1\mathbf{\Lambda}_k\mathbf{B}_1^T}_{\text{banded}}\right)^{-1}\mathbf{B}_1\mathbf{\Lambda}_k.$$

(29)

note that $\mathbf{A}$, $\mathbf{B}_1$, and $\mathbf{\Lambda}_k$ are banded matrices, so the result of the indicated matrix is banded, which makes the algorithm effective. Also, it is clear that no division by zero

happens in (29). Finally, using (29), the majorization-minimization approach can be derived as

$$\mathbf{b} \leftarrow \frac{1}{\lambda}\mathbf{B}_1^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{B}\mathbf{y}$$
$$\mathbf{\Lambda}_k \leftarrow \mathrm{diag}(|\mathbf{u}_k|)$$
$$\mathbf{u}_{k+1} \leftarrow \mathbf{\Lambda}_k\left[\mathbf{b} - \mathbf{B}_1^T(\lambda\mathbf{A}\mathbf{A}^T + \mathbf{B}_1\mathbf{\Lambda}_k\mathbf{B}_1^T)^{-1}\mathbf{B}_1\mathbf{\Lambda}_k\mathbf{b}\right].$$

Now, Table 1 shows algorithm 1 to solve the **LPF/TVD** problem (19), and also algorithm 1 based on the update equation (29). Interestingly, when x is obtained, then the low-pass signal f is obtained by using the low-pass filter $\mathbf{L} = \mathbf{I} - \mathbf{A}^{-1}\mathbf{B}$ to $(\mathbf{y} - \mathbf{x})$ in equation (12).

Table 1: LPF/TVD algorithm

Algorithm 1: to solve problem (19)

Input: $\mathbf{y} \in R^N$, $\lambda > 0$

Output: $\mathbf{x}, \mathbf{f} \in R^N$

1. $\mathbf{b} \leftarrow (1/\lambda)\mathbf{B}_1^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{B}\mathbf{y}$
2. $\mathbf{u} \leftarrow \mathbf{D}\,\mathbf{y}$
3. **repeat**
4. $\mathbf{\Lambda} \leftarrow \mathrm{diag}(|\mathbf{u}|)$
5. $\mathbf{Q} \leftarrow \lambda\mathbf{A}\mathbf{A}^T + \mathbf{B}_1\mathbf{\Lambda}\mathbf{B}_1^T$
6. $\mathbf{u} \leftarrow \mathbf{\Lambda}\left[\mathbf{b} - \mathbf{B}_1^T\mathbf{Q}^{-1}\mathbf{B}_1\mathbf{\Lambda}\mathbf{b}\right]$
7. **Until** convergence
8. $\mathbf{x} \leftarrow \mathbf{S}\mathbf{u}$
9. $\mathbf{f} \leftarrow (\mathbf{y} - \mathbf{x}) - \mathbf{A}^{-1}\mathbf{B}\,(\mathbf{y} - \mathbf{x})$

10. **return** $\mathbf{x}, \mathbf{f}$

Furthermore, the computational cost of each iteration is $O(dN)$, where the order of the high-pass filter $\mathbf{H}$ is d. However, algorithm 1 is programmed in MATLAB by using LAPACK [19], in order to derive an efficient and accurate implementation for solving banded linear systems.

5.1.1 Optimality Condition

The solution $\mathbf{u}^*$ is minimizer of $F(\mathbf{u})$ in (26) if and only if it satisfies certain conditions, using proposition 1.3 of [20]. Then the sub-gradient of F is given by

$$\partial F(\mathbf{u}^*) = \mathbf{H}^T\mathbf{H}(\mathbf{x}-\mathbf{y}) + \lambda \mathbf{D}^T \text{sign}\{\mathbf{D}\mathbf{x}\}$$

Thus,

$$\mathbf{H}^T\mathbf{H}(\mathbf{y}-\mathbf{x}) \in \lambda \mathbf{D}^T \text{sign}\{\mathbf{D}\mathbf{x}\}$$

multiplying both side by $\mathbf{S}^T$, and noting that $\mathbf{S}^T\mathbf{D}^T = \mathbf{I}$, and $\mathbf{H}^T\mathbf{H} = \mathbf{H}$, we get

$$\mathbf{S}^T\mathbf{H}(\mathbf{y}-\mathbf{x}) \in \lambda \text{sign}\{\mathbf{D}\mathbf{x}\}$$

define **g,** as the cumulative sum of the residual

$$\mathbf{g} = \mathbf{S}^T\mathbf{H}(\mathbf{y}-\mathbf{x}), \quad \mathbf{u} = \mathbf{D}\mathbf{x} \tag{30}$$

then **u** must satisfy

$$\begin{aligned} g(\text{n}) &= \text{sign}(\text{u(n)}).\lambda \quad \text{for u(n)} \neq 0 \\ |g(\text{n})| &\leq \lambda \quad \text{for u(n)=0} \end{aligned} \tag{31}$$

5.1.2 Setting the Regularization Parameter

Suppose the noisy data just consists of the noise only such that $\mathbf{y} = \mathbf{w}$. Then **x** =0, and from optimality condition (30), $\mathbf{g} = \mathbf{S}^T\mathbf{H}\mathbf{w}$ and **u**=0. Then (31) is used to obtain the

optimal λ as $\lambda \geq \max(\mathbf{g}) = \max(\mathbf{S}^T\mathbf{H}\mathbf{w})$. To avoid distortion of **x**, we choose the minimal value as follows

$$\lambda = \max(\mathbf{S}^T\mathbf{H}\mathbf{w}) \tag{32}$$

which assumes availability of the noise signal w. Start- and end-transients should be omitted when using (32). In practice, the noise is not known, but its statistics may be known and an approximate maximum value precomputed.

5.2 LPF/CSD Algorithm

Unlike TVD, the signal is modeled to be sparse or having a sparse derivative component. In this case, the cost function of the optimization problem contains linear combination of two regularizers (one for ℓ_1 and the other for total variation regularizers). Also the compound regularization encourages piecewise smooth solution and sparse solution. In this section, an algorithm is derived using the alternating direction method of multipliers (ADMM) to solve the unconstrained optimization problem

$$\arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\left\|\mathbf{H}(\mathbf{y}-\mathbf{x})\right\|_2^2 + \lambda_0 \left\|\mathbf{x}\right\|_1 + \lambda_1 \left\|\mathbf{D}\mathbf{x}\right\|_1 \right\} \tag{33}$$

the optimization problem (33) is defined as the LPF/CSD problem. Many algorithms are given for solving problems containing compound regularization, where the images are sparse and have sparse derivatives. However, problems of the form (33) have been addressed in [21], as the fused lasso signal approximator.

At this point, if the majorization-minimization (MM) approach is used to solve problem (33), then the computational cost for each iteration is $O(dN)$, where N is the length of the signal x. Then this leads to the solution of not banded system of equations,

so it is not sufficient to use MM approach. Consequently, an iterative algorithm based on ADMM [22] has been derived to solve (33).

The ADMM is based on decomposing the objective function in (33) by applying variable splitting as in [23]. Then, the variable splitting will split x into variable, say x and **v**, in order to serve as the argument of $\|\mathbf{Dx}\|_1$, and then minimize the constrained optimization problem subject to the condition that the two variables should be equal. At this point, problem (33) is equivalent to the constrained optimization problem

$$\begin{aligned} &\arg\min_{\mathbf{x},\mathbf{v}} \left\{ \frac{1}{2}\|\mathbf{Hy} - \mathbf{Hx})\|_2^2 + \lambda_0 \|\mathbf{v}\|_1 + \lambda_1 \|\mathbf{Dv}\|_1 \right\} \\ &\text{such that } \mathbf{v} = \mathbf{x} \end{aligned} \tag{34}$$

now, using ADMM, problem (34) can be minimized with respect to **x** and **v** by an iterative algorithm:

$$\mathbf{x} \leftarrow \arg\min_{\mathbf{x}} \ \|\mathbf{Hy} - \mathbf{Hx})\|_2^2 + \mu\|\mathbf{v} - \mathbf{x} - \mathbf{d}\|_2^2 \tag{35a}$$

$$\mathbf{v} \leftarrow \arg\min_{\mathbf{v}} \ \lambda_0 \|\mathbf{v}\|_1 + \lambda_1 \|\mathbf{Dv}\|_1 + 0.5\mu\|\mathbf{v} - \mathbf{x} - \mathbf{d}\|_2^2 \tag{35b}$$

$$\mathbf{d} \leftarrow \mathbf{d} - (\mathbf{v} - \mathbf{x}) \tag{35c}$$

$$\text{Go to (35a)} \tag{35d}$$

note that, each iteration consists of two minimization problems with respect to x and **v**. The ADMM algorithm produces a sequence of $\mathbf{x}, \mathbf{v}$, and $\mathbf{d}$. Also, the parameter μ is specified as a positive scalar, which affects the speed of convergence, but not the solution to which the algorithm converges. Also the algorithm (35) requires to initialize the variables $\mathbf{d}$ and **v** to all-zero vectors the same size as **y**. In line (35a), while the cost function is convex quadratic, then **x** has the explicit solution

$$\mathbf{x} \leftarrow (\mathbf{H}^T\mathbf{H} + \mu\mathbf{I})^{-1}(\mathbf{H}^T\mathbf{Hy} + \mu(\mathbf{v} - \mathbf{d})) \tag{36}$$

equation (20) is used to simplify $\mathbf{H}^T\mathbf{Hy}$ as

$$\mathbf{H}^T\mathbf{H}\mathbf{y} = (\mathbf{A}^{-1}\mathbf{B})^T\mathbf{A}^{-1}\mathbf{B}\mathbf{y} = \mathbf{B}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{B}\mathbf{y} \tag{37}$$

however, the matrix inverse lemma can be used in order to avoid high computational cost as follows:

$$(\mathbf{H}^T\mathbf{H} + \mu\mathbf{I})^{-1} = \frac{1}{\mu}\left[\mathbf{I} - \mathbf{B}^T(\mu\mathbf{A}\mathbf{A}^T + \mathbf{B}\mathbf{B}^T)^{-1}\mathbf{B}\right] \tag{38}$$

line (35a) can be implemented by using (37) and (38) in (36) compactly as

$$\mathbf{g} \leftarrow \frac{1}{\mu}\mathbf{B}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{B}\mathbf{y} + (\mathbf{v} - \mathbf{d}) \tag{39}$$

$$\mathbf{x} \leftarrow \mathbf{g} - \mathbf{B}^T(\mu\mathbf{A}\mathbf{A}^T + \mathbf{B}\mathbf{B}^T)^{-1}\mathbf{B}\mathbf{g} \tag{40}$$

notice that the term $\left(\frac{1}{\mu}\mathbf{B}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{B}\mathbf{y}\right)$ in (39) can be precomputed, because $\mathbf{y}$ does not change during algorithm (35). In line (35b), the solution for the minimization problem can be computed using (7) as

$$\mathbf{v} \leftarrow \text{soft}\,(\text{tvd}\,(\mathbf{x} + \mathbf{d}, \lambda_1/\mu), \lambda_0/\mu) \tag{41}$$

It has been shown that, the LPF/CSD can be implemented by ADMM (35). Note that, the algorithm (35) can be computed efficiently because all matrices are banded. The complete algorithm is listed in table 2.

Table 2: LPF/CSD algorithm

Algorithm 2: to solve problem(33)

Input: $\mathbf{y} \in R^N$, $\lambda_0 > 0$, $\lambda_1 > 0$, $\mu > 0$

Output: $\mathbf{x}, \mathbf{f} \in R^N$

1. $\mathbf{v} \leftarrow 0$
2. $\mathbf{d} \leftarrow 0$

3. $\mathbf{b} \leftarrow (1/\mu)\mathbf{B}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{B}\mathbf{y}$

4. **repeat**

5. $\mathbf{g} \leftarrow \mathbf{b} + \mathbf{v} - \mathbf{d}$

6. $\mathbf{x} \leftarrow \mathbf{g} - \mathbf{B}^T(\mu\mathbf{A}\mathbf{A}^T + \mathbf{B}\mathbf{B}^T)^{-1}\mathbf{B}\mathbf{g}$

7. $\mathbf{v} \leftarrow \text{soft}(\text{tvd}(\mathbf{x} + \mathbf{d}, \lambda_1/\mu), \lambda_0/\mu)$

8. $\mathbf{d} \leftarrow \mathbf{d} - \mathbf{v} + \mathbf{x}$

9. **Until** convergence

10. $\mathbf{f} \leftarrow (\mathbf{y} - \mathbf{x}) - \mathbf{A}^{-1}\mathbf{B}(\mathbf{y} - \mathbf{x})$

11. **return** $\mathbf{x}, \mathbf{f}$

5.3 Design of LTI Filters

In this chapter, in order to use the method described in chapters 4 and 5, section 1.1 and 1.2, discrete-time filters can be completely designed and implemented. Specifically, we are interested in designing zero-phase non-causal recursive high-pass filters. The difference equation of a discrete-time filter is

$$\sum_k a(k)y(n-k) = \sum_k b(k)x(n-k) \tag{42}$$

where $x(n)$ is the input signal and $y(n)$ is the output signal. Consequently the transfer function of the discrete-time filter can be written as $H(\mathrm{e}^{jw}) = B(e^{jw})/A(e^{jw})$. Interestingly, the proposed algorithms described in chapter 5 are aimed at filtering finite-length signals, and sparsity-based denoising problems are usually used with finite-length signals. So the designed LTI filters will use finite-length signals. Consequently, the difference equation for finite length signals is

$$\mathbf{A}\mathbf{y} = \mathbf{B}\mathbf{x} \tag{43}$$

using (43) the output of the discrete-time filter is given by

$$\mathbf{y} = \mathbf{A}^{-1}\mathbf{B}\,\mathbf{x} \tag{44}$$

note that in (44), **A** and **B** are banded matrices; **A** must be square and invertible also **B** need not to be square and invertible. Moreover, there is no need for start-transient and end-transient because the initial states of the filter are not specified. So they are defined to be zero.

5.3.1 Zero-Phase Filters

An important property of zero-phase filters is that the phase response is specified to be zero, otherwise there will be distortion in the phase, and equation (16) assumed that the filter is zero-phase. In particular, the zero-phase condition is satisfied when its impulse response is symmetric, or when the frequency response is real-valued. Another important property of zero-phase that leads to specific properties of **A** and **B**, is that the symmetry property leads to the same filter behavior in backward direction as in forwards direction. Hence, **H** is applied to a reversed version of signal x or applied directly to the signal **x**. Assume **J** is the reversal matrix, the following condition should be satisfied

$$\mathbf{J}\mathbf{H}\mathbf{J} = \mathbf{H} \tag{45}$$

where **H** is a rectangular matrix and **J** is a square matrix. Note that if matrices **A** and **B** satisfy the conditions

$$\begin{aligned} \mathbf{J}\mathbf{A}\mathbf{J} &= \mathbf{A} \\ \mathbf{J}\mathbf{B}\mathbf{J} &= \mathbf{B} \end{aligned} \tag{46}$$

then $\mathbf{H} = \mathbf{A}^{-1}\mathbf{B}$ satisfies condition (45). Moreover the filter matrices **A** and **B** used in the LPF/TVD algorithm should satisfy (46). Now, to make it clear here an example about recursive zero-phase filters will illustrate condition (46).

Example: consider a zero phase non-causal 2^{nd}-order Butterworth filter. The input and output of the LTI system are described as

$$a_1 y(n+1) + a_0 y(\mathrm{n}) + a_1 y(n-1) = -x(n+1) + 2x(\mathrm{n}) - \mathrm{x}(n-1) \qquad (47)$$

from the previous discussion, the output signal can be implemented by using $\mathbf{y} = \mathbf{A}^{-1}\mathbf{B}\,\mathbf{x}$, where matrix **B** has the form

$$\mathbf{B} = \begin{bmatrix} -1 & 2 & -1 & & & & \\ & -1 & 2 & -1 & & & \\ & & -1 & 2 & -1 & & \\ & & & -1 & 2 & -1 & \\ & & & & -1 & 2 & -1 \end{bmatrix} \qquad (48)$$

note that, the dimension of **B** is $(N-2)\times N$. In this example $\mathbf{B}_1 = -\mathbf{D}$ and of dimension $(N-2)\times(N-1)$. Moreover, **A** has the form

$$\mathbf{A} = \begin{bmatrix} a_0 & a_1 & & & \\ a_1 & a_0 & a_1 & & \\ & a_1 & a_0 & a_1 & \\ & & a_1 & a_0 & a_1 \\ & & & a_1 & a_0 \end{bmatrix} \qquad (49)$$

note that, the dimension of **A** is $(N-2)\times(N-2)$. Consequently, equation (46) is easily verified. Furthermore, the output **y** is a vector of size $(N-2)$. The corresponding transfer function $H(z)$ of the difference equation (47) can be obtained as

$$a_1 z y(z) + a_0 y(z) + a_1 z^{-1} y(z) = -z x(z) + 2x(z) - z^{-1} x(z)$$
$$y(z)\left(a_1 z + a_0 + a_1 z^{-1}\right) = x(z)\left(-z + 2 - z^{-1}\right)$$

$$H(z) = \frac{y(z)}{x(z)} = \frac{-z + 2 - z^{-1}}{a_1 z + a_0 + a_1 z^{-1}} \qquad (50)$$

Now, the filter must be low-pass and described by $L(z) = 1 - H(z)$, such that at the Nyquist frequency $\omega = \pi$, the low-pass filter will have zero gain. In order to have unity gain c of the system (47), this can be found by setting $c = H(-1)$

$$c = H(-1) = \frac{-(-1) + 2 - (-1)^{-1}}{a_1(-1) + a_0 + a_1(-1)^{-1}} = \frac{4}{a_0 - 2a_1} \tag{51}$$

consequently, in order to have unity gain for the high-pass filter (47), equation (51) should satisfy $a_0 - 2a_1 = 4$. Moreover, the corresponding frequency response of the high-pass filter can be obtained by setting $z = e^{j\omega}$ such that

$$H(\mathrm{e}^{j\omega}) = \frac{-\mathrm{e}^{j\omega} + 2 - \mathrm{e}^{-j\omega}}{a_1\,\mathrm{e}^{j\omega} + a_0 + a_1\,\mathrm{e}^{-j\omega}} = \frac{2 - (\mathrm{e}^{j\omega} + \mathrm{e}^{-j\omega})}{a_0 + a_1(\mathrm{e}^{j\omega} + \mathrm{e}^{-j\omega})}$$

using $\cos\omega = \dfrac{e^{j\omega} + e^{-j\omega}}{2}$ and $a_1 = \dfrac{a_0 - 4}{2}$ then

$$H(\mathrm{e}^{j\omega}) = \frac{2 - 2\cos\omega_c}{a_0 + (a_0 - 4)\cos\omega_c}$$

specifically, in this example the cut-off frequency can be set such that the frequency response equals 0.5, then the coefficient a_0 can be obtained as

$$\frac{1}{2} = \frac{2 - 2\cos\omega_c}{a_0 + (a_0 - 4)\cos\omega_c}$$

$$a_0 + (a_0 - 4)\cos\omega_c = 4 - 4\cos\omega_c$$

$$a_0 = \frac{4}{1 + \cos\omega_c}$$

now, in order to find the coefficients of the frequency response $H(\mathrm{e}^{j\omega})$, the cut-off frequency can be set to $\omega_c = 0.1\pi$. Then a_0 and a_1 can be found as $a_0 = \dfrac{4}{1 + \cos 0.1\pi} = 2.05$ and $a_1 = \dfrac{a_0 - 4}{2} = -0.975$.

hence, the transfer function becomes

$$H(z) = \frac{-z + 2 - z^{-1}}{-0.975z + 2.05 - 00.975z^{-1}}$$

the poles of the transfer function can be obtained from the roots of the denominator of $H(z)$

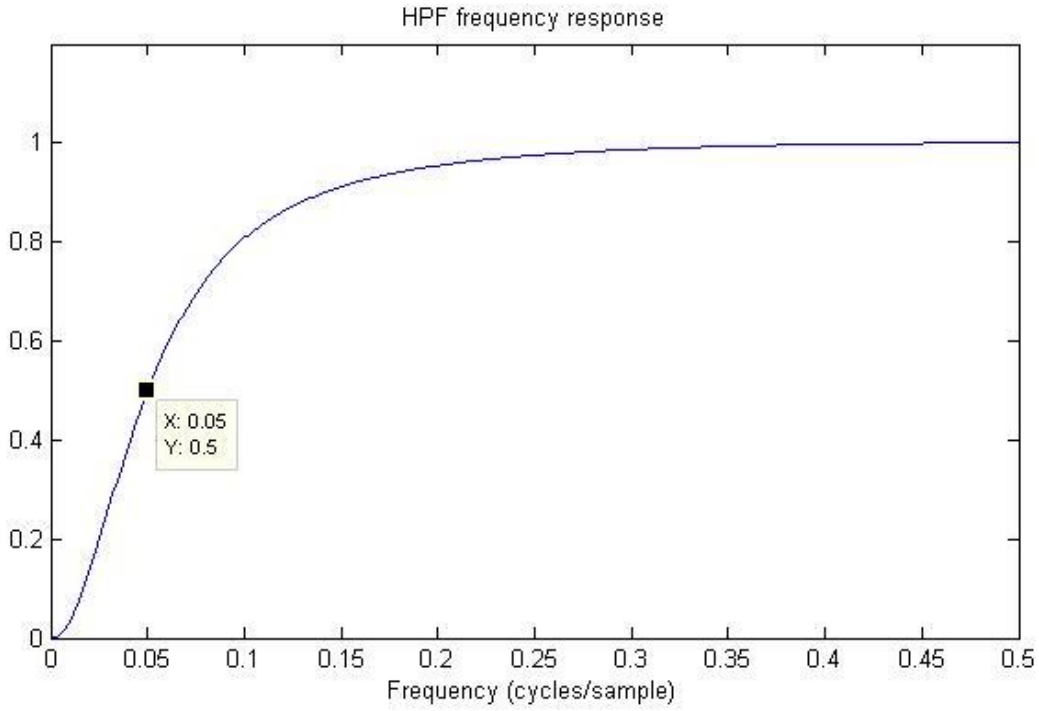
$$-0.975z + 2.05 - 00.975z^{-1} = 0$$
$$0.975z^2 - 2.05z + 0.975 = 0$$

$$z = \frac{2.05 \pm \sqrt{2.05^2 - 4 \times 0.975^2}}{2 \times 0.975}$$
$$z = 0.725 \text{ and } z = 1.38$$

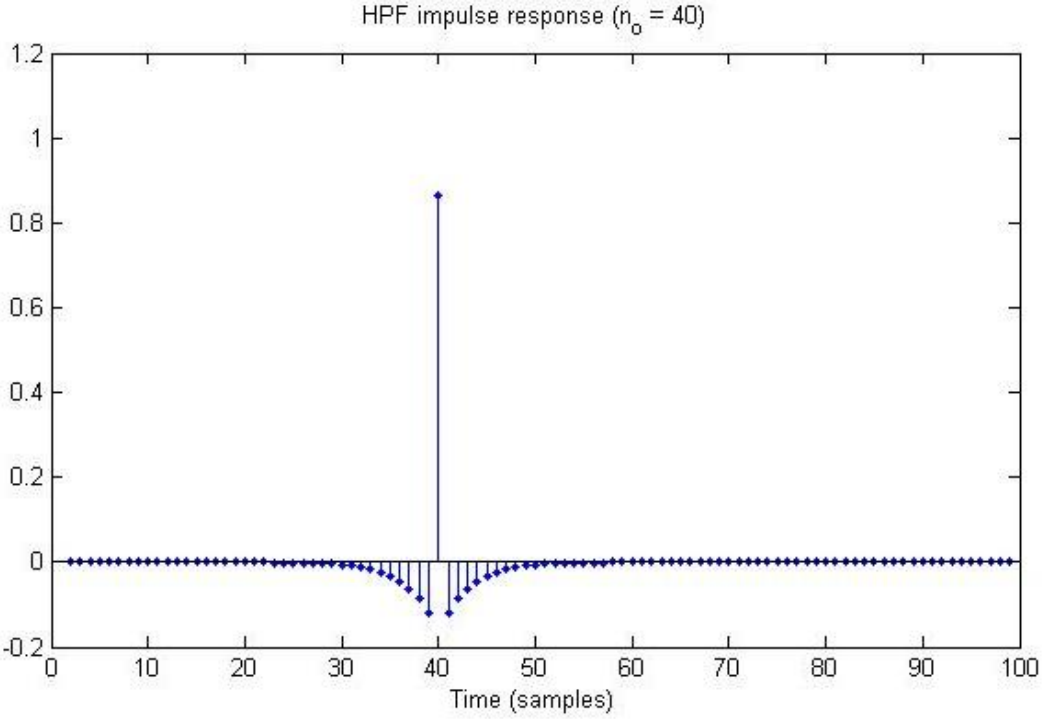
similarly, the zeros of the transfer function are the roots of the numerator which can be solved as

$$-z + 2 - z^{-1} = 0$$
$$z^2 - 2z + 1 = 0$$
$$(z-1)(z-1) = 0$$
$$z = 1$$

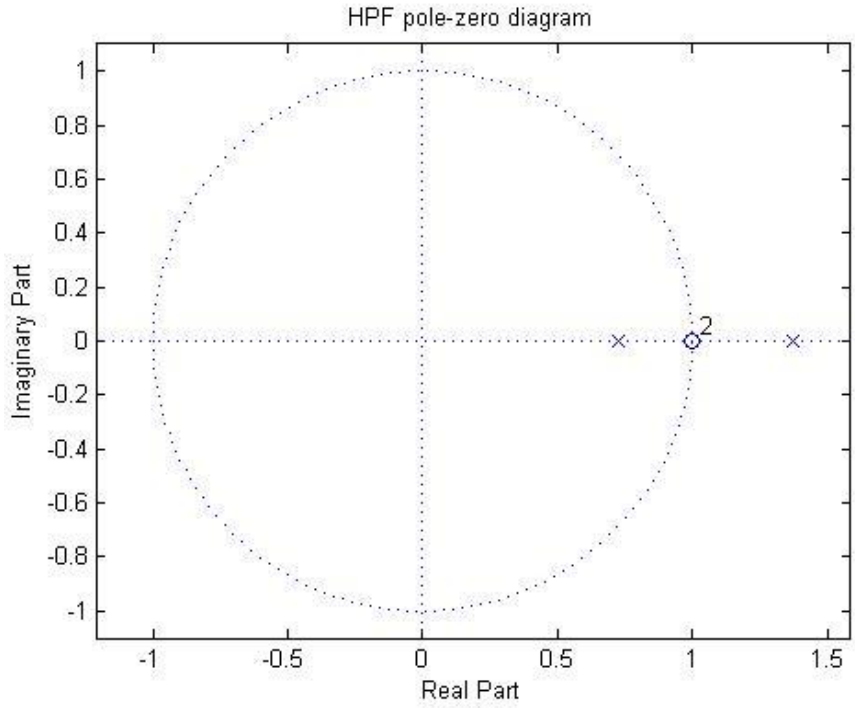
there is a second-order zero located at z=1. The filter will not pass a constant signal and a ramp signal. So when the input signal is a linear combination of the form $x(n) = k_0 + k_1 n$ then the output is zero. Consequently, the 2^{nd} order high-pass filter is illustrated in Figure 1.


HPF frequency response
X: 0.05
Y: 0.5
Frequency (cycles/sample)

(a)


HPF impulse response (n_o = 40)
Time (samples)

(b)



(c)

Figure 1: Second order high-pass filter (a) HPF frequency response (b) HPF impulse response (c) HPF pole-zero diagram.

5.3.2 Zero-Phase Higher-Order High-Pass Filter

The transfer function of a higher-order high-pass filter is given by

$$H(\mathrm{z}) = \frac{(-\mathrm{z}+2-\mathrm{z}^{-1})^d}{(-\mathrm{z}+2-\mathrm{z}^{-1})^d + \alpha(\mathrm{z}+2+\mathrm{z}^{-1})^d} \tag{52}$$

therefore, the 2d-order zero of the filter $H(\mathrm{z})$ is located at z=1. Then $H(\mathrm{e}^{j\omega})$=0 at ω =0. Interestingly, the transfer function (52) can be rewritten as

$$H(\mathrm{z}) = 1 - \frac{\alpha(\mathrm{z}+2+\mathrm{z}^{-1})^d}{(-\mathrm{z}+2-\mathrm{z}^{-1})^d + \alpha(\mathrm{z}+2+\mathrm{z}^{-1})^d}$$

in addition, the frequency response $H(\mathrm{e}^{j\omega})$ of the transfer function (52) can be obtained as follows

$$H(\mathrm{e}^{j\omega}) = \frac{(-\mathrm{e}^{j\omega}+2-\mathrm{e}^{-j\omega})^d}{(-\mathrm{e}^{j\omega}+2-\mathrm{e}^{-j\omega})^d + \alpha(\mathrm{e}^{j\omega}+2+\mathrm{e}^{-j\omega})^d}$$
$$= \frac{(2-(\mathrm{e}^{j\omega}+\mathrm{e}^{-j\omega}))^d}{(2-(\mathrm{e}^{j\omega}+\mathrm{e}^{-j\omega}))^d + \alpha(2+(\mathrm{e}^{j\omega}+\mathrm{e}^{-j\omega}))^d}$$
$$= \frac{(2-2\cos\omega)^d}{(2-2\cos\omega)^d + \alpha(2+2\cos\omega)^d}$$

note that, the transfer function (52) represents a zero-phase high-pass Butterworth filter. Now, if we want to find ω_c, so specify $H(\mathrm{e}^{j\omega})$=0.5 as follows

$$\frac{1}{2} = \frac{(1-\cos\omega_c)^d}{(1-\cos\omega_c)^d + \alpha(1+\cos\omega_c)^d}$$

the previous equation can be used to find α as follows

$$2(1-\cos\omega_c)^d = (1-\cos\omega_c)^d + \alpha(1+\cos\omega_c)^d$$
$$\alpha = \left(\frac{1-\cos\omega_c}{1+\cos\omega_c}\right)^d$$

note that, the output of the filter (52) can be defined as $\mathbf{y} = \mathbf{A}^{-1}\mathbf{B}\,\mathbf{x}$; hence, the matrix **B** is banded and has size of $(N-2\mathrm{d})\times N$; also matrix **A** is symmetric square and has size of $(N-2\mathrm{d})\times(N-2\mathrm{d})$, note that the bandwidth of matrices **A** and **B** is 2d+1. In addition, the length of the output is $N-2\mathrm{d}$.

Example: Consider d=2 and ω_c=0.1π, then obtain $\alpha = 6.29\times10^{-4}$. Also the matrix **B** can be obtained from

$$B(\mathrm{z}) = (-\mathrm{z}+2-\mathrm{z}^{-1})^2 = z^2 - 4z + 6 - 4z^{-1} + z^{-2}$$

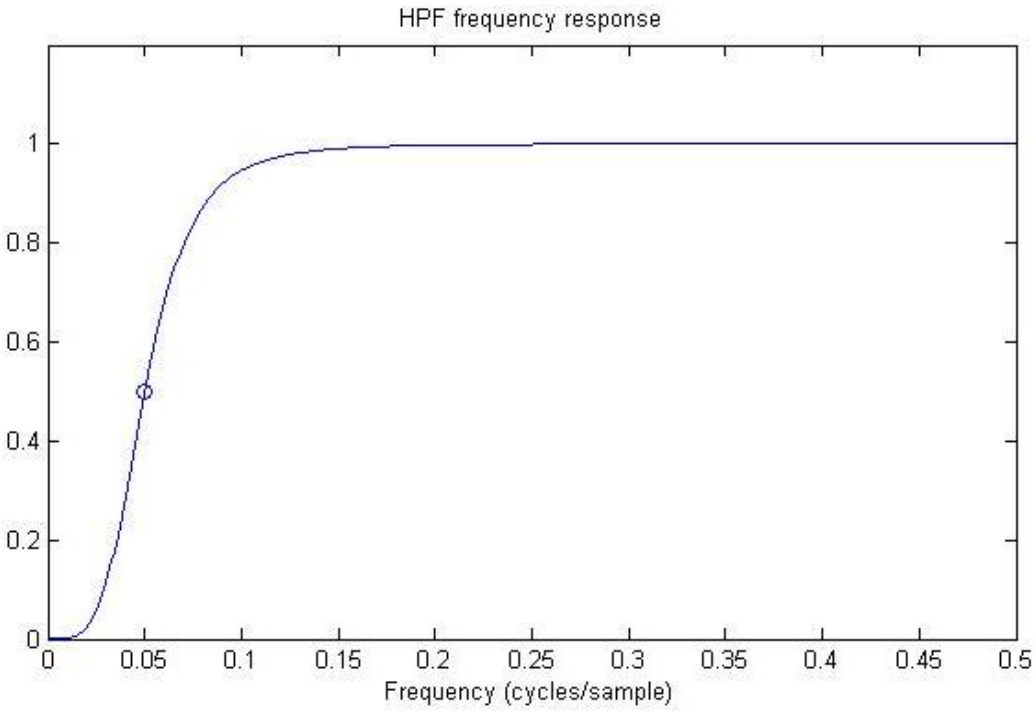
then matrix **B** has the form

$$\mathbf{B}_{(N-4)\times N} = \begin{bmatrix} 1 & -4 & 6 & -4 & 1 & & \\ & 1 & -4 & 6 & -4 & 1 & \\ & & \ddots & & & & \ddots \\ & & & 1 & -4 & 6 & -4 & 1 \end{bmatrix}$$

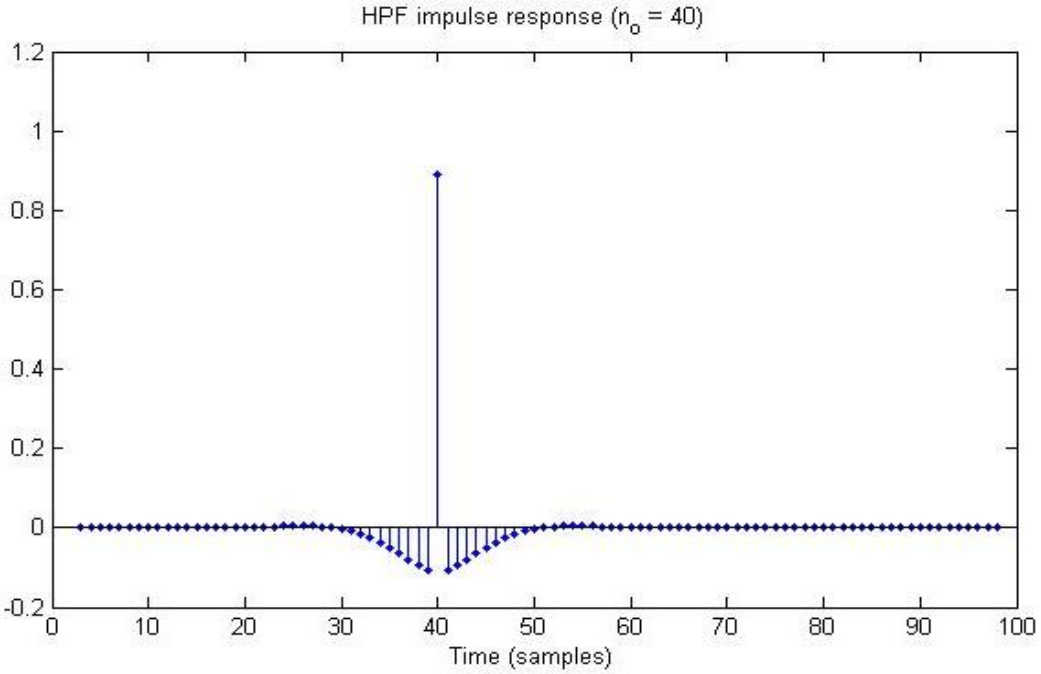
Hence, the bandwidth of **B** is 5, and $\mathbf{B}_1$ is of the size $(N-4)\times(N-1)$ with nonzero elements in each row [-1, 3, -3, 1]. Matrix **A** is square symmetric banded, with bandwidth equal to 5, and has the form

$$\mathbf{A}_{(N-4)\times(N-4)} = \begin{bmatrix} a_0 & a_1 & a_2 & \\ a_1 & a_0 & \ddots & a_2 \\ a_2 & \ddots & \ddots & a_1 \\ & a_2 & a_1 & a_0 \end{bmatrix}$$

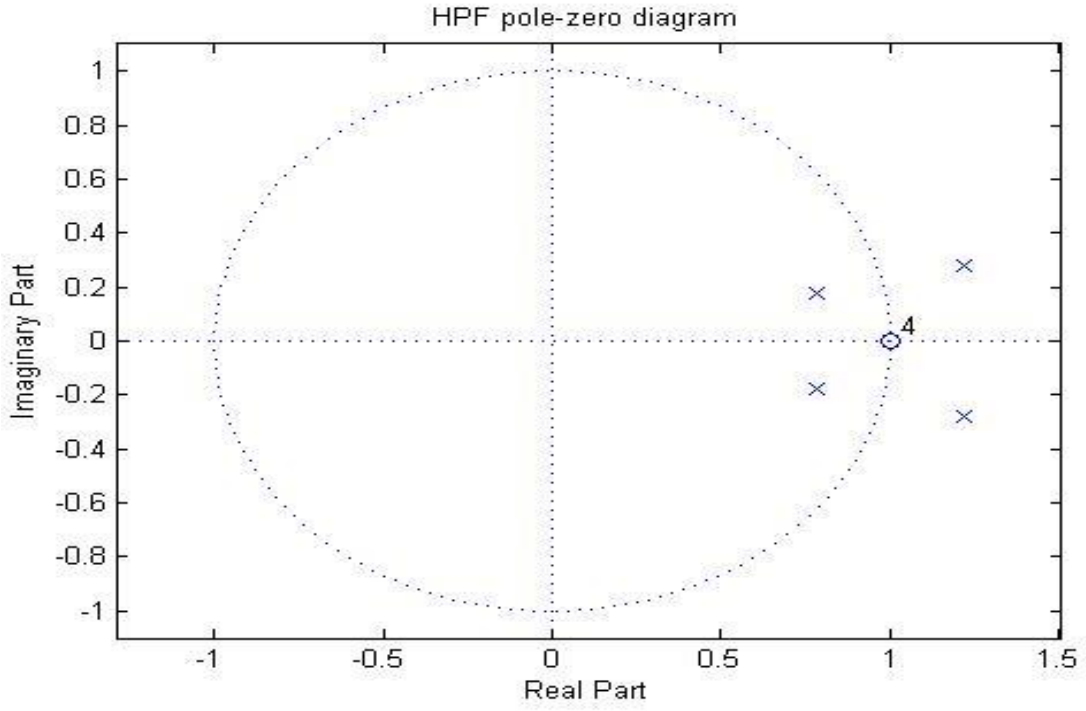
where $a_0 = 6.0038$, $a_1 = -3.9975$, and $a_2 = 1.0006$. The results of the fourth-order high-pass filter are shown in Figure 2.



(a)



(b)



(c)

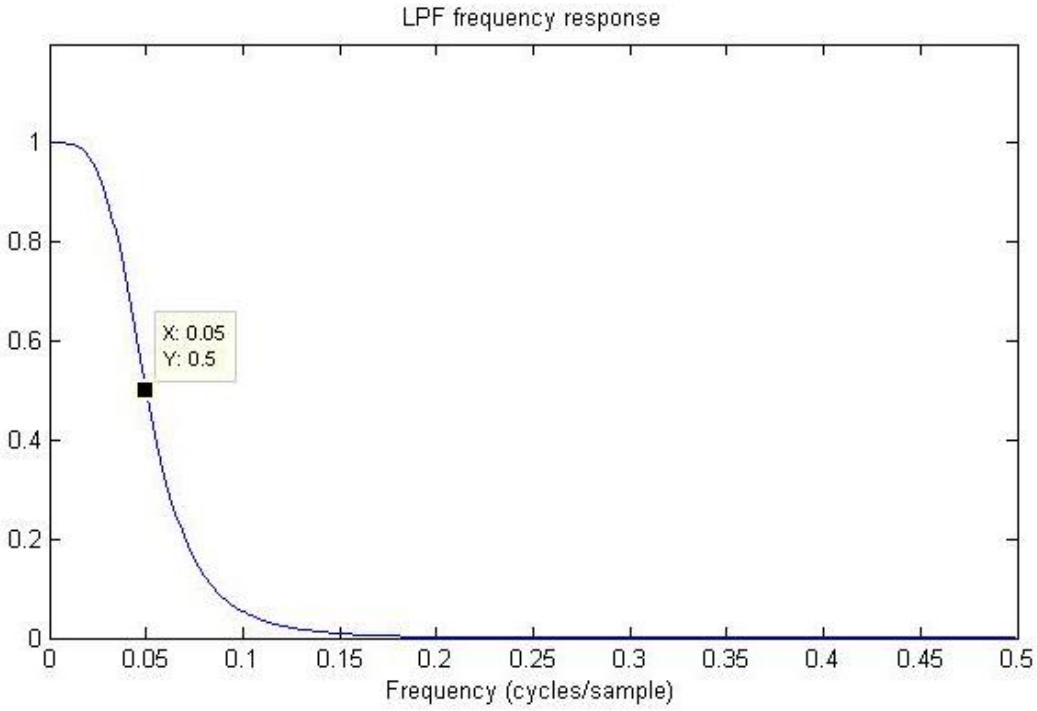
Figure 2: Fourth-order non-causal high-pass filter (52) with d=2 (a) frequency response of the filter (b) impulse response (c) pole-zero diagram.

5.3.3 Low-Pass Filter

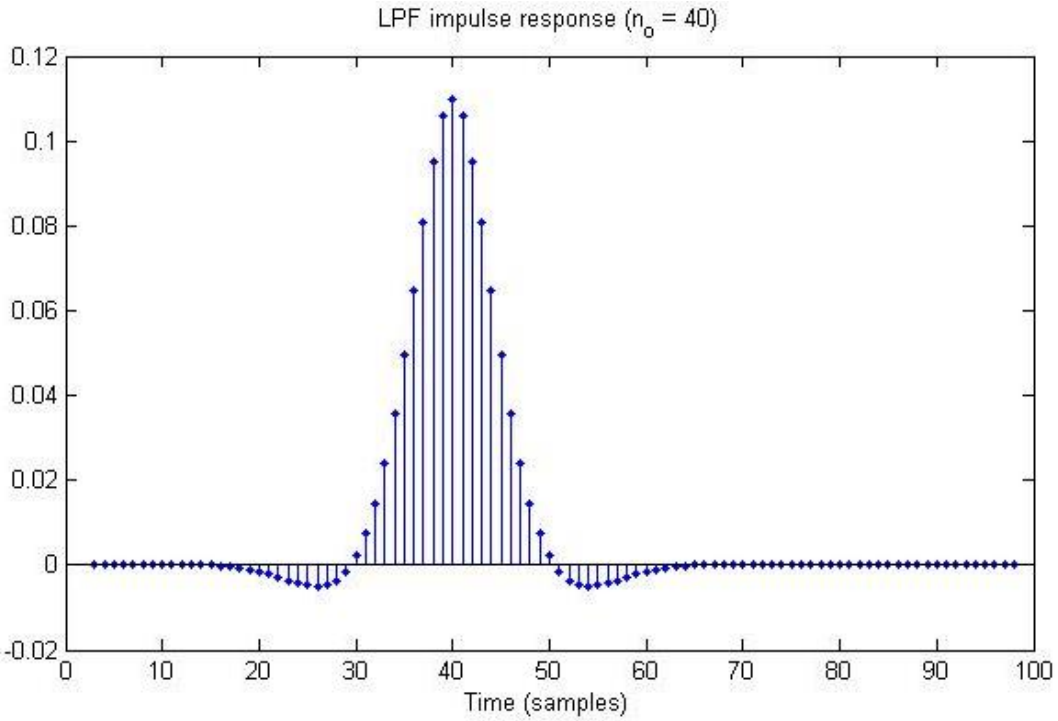
The LPF/TVD algorithm provides an estimate x, of the sparse-derivative component and calls for the high-pass filter $\mathbf{H} = \mathbf{A}^{-1}\mathbf{B}$. The algorithm does not use a low-pass filter. But, to obtain an estimate f of the low-pass component, recall that we need the low-pass filter denoted above as $\mathrm{LPF} = 1 - \mathrm{HPF}$. A low-pass filter of this form is trivially performed by subtracting the high-pass filter output from its input. However, note that for the high-pass filter the matrices **B** and **H** are rectangular. Consequently, the output of the high-pass filter is shorter than its input by 2d samples (d at the beginning and d at the end). Hence, to implement the low-pass filter, the input signal should likewise be truncated so that the subtraction involves vectors of equal length. Consequently, the low-pass filter can be expressed as $\mathrm{LPF(x)} = \mathrm{TRUNC(x)} - \mathrm{HPF(x)}$, where TRUNC(x) denotes the symmetric truncation of x by 2d sample. The low-pass filter matrix L is therefore given by $\mathbf{L} = \bar{\mathbf{I}}_d - \mathbf{A}^{-1}\mathbf{B}$ where $\bar{\mathbf{I}}_d$ is of size $(N-2d)\times N$. the low-pass filter transfer function of filter (52) is given by

$$L(\mathrm{z}) = \frac{\alpha(\mathrm{z}+2+\mathrm{z}^{-1})^d}{(-\mathrm{z}+2-z^{-1})^d + \alpha(\mathrm{z}+2+\mathrm{z}^{-1})^d} \tag{56}$$

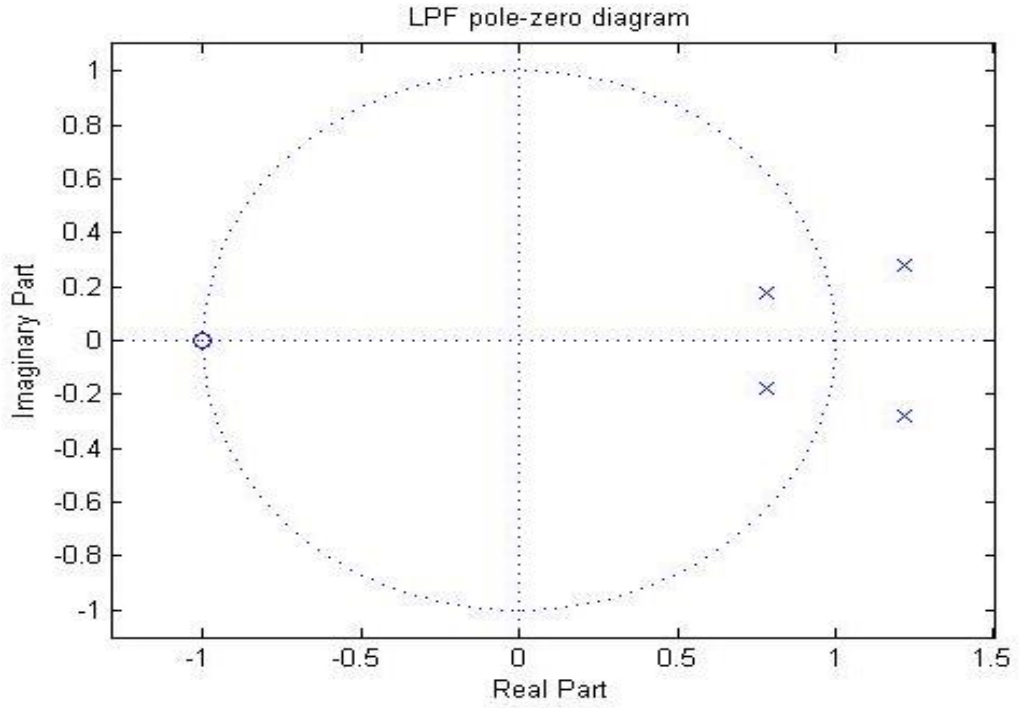
Example: We use high-pass filter illustrated in Figure 2 to implement the low-pass filter shown in Figure 3 with d=2 and cut-off frequency equal 0.1π. The filter output is $\mathbf{y} = \bar{\mathbf{I}}_2\mathbf{x} - \mathbf{A}^{-1}\mathbf{B}\mathbf{x}$.


LPF frequency response
X: 0.05
Y: 0.5
Frequency (cycles/sample)

(a)


LPF impulse response (n_o = 40)
Time (samples)

(b)



(c)

Figure 3: Fourth-order non-causal low-pass filter (52) with d=2 (a) frequency response of the filter (b) impulse response (c) pole-zero diagram.

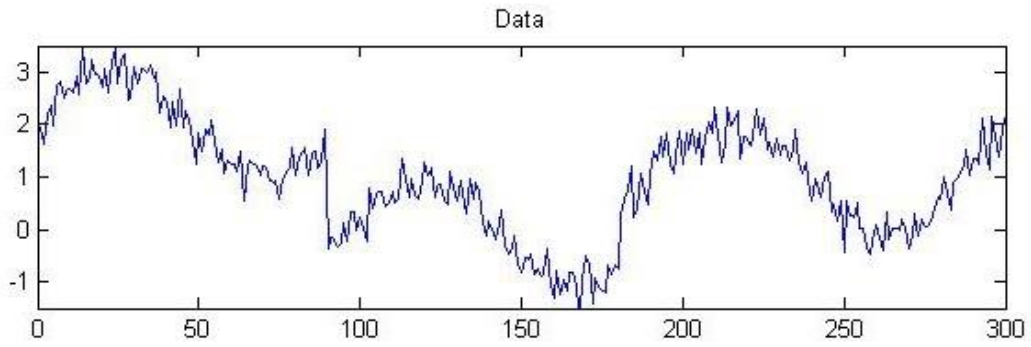
Chapter 6

SIMULATION AND RESULTS

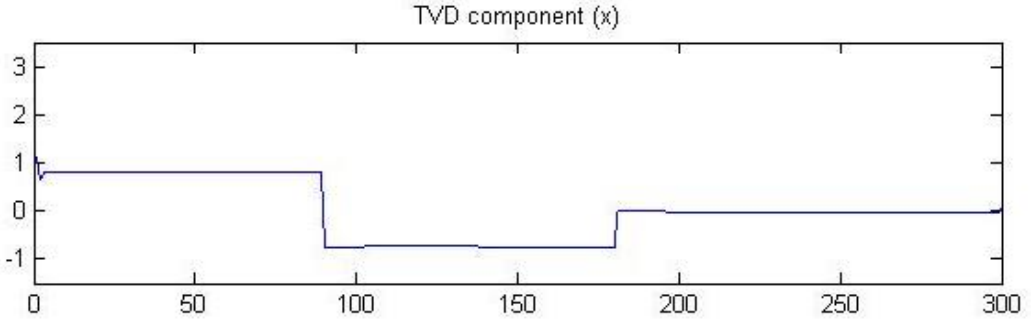
In this chapter we introduce examples and results in order to illustrate the LPF/TVD and LPF/CSD problems in chapter 5.

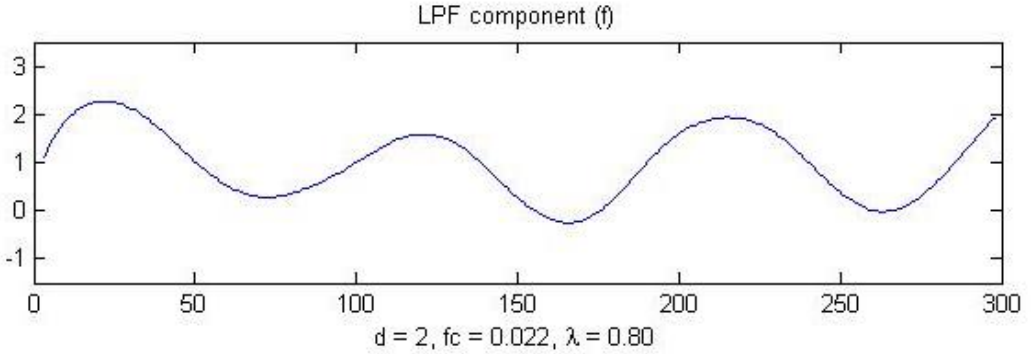
6.1 Example 1 of LPF/TVD Problem

In this section we introduce test signal in order to apply algorithm 1, the test signal contains two step discontinuities, sinusoidal signal, and white Gaussian noise ($\sigma = 0.3$), so we have to specify H and λ , with $\omega_c = 0.044\pi$ and fourth-order high-pass filter (52), the algorithm was run for 30 iterations. Using (32) to set the regularization parameter as $\lambda = 0.8$, the output of algorithm 1 is illustrated in Figure 4. The optimality condition (31) is illustrated in Figure 4(e) as a scatter plot. Each point represents a pair (g(n), u(n)), where g(n) and u(n) denote the n-th time samples of signals g and u. Note that (31) means that if each pair (u, g) lies on the graph of the step function indicated as a dashed line in Figure 4 (e), then the computed x does minimize the objective function in (19). It is seen that most of the points lie on the line, which reflects the sparsity of **Dx** .

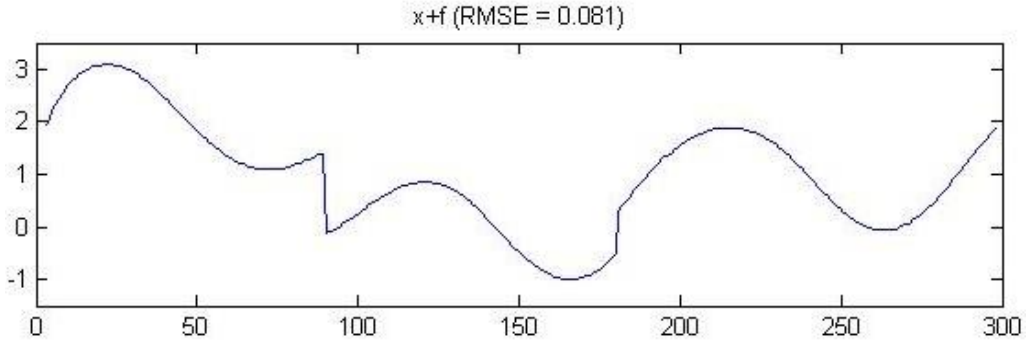

Data
3
2
1
0
-1
0
50
100
150
200
250
300

(a)

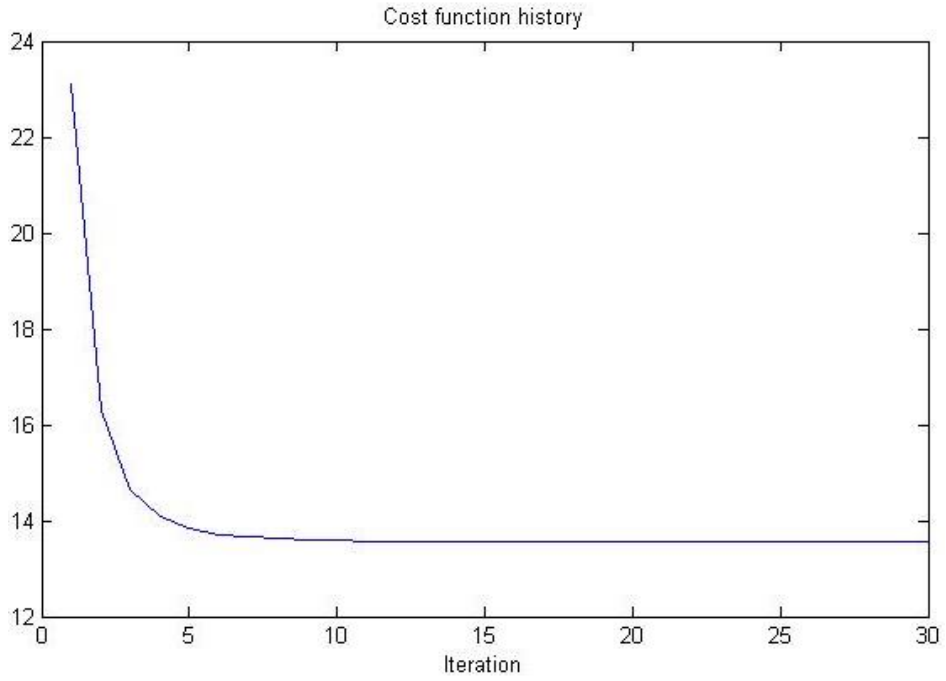

TVD component (x)
3
2
1
0
-1
0
50
100
150
200
250
300


LPF component (f)
3
2
1
0
-1
0
50
100
150
200
250
300
d = 2, fc = 0.022, λ = 0.80

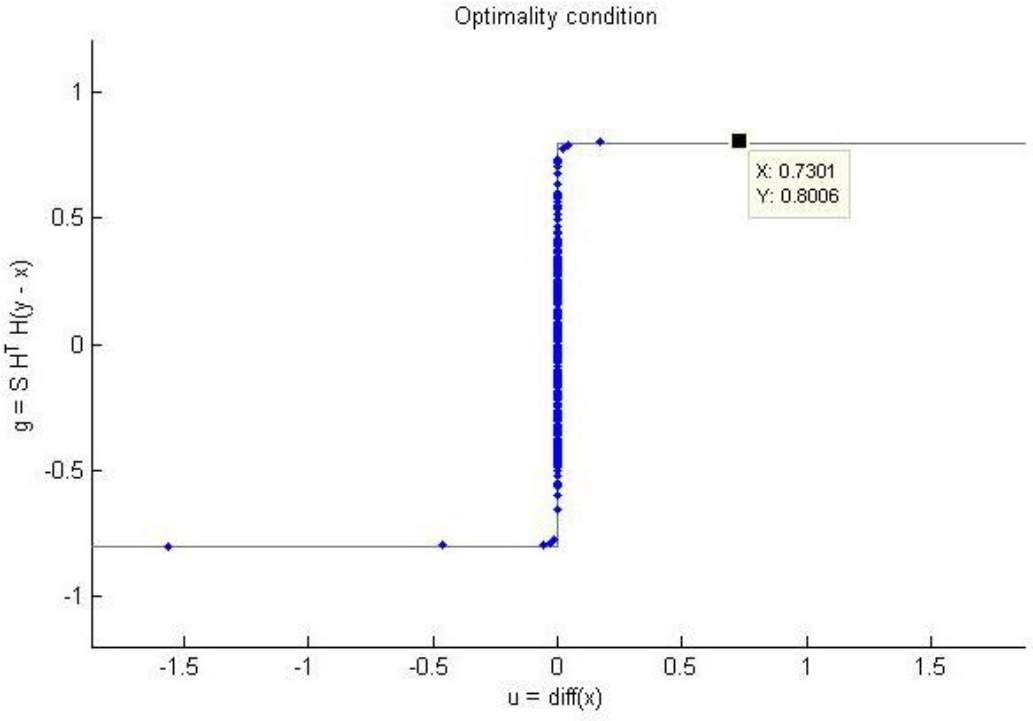
(b)


x+f (RMSE = 0.081)
3
2
1
0
-1
0
50
100
150
200
250
300

(c)



(d)



(e)

Figure 4: Results of example 1 (a) Noisy data (b) sparse-derivative signal x and low-pass signal f (c) sum of two component (d) cost function history (e) the scatter plot of condition(31) is satisfied. Algorithm parameter d=2, $\omega_c = 0.044\pi$, $\lambda = 0.8$.

The scatter plot in Figure 4(e) illustrates the optimality condition (31), and it is clear from the plot that $\lambda = 0.8$. Now to show the effect of the regularization parameter we choose $\lambda = 0.4$; this will increase the root-mean-square error to 0.091 and change the fluctuation of the output signal. Figure 5 illustrates the output of low-pass filtering and total variation denoising when $\lambda = 0.4$.

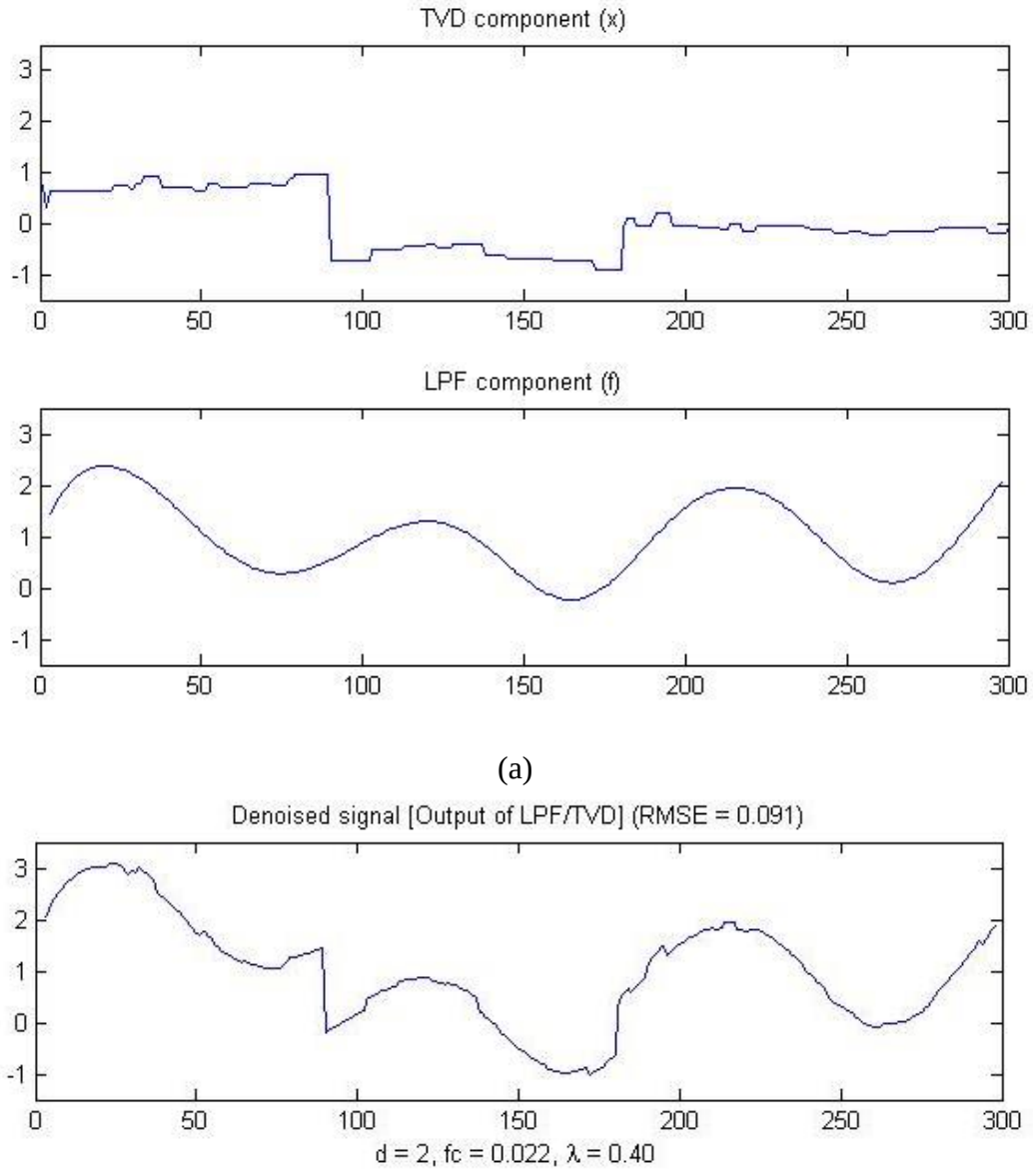
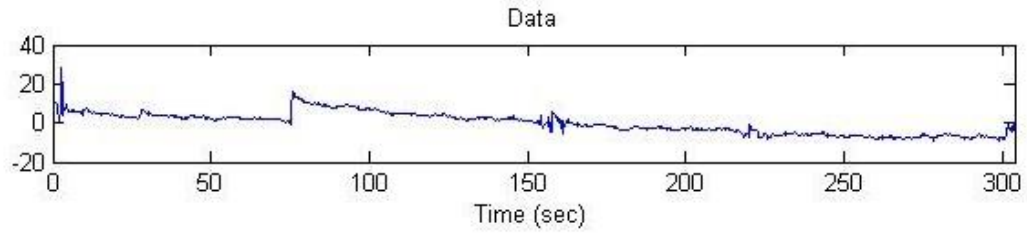


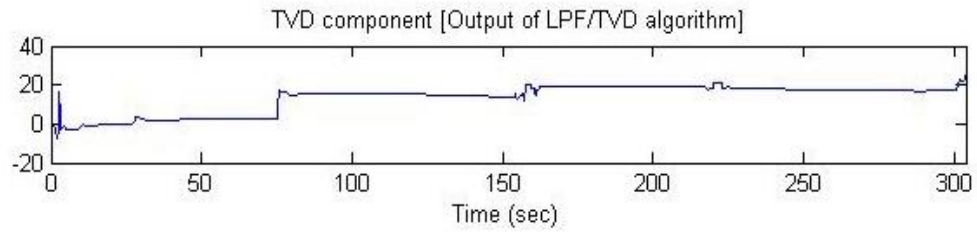
Figure 5: Results of example 1 (a) sparse-derivative signal x and low-pass signal f (b) sum of two component. Algorithm parameter d=2, $\omega_c = 0.044\pi$, $\lambda = 0.4$.

6.2 Example 2 of LPF/TVD Problem

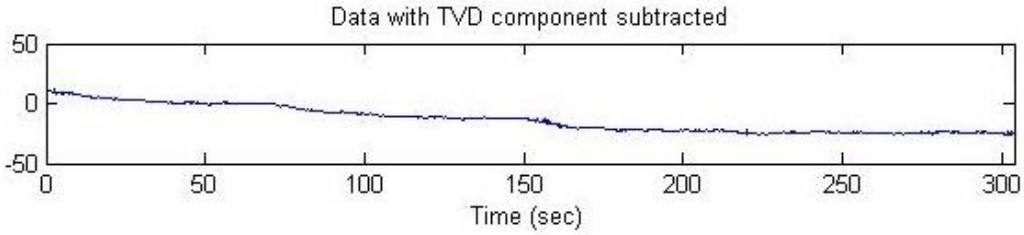
This example used brain signal measured for 304 seconds by keeping the NIRS equipment on back of the head and subjecting it to motion artefacts, with channels that have specific particular source and detector pair. The measurements of NIRS data are very sensitive to subject motion artefacts, which make shift in the baseline value at approximately 80 seconds. In order to apply algorithm 1, we have to specify λ =1.2 with cut-off frequency $\omega_c = 0.04\pi$, and the number of iterations equal to 30 iterations. Figure 6 shows the output of the LPF/TVD algorithm 1 for this example.



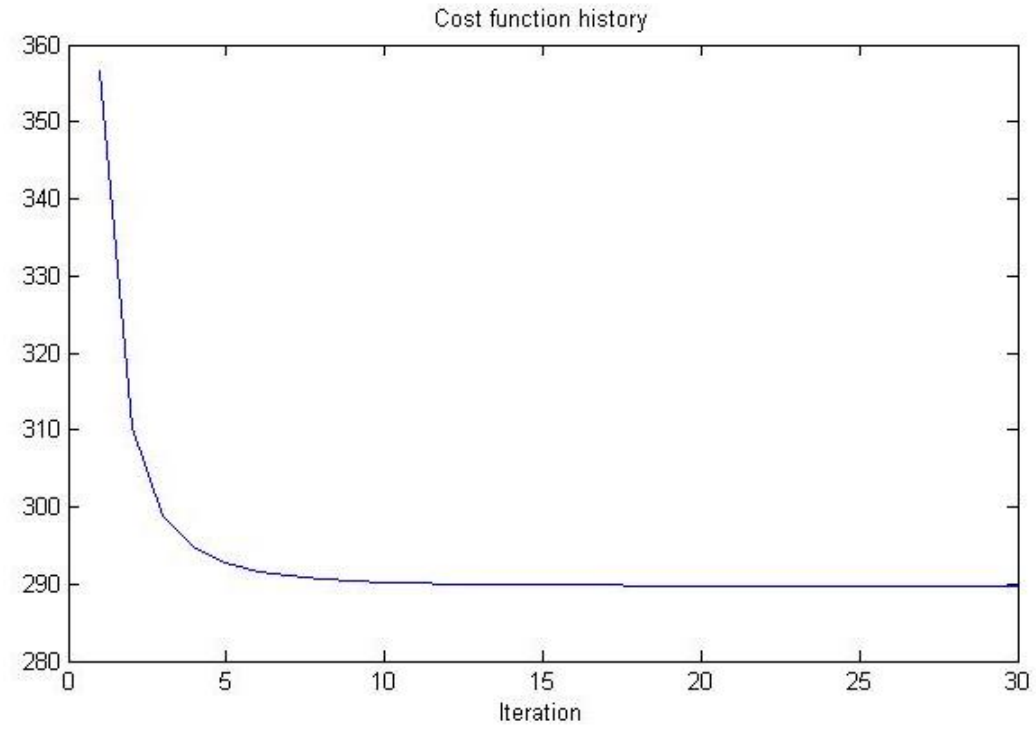
(a)



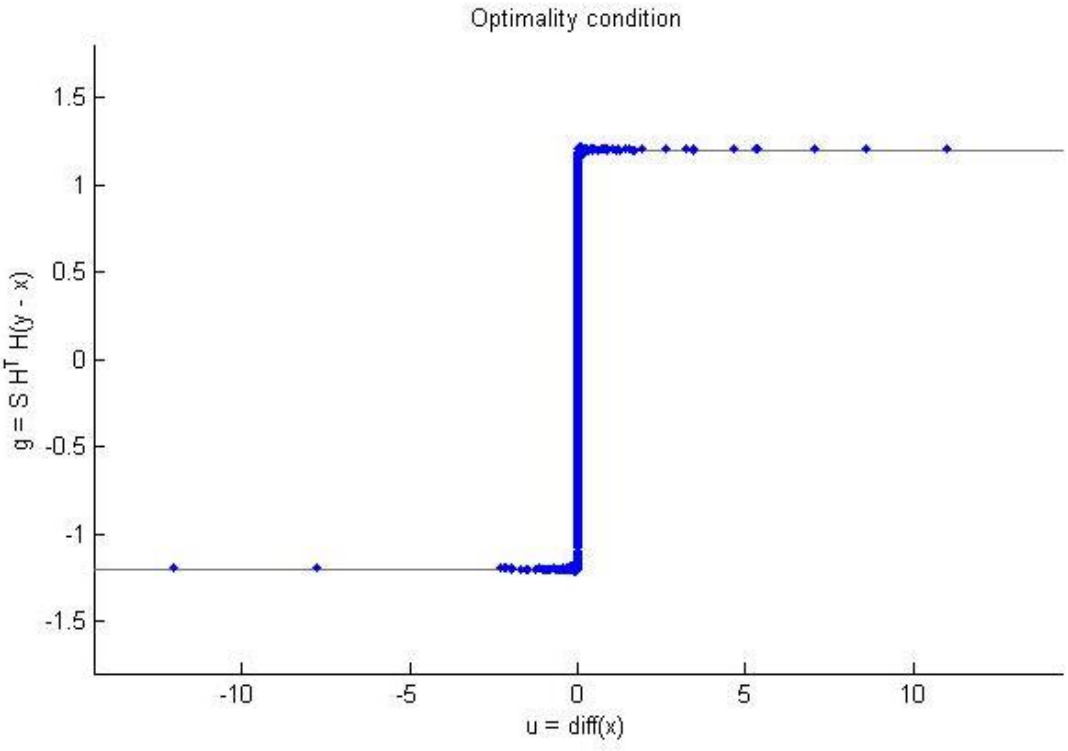
(b)



(c)



(d)



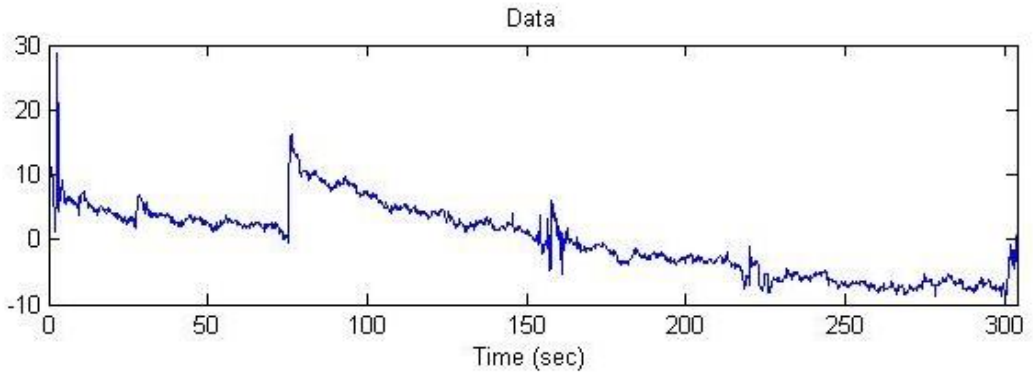
(e)

Figure 6: Results of example 2 (a) NIRS time series (b) TVD component (c) Data with TVD component subtracted (d) cost function history (e) optimality condition. Algorithm parameter d=1, $\omega_c = 0.04\pi$, $\lambda = 1.2$.

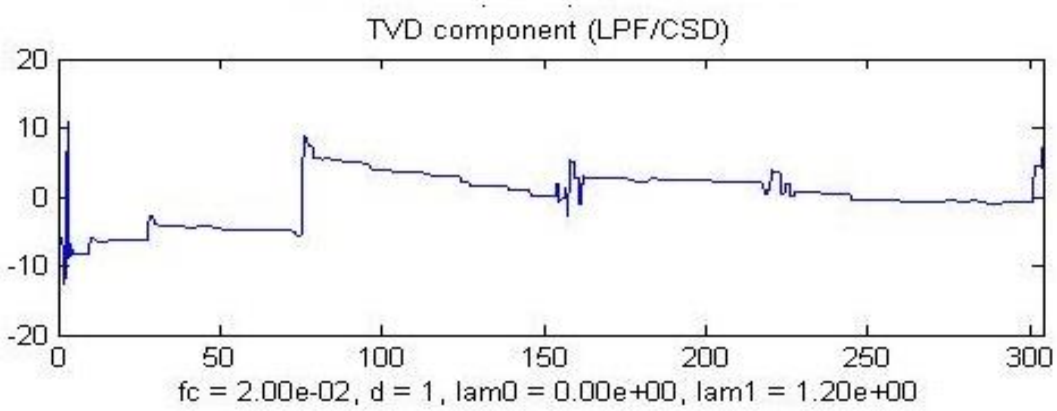
Note that in Figure 6.b the TVD component illustrates the discontinuities present in the NIRS time series data. When $\lambda = 1.2$ the RMSE=0.354 which satisfies the optimality condition.

6.3 Example 3 of LPF/CSD Problem

In this example we used the same data in example 2, in order to filter the data which is sparse and has sparse derivative. Consequently, algorithm 2 is used instead of using algorithm 1 to solve the LPF/CSD problem (33) with $\lambda_0 = 0$. Also we have to specify a parameter μ which affects the overall convergence speed. The vectors **d** and **v** are initialized to zero vectors of the same size of **y**. In this example $\lambda = 1.2$. Figure 7 shows the output of algorithm 2.



(a)

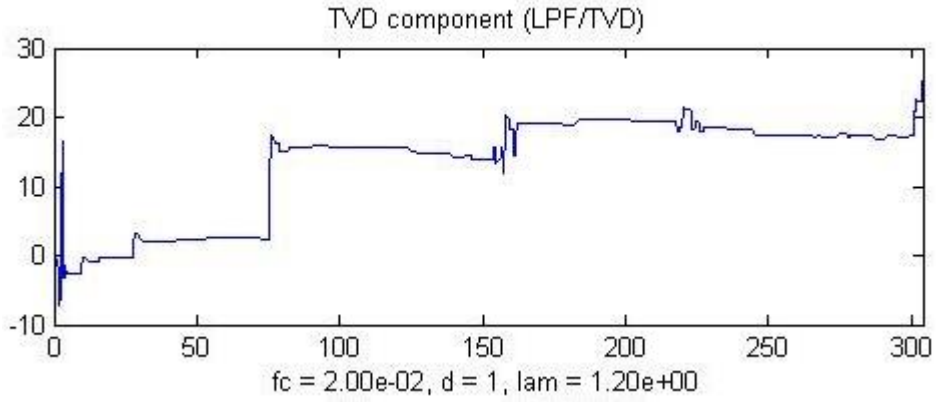


(b)

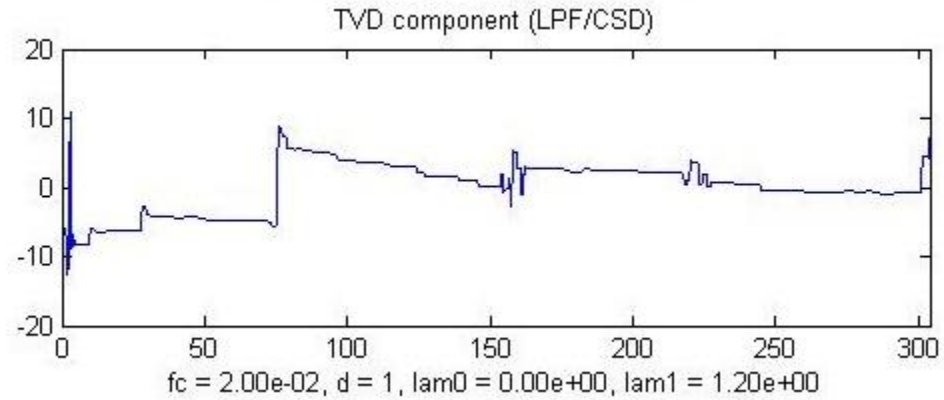
Figure 7: Results of example 3 (a) NIRS time series (b) TVD component for LPF/CSD.

6.4 Comparison between LPF/TVD and LPF/CSD Algorithms

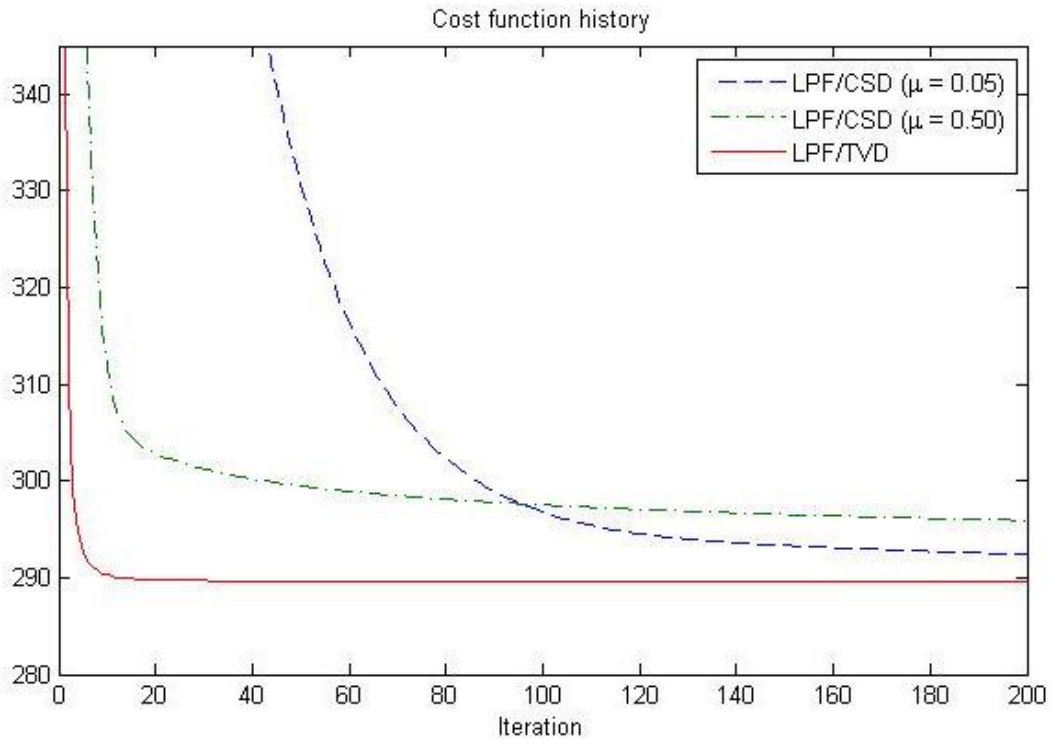
For comparison between algorithm 1 and 2, we have to apply both of them to the same noisy data. The output of each algorithm is illustrated in Figure 8. It is shown that the outputs of the two algorithms are the same, but for algorithm 2 we have to specify μ; this value will not change the solution, it will change the overall convergence. From Figure 8(c) it is observed that LPF/TVD requires less number of iterations and there is no requirement for the specification of the parameter μ compared to LPF/CSD algorithm. It is also observed that for LPF/CSD algorithm for u = 0.05 initial convergence is very poor but final convergence is good compared to u = 0.5, and for u = 0.5 initial convergence is good but final convergence is poor compared to u = 0.05.



(a)



(b)



(c)

Figure 8: Result of comparison between LPF/TVD and LPF/CSD algorithms (a) TVD component for LPF/TVD (b) TVD component for LPF/CSD (c) convergence behaver for algorithms.

6.5 Example 4 of LPF/TVD Problem

To illustrate simultaneous low-pass filtering and total variation denoising for an image, we apply algorithm 1 to a noisy image. We use the second order high-pass filter, and the parameter λ was set to 1.5; also we add noise with $\sigma = 25$. Figure 9 illustrates the use of algorithm 1 on the Barbara picture.

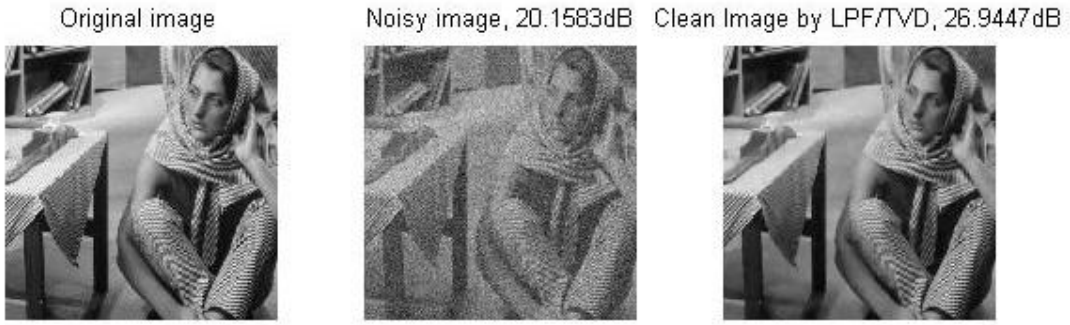


Figure 9: Denoising using LPF/TVD algorithm. Algorithm parameter d=1, $\lambda = 1.5$

6.6 Example 5 of LPF/CSD Problem

To illustrate simultaneous low-pass filtering and total variation denoising for image, we apply algorithm 2 to the noisy image. We use the second-order high-pass filter, and the parameter λ was set to 1.8; also we add noise with $\sigma = 25$. Figure 10 illustrates the use of algorithm 2 on the Barbara picture.

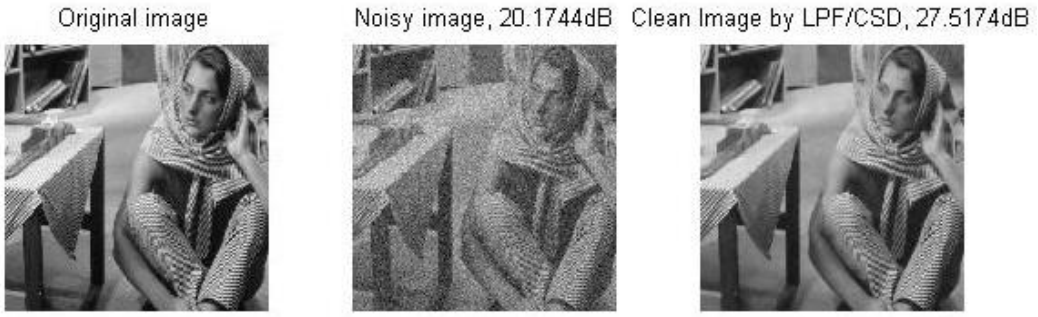


Figure 10: Denoising using LPF/CSD algorithm. Algorithm parameter d=1, $\lambda = 1.8$

6.7 Comparison between LPF/TVD, LPF/CSD, and K-VSD Algorithms

In order to compare the Peak Signal-to-Noise Ratio (PSNR) of Algorithms 1, 2 and K-SVD, we apply all of them to the Barbara picture. Table 3 summarizes the results of the implementations on the Barbara picture.

Table 3: Comparison between algorithms

Algorithm	PSNR (dB)
LPF/TVD	26.9
LPF/CSD	27.5
K-SVD	29.6

Chapter 7

CONCLUSION AND FUTURE WORK

Generally, LTI filtering is suitable to denoise low-frequency signals and total variation denoising is suitable to filter sparse represented signals. By using LTI filters the proposed Near-infrared spectroscopy (NIRS) measured signals are not recovered well. So to recover these types of signals both total variation de-noising and linear time-invariant filtering are effectively combined. In order to filter this type of signals one optimization approach and two algorithms namely LPF/TVD and LPF/CSD are developed. The first one is developed according to the majorization minimization technique, and the second one is developed according to the alternating direction method of multipliers. First algorithm is a special case of the second algorithm. So instead of the first one second algorithm can also be used; but the first algorithm converges faster than the second one and also it doesn't require a step size parameter, whereas second algorithm does. Here for enhanced sparsity total variation denoising and ℓ_1- norm are used.

The proposed algorithms and problem formulation can be used in many fields such as biology, audio signal processing etc. Also, it can be extended in several ways depending on signal classification. Reweighted ℓ_1 minimization [24] or ℓ_p pseudo-norm can be used instead of ℓ_1 norm. In addition, the low-pass filter can be replaced by a notch filter or band-pass filter depending on specific signals.

REFERENCES

[1] Parks, T. W., & Burrus, C. S. (1987). *Digital filter design*: Wiley-Interscience.

[2] Chen, S. S., Donoho, D. L., & Saunders, M. A. (1998). Atomic decomposition by basis pursuit. *SIAM journal on scientific computing, 20*(1), 33-61.

[3] Elad, M., Figueiredo, M. A., & Ma, Y. (2010). On the role of sparse and redundant representations in image processing. *Proceedings of the IEEE, 98*(6), 972-982.

[4] Rao, B. D., Engan, K., Cotter, S. F., Palmer, J., & Kreutz-Delgado, K. (2003). Subset selection in noise based on diversity measure minimization. *Signal Processing, IEEE Transactions on, 51*(3), 760-770.

[5] Selesnick, I. W., Graber, H. L., Pfeil, D. S., & Barbour, R. L. (2014). Simultaneous low-pass filtering and total variation denoising. *Signal Processing, IEEE Transactions on, 62*(5), 1109-1124.

[6] Selesnick, I. W., Arnold, S., & Dantham, V. R. (2012). Polynomial smoothing of Time series with additive step discontinuities. *Signal Processing, IEEE Transactions on, 60*(12), 6305-6318.

[7] Gholami, A., & Hosseini, S. M. (2013). A balanced combination of Tikhonov and total variation regularizations for reconstruction of piecewise-smooth signals. *Signal Processing, 93*(7), 1945-1960.

[8] Aujol, J.-F., Gilboa, G., Chan, T., & Osher, S. (2006). Structure-texture image decomposition—modeling, algorithms, and parameter selection. *International Journal of Computer Vision, 67*(1), 111-136.

[9] Starck, J.-L., Elad, M., & Donoho, D. L. (2005). Image decomposition via the combination of sparse representations and a variational approach. *Image Processing, IEEE Transactions on, 14*(10), 1570-1582.

[10] Burrus, C. S., Gopinath, R. A., & Guo, H. (1997). Introduction to wavelets and wavelet transforms: a primer.

[11] Jacques, L., Duval, L., Chaux, C., & Peyré, G. (2011). A panorama on multiscale geometric representations, intertwining spatial, directional and frequency selectivity. *Signal Processing, 91*(12), 2699-2730.

[12] Bruni, V., & Vitulano, D. (2006). Wavelet-based signal de-noising via simple singularities approximation. *Signal Processing, 86*(4), 859-876.

[13] Durand, S., & Nikolova, M. (2007). Denoising of frame coefficients using ℓ-1 data-fidelity term and edge-preserving regularization. *Multiscale Modeling & Simulation, 6*(2), 547-576.

[14] Geman, D., & Reynolds, G. (1992). Constrained restoration and the recovery of discontinuities. *IEEE Transactions on Pattern Analysis & Machine*

Intelligence(3), 367-383.

[15] Barbour, R. L., Graber, H. L., Xu, Y., Pei, Y., Schmitz, C. H., Pfeil, D. S., Barbour, S.-L. S. (2012). A programmable laboratory testbed in support of evaluation of functional brain activation and connectivity. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 20*(2), 170-183.

[16] Combettes, P. L., & Pesquet, J.-C. (2007). Proximal thresholding algorithm for minimization over orthonormal bases. *SIAM Journal on Optimization, 18*(4), 1351-1376.

[17] Figueiredo, M. A., Bioucas-Dias, J. M., & Nowak, R. D. (2007). Majorization–minimization algorithms for wavelet-based image restoration. *Image Processing, IEEE Transactions on, 16*(12), 2980-2991.

[18] Figueiredo, M., Dias, J. B., Oliveira, J. P., & Nowak, R. D. (2006). *On total variation denoising: A new majorization-minimization algorithm and an experimental comparisonwith wavalet denoising.* Paper presented at the Image Processing, 2006 IEEE International Conference on.

[19] Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., & McKenney, A. (1999). *LAPACK Users' guide* (Vol. 9): Siam.

[20] Bach, F., Jenatton, R., Mairal, J., & Obozinski, G. (2012). Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning.*

[21] Friedman, J., Hastie, T., & Holger, H. (2007). *xf6; fling, and Robert Tibshirani. Pathwise coordinate optimization*. Retrieved from

[22] Boyd, S., Parikh, N., Chu, E., Peleato, B., & Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning, 3*(1), 1-122.

[23] Afonso, M. V., Bioucas-Dias, J. M., & Figueiredo, M. A. (2010). Fast image recovery using variable splitting and constrained optimization. *Image Processing, IEEE Transactions on, 19*(9), 2345-2356.

[24] Candes, E. J., Wakin, M. B., & Boyd, S. P. (2008). Enhancing sparsity by reweighted ℓ-1 minimization. *Journal of Fourier analysis and applications, 14*(5-6), 877-905.