# SPARSITY AND CONVEX PROGRAMMING IN TIME-FREQUENCY PROCESSING

A DISSERTATION SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

By
Zeynel Deprem
December, 2014

SPARSITY AND CONVEX PROGRAMMING IN TIME-FREQUENCY PROCESSING

By Zeynel Deprem

December, 2014

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Prof. Dr. Ahmet Enis Çetin (Advisor)

Prof. Dr. Orhan Arıkan

Prof. Dr. M. Kemal Leblebicioğlu

Prof. Dr. Uğur Güdükbay

Assoc. Prof. S. Serdar Kozat

Approved for the Graduate School of Engineering and Science:

Prof. Dr. Levent Onural
Director of the Graduate School

# ABSTRACT

## SPARSITY AND CONVEX PROGRAMMING IN TIME-FREQUENCY PROCESSING

Zeynel Deprem
Ph.D. in Electrical and Electronics Engineering
Advisor: Prof. Dr. Ahmet Enis Çetin
December, 2014

In this thesis sparsity and convex programming-based methods for time-frequency (TF) processing are developed. The proposed methods aim to obtain high resolution and cross-term free TF representations using sparsity and lifted projections. A crucial aspect of Time-Frequency (TF) analysis is the identification of separate components in a multi component signal. Wigner-Ville distribution is the classical tool for representing such signals but suffers from cross-terms. Other methods that are members of Cohen's class distributions also aim to remove the cross terms by masking the Ambiguity Function (AF) but they result in reduced resolution. Most practical signals with time-varying frequency content are in the form of weighted trajectories on the TF plane and many others are sparse in nature. Therefore the problem can be cast as TF distribution reconstruction using a subset of AF domain coefficients and sparsity assumption in TF domain. Sparsity can be achieved by constraining or minimizing the $l_1$ norm. Projections Onto Convex Sets (POCS) based $l_1$ minimization approach is proposed to obtain a high resolution, cross-term free TF distribution. Several AF domain constraint sets are defined for TF reconstruction. Epigraph set of $l_1$ norm, real part of AF and phase of AF are used during the iterative estimation process. A new kernel estimation method based on a single projection onto the epigraph set of $l_1$ ball in TF domain is also proposed. The kernel based method obtains the TF representation in a faster way than the other optimization based methods. Component estimation from a multicomponent time-varying signal is considered using TF distribution and parametric maximum likelihood (ML) estimation. The initial parameters are obtained via time-frequency techniques. A method, which iterates amplitude and phase parameters separately, is proposed. The method significantly reduces the computational complexity and convergence time.

*Keywords:* sparsity, time-frequency distribution, Cohen's class distributions, Polynomial Phase.

# ÖZET

# SEYREKLİK VE KONVEKS PROGRAMLAMA İLE ZAMAN-FREKANS İŞLEME

Zeynel Deprem
Elektrik ve Elektronik Mühendisliği, Doktora
Tez Danışmanı: Prof. Dr. Ahmet Enis Çetin
Aralık, 2014

Bu tezde zaman-frekans (TF) sinyallerini işlemek için seyreklik ve konveks programlamaya dayalı yöntemler geliştirilmiştir. Önerilen yöntemler, seyreklik ve yükseltilmiş izdüşüm kullanarak yüksek çözünürlüklü ve çapraz terim içermeyen zaman-frekans dağılımları elde etmeyi hedeflemektedir. Zaman frekans çözümlemenin en önemli yönü, çok bileşenli bir sinyalde ayrı bileşenlerin ayırt edilmesidir. Bu tür sinyallerin gösteriminde klasik bir araç olan Wigner-Ville dağılımı kullanılır, fakat çapraz terimler içerir. Cohen sınıfındaki diğer dağılımlar da çapraz terimleri, Belirsizlik Fonksiyonunu (AF) maskeleme ile yok etmeye çalışır, fakat bu çözünürlüğün azalmasına sebep olur. Frekans içeriği zamana bağımlı değişim gösteren bir çok sinyal TF düzleminde ağırlıklandırılmış izler şeklindedir, bir çoğu da seyrek bir yapıya sahiptir. Bu sebeple problem, TF dağılımının, AF düzlemindeki bir altkümenin ve seyreklik kullanılarak oluşturulması olarak gözönüne alınabilir. Seyreklik, $l_1$ normunu koşut koyarak veya azaltarak elde edilebilir. Yüksek çözünürlüklü ve çapraz terim içermeyen TF dağılım elde etmek için, dışbükey kümeler üzerine iz düşüm (POCS)'e dayalı $l_1$ azaltma yöntemi önerilmektedir. TF dağılım oluşturmak için çeşitli AF düzlemi kümeleri tanımlanmaktadır. Tekrarlı kestirim sürecinde AF düzlemindeki değerlerin reel kısmı, faz kısmı ve $l_1$ normuna ait epigraf kümesi kullanılmaktadır. TF düzlemindeki $l_1$ normuna ait epigraf kmesi üzerine tek bir izdüşüme dayalı yeni bir çekirdek kestirim yöntemi önerilmektedir. Çekirdeğe dayalı yöntem, diğer optimizasyon tabanlı yöntemlere göre TF dağılımını daha hızlı elde etmektedir. Zamana göre değişen çok bileşenli bir sinyalden bileşenleri kestirmek için TF dağılımı ve parametrik en büyük olabilirlik (ML) kestirim yöntemi kullanılmaktadır. Başlangıç parametreleri zaman-frekans yöntemleri ile elde edilmektedir. Genlik ve faz adımlarını ayrı ayrı ilerleten bir yöntem önerilmektedir. Yöntem işlem karmaşıklığını ve yakınsama süresini önemli derecede azaltmaktadır.

*Anahtar sözcükler*: Seyreklik, zaman-frekans dağılımı, Cohen sınıfı dağılımlar, Polinom Faz.

# Acknowledgement

First of all I would like to thank Prof. Çetin for allowing me to restart my studies that I have suspended more than a decade before and giving me valuable guidance during my research path. He has showed a patience of job against me. I would like to thank him for all his patience and belief in me.

I would like to thank Prof. Arıkan, for his valuable guidance and for his patience against me. I also would like to thank for his suggestions, and encouragement through the development of this thesis.

I would like to thank Prof. Leblebicioğlu for his valuable guidance and for allocating his time to me whenever I need. I would like to thank him not just because of his technical suggestions and talks but also for his social talks during our meetings.

Special thanks to Prof. Dr. Uğur Güdükbay and Assoc. Prof. S. Serdar Kozat for reading and commenting on the thesis.

I dedicate a big portion of this thesis to my son Eren, from him I borrowed the time to complete this study. I accept that some of that time will not be compensated but I hope he will understand me in future.

I would in particular like to thank my wife, Serpil, who encouraged me to restart my PhD studies. I also would like to thank her for giving me the motivation in cases when I felt pessimistic.

I would like to thank my cousin Taylan both for the enjoying times we had, when we were together, and for his sincere interest in how my PhD studies are going.

# Contents

**1 Introduction** **1**

**2 Time-frequency Representations and Sparsity-based Signal Reconstruction** **5**

2.1 Review of Time-frequency Representations . . . . . 5

2.2 Compressive Sensing . . . . . 11

**3 Cross term-free Time-frequency Distribution Reconstruction via Lifted Projections** **16**

3.1 Sparsity-based Time-frequency Distribution Reconstruction . . . 16

3.2 Time-frequency Distribution Reconstruction with Lifted POCS . . 22

3.3 Projection onto the sets $C_f$ and $C_{AF}$ and the iterative algorithm . 26

3.4 Experimental Results . . . . . 28

**4 Cross term-free TF Reconstruction using Partial AF Information** **44**

4.1 Introduction . . . . . 44

4.2 Reconstruction with Real Part of AF Coefficients . . . . . 46

4.3 Reconstruction with only the Phase of AF Coefficients . . . . . 51

**5 Smoothing Kernel Estimation by Projection onto the Epigraph Set of $l_1$ norm** **57**

5.1 Kernel Design with Optimization . . . . . 57

5.2 Kernel Estimation by Projection onto the Epigraph Set of $l_1$ Norm 60

5.3 Simulation Results . . . . . 74

**6 Mixed TF and Parametric Component Estimation for Time-varying Signals 82**

6.1 Introduction . . . . . 82

6.2 Problem Formulation and ML Estimation . . . . . 85

6.3 Alternating Phase and Amplitude Method . . . . . 88

6.3.1 Analysis of Computational Cost . . . . . 94

6.3.2 Expectation Maximization with Alternating Phase and Amplitude Method . . . . . 99

6.3.3 Cramer-Rao Bounds for Mean Square Reconstruction Error 102

6.3.4 Simulation Results . . . . . 103

6.4 Parameter Estimation with Sparsity Constraint . . . . . 114

**7 Conclusion 117**

**Bibliography 120**

**Appendices 132**

**A The Projection onto Epigraph Set of a Convex Function 132**

A.1 The Projection onto Epigraph Set by Succesive Projections . . . 132

A.2 The pseudo-code for projection onto epigraph set of a convex cost function . . . . . 134

**B Signal Examples Used in Simulations 135**

**C Hadamard Product 139**

# List of Figures


3.1 Effect of shaping the ambiguity function on the WV distribution: top-left: Ambiguity Function of the time-varying signal: top-right WV distribution: bottom-left: masked Ambiguity Function; bottom-right: WV distribution corresponding to the masked Ambiguity Function. The horizontal and the vertical axes show the time and the normalized frequency, respectively. . . . . 17

3.2 TF distribution obtained by the minimization of (3.3) using $l_1$-MAGIC TOOLBOX (top) and its 3D plot. The frequency is normalized according to the sampling frequency. . . . . 20

3.3 Reassigned Smoothed Pseudo WV (RSPWV) distribution and its 3D plot obtained by using the Time-Frequency-Toolbox [1]. . . . . 21

3.4 Left: POCS iterates converge to a vector in the intersection of convex sets $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$. The vector $\boldsymbol{x}_0$ is the initial vector and $\boldsymbol{x}^*$ is in the intersection of sets $\boldsymbol{C}_1$and $\boldsymbol{C}_2$. Right: Iterates oscillate between the two vectors when the intersection is empty. Vectors $\boldsymbol{x}_1^*$ and $\boldsymbol{x}_2^*$ minimize the distance between the sets $\boldsymbol{C}_1$and $\boldsymbol{C}_2$. . . . 23

3.5 Graphical representation of the epigraph set of $l_1$ norm $C_f$ and the projection of the vector $\boldsymbol{P}_0$ onto the set $C_f$ . . . . . 25

3.6 Signal 1: the TF reconstruction using, left column: the ideal model, Spectrogram, $l_1$-MAGIC TOOLBOX, L-Class Polynomial WV distribution (LPWVD), right column: WV, the Smoothed Pseudo WV (SPWV), Reassigned SPWV (RSPWV), lifted POCS. 30

3.7 Signal 1: 3D plot of the TF distribution obtained by lifted POCS. 31

3.8 Signal 2: The TF reconstruction using, left column: the ideal model, Spectrogram, $l_1$-MAGIC TOOLBOX, L-Class Polynomial WV distribution (LPWVD), right column: WV, the Smoothed Pseudo WV (SPWV), Reassigned SPWV (RSPWV), lifted POCS. 32
3.9 Signal 2: 3D plot of the TF distributions obtained by reassigned SPWV (RSPWV) and Lifted POCS (bottom). . . . . 33
3.10 Signal 3: the TF reconstruction using, left column: the ideal model, Spectrogram, $l_1$-MAGIC TOOLBOX, L-Class Polynomial WV distribution (LPWVD), right column: WV, the Smoothed Pseudo WV (SPWV), Reassigned SPWV (RSPWV), lifted POCS. 34
3.11 Signal 3: 3D plot of the TF distributions obtained by Reassigned SPWV (RSPWV) and Lifted POCS (bottom). . . . . 35
3.12 Signal 4: the TF reconstruction using, left column: the ideal model, Spectrogram, $l_1$-MAGIC TOOLBOX, L-Class Polynomial WV distribution (LPWVD), right column: WV, the Smoothed Pseudo WV (SPWV), Reassigned SPWV (RSPWV), Lifted POCS. 36
3.13 Signal 4: 3D plot of the TF distributions obtained by Reassigned SPWV (RSPWV) and Lifted POCS (bottom). . . . . 37
3.14 The convergence plot of the lifted POCS iterations for Signal 1: The plot shows the $l_1$-norm of the TF distribution versus the number of iterations. . . . . 38
3.15 The signal in Figure 3.8 is corrupted by additive zero-mean white Gaussian noise. The $SNR$ value is 10 dB. The TF reconstruction result obtained by Lifted POCS method (bottom right) is comparable to the Reassigned Smoothed Pseudo WV (top right). The frequency is the normalized frequency. . . . . 39
3.16 Signal 5: TFD of a Frequency Hopping MFSK signal. First row: the ideal model and Reassigned SPWV(RSPWV). Bottom row: L-Class Polynomial WV Distribution (LPWVD) and Lifted POCS. The frequency is normalized. . . . . 40

3.17 Signal 6. TFD of a dolphin's click-signal segment. Top-row: spectrogram (SP) and Reassigned SPWV (RSPWV). Bottom-row: L-Class Polynomial WV Distribution (LPWVD) and Lifted POCS. The frequency is normalized frequency. . . . . 40

4.1 Signal 1: the TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right). . . . . 48
4.2 Signal 2: the TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right). . . . . 48
4.3 Signal 3: the TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right). . . . . 49
4.4 Signal 4: the TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right). . . . . 49
4.5 Signal 5: TFD of a Frequency Hopping MFSK signal. The TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right). . . . . 50
4.6 Signal 6: TFD of a dolphin's click-signal. The TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right). 50
4.7 Signal 1: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right). . . . . 54

4.8 Signal 2: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right). . . . . 54
4.9 Signal 3: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right). . . . . 55
4.10 Signal 4: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right). . . . . 55
4.11 Signal 6: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right). . . . . 56
4.12 Signal 4: the TF reconstruction with initial TF as an impulse given by 4.10 at origin of TF plane: WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right). 56

5.1 Smoothing the WV distribution with a circular kernel (bottom left $r_0 = N/16$) and with Gaussian kernel (bottom right,$\sigma(\phi) = N/16$) 59
5.2 Graphical representation of the de-noising process using projection onto the epigraph set of $l_1$ cost function, where $vec(\boldsymbol{P}) \in \boldsymbol{R}^{N^2}$ and $\boldsymbol{w} = [vec(\boldsymbol{P})^T \quad v]^T \in \boldsymbol{R}^{N^2+1}$ in the lifted domain . . . . . 62
5.3 Projecting the initial smoothed TF with Gaussian kernel onto the epigraph set of $l_1$ function results in an over localized solution which is not acceptable as the TF distribution corresponding to the original signal. . . . . 63
5.4 Projecting the initial smoothed TF with Gaussian kernel onto the epigraph set of $l_1$ function results in an over localized solution which is not acceptable as the TF distribution corresponding to the original signal. . . . . 64

5.5 The initial Gaussian kernel with $\sigma = N/16$ (top), $N$ being the signal length, and the estimated kernel from initial smoothed TF and epigraph projection (bottom). While the support of initial kernel is circular or has the same $\sigma$ in all directions, the estimated one is aligned in Doppler direction which is in accordance with signal layout in TF plane shown in Figure 5.3 (top) . . . . . 65

5.6 The initial Gaussian kernel with $\sigma = N/16$ (top), $N$ being the signal length, and the estimated kernel from initial smoothed TF and epigraph projection (bottom). While the support of initial kernel is circular or has the same $\sigma$ in all directions, the estimated one is aligned in Doppler direction which is in accordance with signal layout in TF plane shown in Figure 5.3 (top) . . . . . 66

5.7 Comparison of TF smoothing with estimated kernel to other methods for the Example signal 1: left column: the ideal model, fixed kernel or Spectrogram(SP), reassigned SPWV (RSPWV), lifted POCS, right column: WV distribution, the Smoothed Pseudo WV (SPWV) distribution,TF with optimal kernel($\alpha = 1.4$), TF with the estimated kernel. . . . . 68

5.8 Comparison of TF smoothing with estimated kernel to other methods for the Example signal 2: left column: the ideal model, fixed kernel or Spectrogram(SP), reassigned SPWV (RSPWV), lifted POCS, right column: WV distribution, the Smoothed Pseudo WV (SPWV) distribution,TF with optimal kernel($\alpha = 1.4$), TF with the estimated kernel. . . . . 69

5.9 The estimated kernel has spurious structures due to initial coarse low-pass filter. In order to remove them a a Gaussian mask is applied to the initial smoothed TF distribution before epigraph projection. The estimated kernel without Gaussian mask is shown on top and the estimation with Gaussian mask is shown at bottom. 72

5.10 The estimated kernel has spurious structures due to initial coarse low-pass filter. In order to remove them a Gaussian mask is applied to the initial smoothed TF distribution before epigraph projection. The estimated kernel without Gaussian mask is shown on top and the estimation with Gaussian mask is shown at bottom. . . . . 73
5.11 The support of the second Gaussian mask applied to initial TF at Step 2 of Table 5.1 (red corresponds to 1 and blue color corresponds to zero). . . . . 74
5.12 The optimized kernel (top) and the estimated kernel (bottom). . 75
5.13 The optimized kernel (top) and the estimated kernel (bottom). . 76
5.14 Comparison of TF smoothing with estimated kernel to other methods for the Example signal 3: left column: The ideal model, fixed kernel or Spectrogram(SP), reassigned SPWV (RSPWV), lifted POCS, right column: WV distribution, the Smoothed Pseudo WV (SPWV) distribution,TF with optimal kernel($\alpha = 1.4$), TF with the estimated kernel. . . . . 78
5.15 Comparison of TF smoothing with estimated kernel to other methods for the Example signal 4: left column: The ideal model, fixed kernel or Spectrogram(SP), reassigned SPWV (RSPWV), lifted POCS, right column: WV distribution, the Smoothed Pseudo WV (SPWV) distribution,TF with optimal kernel($\alpha = 1.4$), TF with the estimated kernel. . . . . 79
5.16 Comparison of TF smoothing with estimated kernel to other methods for the Example signal 5: left column: The result with Smoothed Pseudo WV (SPWV), lifted POCS, right column: TF with the optimal kernel, TF with the estimated kernel. . . . . 80
5.17 Comparison of TF smoothing with estimated kernel to other methods for the Example signal 6: left column: The result with Smoothed Pseudo WV (SPWV), lifted POCS, right column: TF with the optimal kernel, TF with the estimated kernel. . . . . 80

6.1 The multi-component signals for Ex1 (top) and Ex2 (bottom) with two components. . . . . 104
6.2 The multi-component signal Ex3 with 3 components. . . . . 105

6.3 Experimental MSE vs. SNR for Ex1 Component1 . . . . . 109
6.4 Experimental MSE vs. SNR for Ex1 Component2 . . . . . 109
6.5 Experimental MSE vs. SNR for Ex2 Component1 . . . . . 110
6.6 Experimental MSE vs. SNR for Ex3 Component2 . . . . . 110
6.7 Experimental MSE vs. computational cost for Ex1 at 8dB (Component 1) . . . . . 111
6.8 Experimental MSE vs. computational cost for Ex1 at 8dB (Component 2) . . . . . 111
6.9 Experimental MSE vs. computation cost for Ex1 at 14dB (Component 1 . . . . . 112
6.10 Experimental MSE vs. computational cost for Ex1 at 20dB (Component 1 . . . . . 112
6.11 Experimental MSE vs. computational cost for Ex2 at 8dB (Component 1 . . . . . 113
6.12 Experimental MSE vs. computational cost for Ex3 at 8dB (Component 2 . . . . . 113
6.13 Experimental MSE vs. SNR for Ex1 Component 1 (top) and Component 2 (bottom) . PO: phase only method, APA: alternating phase and amplitude method, APAS: APA with sparsity constraint using equitation (6.66) with $\lambda = 0.25$. . . . . 116

A.1 Projection onto epigraph set $C_f$ by successive projections onto supporting hyperplanes . . . . . 133

B.1 Time domain representation (top) and the Fourier transform (FFT) of the Example signal 1 where the frequency is normalized to sampling frequency. . . . . 135
B.2 Time domain representation (top) and the Fourier transform (FFT) of the Example signal 2 where the frequency is normalized to sampling frequency. . . . . 136
B.3 Time domain representation (top) and the Fourier transform (FFT) of the Example signal 3 where the frequency is normalized to sampling frequency. . . . . 136

B.4 Time domain representation (top) and the Fourier transform (FFT) of the Example signal 4 where the frequency is normalized to sampling frequency. . . . . 137
B.5 Time domain representation (top) and the Fourier transform (FFT) of the Example signal 5 where the frequency is normalized to sampling frequency. The signal is a segment from a Frequency Hopping MFSK signal. . . . . 137
B.6 Time domain representation (top) and the Fourier transform (FFT) of the Example signal 6 where the frequency is normalized to sampling frequency. The signal was taken from a dolphin's click-signal segment . . . . . 138

# List of Tables


3.1 Pearson correlation coefficient between TF distributions and the model TF for tested signals. A higher value shows better similarity to the model. . . . . 42
3.2 Rényi entropy of all the TF distributions for tested signals. A lower value indicates better localization. . . . . 42

4.1 Pearson correlation coefficient between TF distributions and the model TF for tested signals. A higher value shows better similarity to the model. . . . . 51
4.2 Rényi entropy of all the TF distributions for tested signals. A lower value indicates better localization. . . . . 51

5.1 Signal dependent kernel estimation steps and smoothing . . . . . 70
5.2 Signal dependent kernel estimation steps with a pre-filter . . . . 71
5.3 Pearson correlation coefficient between TF distributions and the model TF for tested signal examples. A higher value shows better similarity to the model. . . . . 81
5.4 Rényi entropy of TF distributions for tested signal examples. A lower value indicates better localization. . . . . 81

6.1 The alternating phase and amplitude (APA) algorithm. . . . . 92
6.2 Phase iterations for $J(\boldsymbol{b})$ using quasi-Newton (BFGS) algorithm . 97
6.3 Phase iterations for $f_{\boldsymbol{b}}(\boldsymbol{b})$ using quasi-Newton (BFGS) algorithm 97
6.4 Amplitude iterations via Minimization of $f_{\boldsymbol{a}}(\boldsymbol{a})$ using conjugate gradient (CG) algorithm . . . . . 98

6.5 Expectation Maximization (EM) iteration steps for multi-component signal parameter estimation . . . . . 101
6.6 Amplitude and phase orders for the components . . . . . 106

# Chapter 1

# Introduction

Signals with time-varying frequency content are encountered in many areas such as AM/FM communication [2], radar [3–7], sonar applications [8], medicine (EEG) [9,10], gravitational analysis [11,12], motor fault detection [13–16], speech, and audio [17,18] and classification of these type of signals. An important aspect of Time-Frequency (TF) analysis is the identification of separate components in a multi-component signal. High-resolution time-frequency (TF) representations and instantaneous frequency (IF)-based methods are needed for analysis, detection and classification of these type signals. Time-frequency distributions (TFDs) are two dimensional functions which designate the energy content of signals in the TF plane [19], [20]. TF signal representations enable separation of time-varying components overlapping both in time and frequency domains. It may not be possible to isolate some signal components in one domain using ordinary frequency domain filtering. The performance of a TFD is evaluated based on how good it represents the energy content of a signal in time-frequency plane without spurious terms.

The classical tool for TF analysis is the Wigner-Ville (WV) distribution [21], [22]. Smoothed versions of the WV distribution are grouped under the name of

Cohen Class of distributions [19]. The WV distribution is a quadratic TF representation, which provides a good time-frequency resolution especially for chirp-type signals. Because of its quadratic definition, the WV representation shows cross-terms together with actual components or auto-terms for multi-component signals [23]. Since the cross-terms result from cross correlation of different components they have an oscillatory shape on TF plane. Therefore in smoothed versions of WV they are attenuated or completely removed but this is achieved at the expense of highly reduced resolution. Because of this trade off between resolution and cross-term reduction, there have been many smoothing efforts, trying to reduce cross terms while maintaining a good TF resolution [24].

Compressive Sensing (CS) is a recently introduced concept which tries to recover a signal from limited number of random measurements with the assumption that the signal under consideration is sparse in some transform domain [25], [26], [27]. In CS problems sparsity assumption is imposed on the recovered signal by minimizing a cost function based on $l_0$ or $l_1$ norm. Frequency modulated (FM) signals used in radar signal processing can be considered to be sparse in TF plane. The problem of obtaining a high resolution and cross-term free TF distribution was studied in [28] using the CS perspective [29]. In this approach a sparse TF distribution is obtained using $l_1$ minimization among all TF distributions whose Fourier Transform coefficients equal to a given subset in the ambiguity domain. The cost function used in [28] consists of linearly combined two terms or an upper bound variance of the error. A proper choice of the mixture parameter (regularization parameter) is required to obtain a sparse solution. Regularization parameter selection is left as an open problem in [28].

In this thesis sparsity based methods are investigated for TF signal analysis. First, the Projection Onto Convex Sets (POCS) framework is used to solve the high resolution and cross-term free TF distribution estimation problem. A lifted domain POCS method is developed. The new algorithm is designed based on making orthogonal projections onto the epigraph set of the $l_1$ cost function. It successively imposes constraints on iterates in the TF and ambiguity function domains as in the well-known Papoulis-Gerchberg algorithm [30]- [30]. The new method does not require any regularization parameter adjustment as in [28],

[29]. It does not require any user specified bounds on the $l_1$ norm of the signal [31], either. Projection onto the epigraph set of $l_1$ cost function automatically determines the bound of the $l_1$ ball.

Results obtained with the proposed method are partially presented in [32]. In the subsequent and related chapters both the theoretical and the practical issues concerning the proposed POCS based TF distribution are presented.

Basic requirement for time-varying multi-component signals is to estimate the components from the noisy signal. There are two main approaches. In first approach the components are extracted separately and in this respect it is suboptimal. With the second approach all the components are estimated or extracted as a whole. Either with the first or the second approach, the components are estimated using TF techniques and/or parametric estimation, where the components are represented by a set of amplitude and phase parameters. The number of parameters is lower than the discrete signal length. The parameters are usually the polynomial coefficients. A mixed TF and parametric method is developed for component estimation. The method minimizes two cost functions, one for amplitude and one for phase, in an alternating manner. The cost function for the amplitude is convex and the phase cost function is non-convex. The method is shown to reduce the computation complexity substantially. Results are presented in [33].

In Chapter 2, preliminaries on Time-frequency representations are reviewed. Various cross term reduction methods are explained. Sparsity and compressive sensing concept is also reviewed in this chapter.

In Chapter 3, the TF distribution estimation problem is defined with a CS perspective. The chapter explains how a cross-term free TF distribution is estimated using the lifted projections method. The key to the method is the projection onto the epigraph set of the $l_1$ cost function. Two convex and closed sets are defined. The first one is the epigraph set of the $l_1$ cost function, the second set is subset of ambiguity domain coefficients located around origin. Based on an

initial smoothed TF, which is obtained from masked ambiguity function, successive projections onto convex sets are computed to obtain the final high resolution TF distribution. TF estimation examples are presented and results are compared with other methods in terms of localization and similarity measures.

In Chapter 4, alternative lifted POCS based TF reconstruction methods are developed based on different AF domain constraint sets. The selection of the type and size of this set have an effect on both resolution and similarity to the desired model TF. One example is the AF domain set obtained by real part of the coefficients in a given area. Another example is the AF domain set defined as the collection of all phase components and the DC magnitude term. In this way, optimum size selection problem in AF domain set is avoided.

Because of successive projections, the overall computation cost of the POCS method proposed in Chapter 3 is high compared to the Wigner-Ville distribution or its smoothed versions. In Chapter 5, a kernel method obtained using a projection onto the epigraph set of $l_1$-norm is developed to obtain a cross-term free high resolution TF distribution. This signal dependent kernel is estimated by a single projection onto the epigraph set of $l_1$ cost function and a subsequent scaling in the original AF domain. A localized cross-term-free TF representation is obtained by smoothing with the estimated kernel. The proposed method is comparable to the optimal kernel and achieves substantial saving in computations compared to the POCS method.

In many applications, separating TF components from each other which are sparsely distributed on TF plane is needed. Component estimation from a multicomponent time-varying signal is considered using mixed TF distribution and parametric maximum likelihood (ML) estimation framework in Chapter 6. The initial parameters are obtained via time-frequency techniques and final estimates are obtained by parametric ML estimation. A method, which iterates between amplitude and phase parameters separately, is proposed. The method reduces the computational complexity and convergence time significantly compared to other methods in the literature [34]. In Chapter 7, remarks and conclusions are presented.

# Chapter 2

# Time-frequency Representations and Sparsity-based Signal Reconstruction

## 2.1 Review of Time-frequency Representations

Time-frequency distributions (TFDs) are two dimensional functions which designate the energy content of signals in the TF plane [19], [20]. TF signal representations enable separation of time-varying components overlapping both in time and frequency domains.

Linear and the quadratic representations are the most widely used TF representations for signals with time-varying frequency content [35]. The windowed short-time Fourier Transform (STFT), which is a linear transformation, of a signal $x(t)$ is given by

$$STFT_x^h(t,f) = \int_{-\infty}^{+\infty} x(\tau)h(\tau - t)e^{-j2\pi f\tau}d\tau, \tag{2.1}$$

where $h(t)$ is the window or kernel function of the transformation. Besides linearity, STFT has some other nice features. In fact, STFT with Hermite-Gaussian

kernel is the only linear transformation [36, 37] which has shift-invariance and rotation-invariance in both time-frequency and all fractional Fourier domains [38, 39]. Linearity is a favored property in analysis, but the selection of the window length is the main challenge for STFT. While a long window provides good frequency resolution, it will reduce the time resolution or vice versa. There are efforts [20], [36,37] to adapt the window length to the signal so that signal dependency and better TF resolution is obtained. The representative of quadratic group is the Wigner-Ville distribution (WV) [21], [22] which is defined as follows:

$$W_x(t,f) = \int_{-\infty}^{+\infty} x(t+\tau/2)x^*(t-\tau/2)e^{-j2\pi f\tau}d\tau$$
$$= \int_{-\infty}^{+\infty} R_x(t,\tau)e^{-j2\pi f\tau}d\tau, \tag{2.2}$$

where $R_x(t,\tau)$ can be considered as the time-dependent autocorrelation function and Wigner distribution is defined as the Fourier transform of it. With this definition, Wigner distribution can interpreted as the time dependent spectrum or energy distribution among frequencies.

The discrete version of the Wigner distribution is given by

$$W_x[n,k] = 2\sum_{m=-N/2+1}^{m=N/2-1} x[n+m]x^*[n-m]e^{-j\frac{4*\pi}{N}km}, \tag{2.3}$$

where N is the length of the discrete signal.

The 2-D inverse Fourier transform (FT) of the WD is called the (symmetric) ambiguity function (AF), and it has found important application areas including time-frequency signal analysis and radar signal processing

$$A_x(\tau,\theta) = \int_{-\infty}^{+\infty} W_x(t,f)e^{j(\theta t+f\tau)}dtdf$$
$$= \int_{-\infty}^{+\infty} x(t+\tau/2)x^*(t-\tau/2)e^{-j\theta t}dt. \tag{2.4}$$

The ambiguity function is a 2-D correlation function which correlates the signal $x(t)$ by its time and frequency shifted versions. The parameter $\tau$ (time lag) and the Doppler parameter $\theta$ (or frequency lag) represent the time and frequency shifts, respectively.

In radar applications, the cross ambiguity function needs to be computed between a reference and a surveillance signal for target detection. But this requires a discrete computation for the AF function on a sufficiently finer grid. Therefore fast discrete computation is an important issue in radar target detection. A fast computation of AF or WV distribution is shown in [40].

Cohen's class of TF distributions [19] are generalized versions of the VW distributions:

$$P_x(t,f) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} A_x(\tau,\theta)\Phi(\tau,\theta)e^{-j2\pi\theta t - j2\pi f\tau}d\theta d\tau, \tag{2.5}$$

where $\Phi(\tau,\theta)$ is the kernel of Cohen's class TF distribution. WV distribution corresponds to $\Phi(\tau,\theta) = 1$. Multiplication of $A_x(\tau,\theta)$ by $\Phi(\tau,\theta)$ corresponds to 2-D convolution of $W_x(t,f)$ with 2-D Fourier transformed kernel function in the TF plane. This comes from the fact that the ambiguity function and WV are related to each other via the 2-D Fourier transformation (FT). Therefore $P_x(t,f)$ is a smoothed version of $W_x(t,f)$ given by

$$P_x(t,f) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} W_x(u,v)\Psi(t-u,f-v)dudv, \tag{2.6}$$

where $\Psi(u,v)$ is the 2D Fourier transform of the kernel function $\Phi(\tau,\theta)$.

Among many other nice properties of the WV distribution, the main appreciated feature is the high resolution. But high resolution is achieved at the expense of the so-called cross terms. Because of its quadratic definition, WV representation shows cross-terms together with actual components or auto-terms for multi-component signals [23]. Even with a mono component signal having a non-linear instantaneous frequency (IF) function, the WV distribution may have cross terms. The kernel function has the role of shaping ambiguity function $A_x(\tau,\theta)$. Cross terms are located between auto-terms and are oscillatory in nature. Therefore the smoothed version $P_x(t,f)$ of $W_x(t,f)$ will have cross terms attenuated or even removed depending on the kernel function $\Phi(\tau,\theta)$. One example to the smoothing is the pseudo WV distribution given by

$$PWD_x(t,f) = \int_{-\infty}^{+\infty} h(\tau)x(t+\tau/2)x^*(t-\tau/2)e^{-j2\pi f\tau}d\tau, \tag{2.7}$$

where $h(\tau)$ is the window function, which has smoothing effect in frequency domain only. The quantity $x(t+\tau/2)x^*(t-\tau/2)$ in (2.2) needs to be computed for $-\infty \leq \tau \leq \infty$. But this can be a problem in practice. Therefore pseudo WV distribution is a necessity in practice. Luckily, by selecting a proper length and shape for $h(\tau)$ this necessity can be turned to an advantage.

Spectrogram [35], which is the square magnitude of STFT,

$$S_x^h(t,f) = |STFT_x^h(t,f)|^2 \tag{2.8}$$

is another example of the smoothed WV distribution. In spectrogram analysis window length causes a trade off between time and frequency resolution. Smoothed Pseudo Wigner Distribution (SPWD) [35] is a solution to this trade off via selection of independent smoothing functions for time and frequency parameters. SPWD is given by

$$SPWD_x(t,f) = \int_{-\infty}^{+\infty} h(\tau) \int_{-\infty}^{+\infty} g(s-t)x(s+\tau/2)x^*(s-\tau/2)ds e^{-j2\pi f\tau} d\tau, \tag{2.9}$$

where $g(t)$ is the time smoothing widow and $h(t)$ is the frequency smoothing window in time domain. The separable smoothing function is expressed as $g(t)H(-f)$. By selecting $g(t) = \delta(t)$ we will obtain pseudo-WV distribution which has smoothing only along frequency axis.

Beside Cohen's class, given in (2.5), there are other generalizations of WV distribution which aim to achieve high resolution. One example is L-Class WV distribution proposed by Stankovic [41]. In this distribution, FT of $x^L(t+\tau/2L)x^{L*}(t-\tau/2L)$ is computed to get the distribution, where $L$ is an integer. $L=1$ corresponds to the WV distribution.

The WV distribution is ideally suitable to the chirp type signals which have linear frequency variation or second order polynomial phase functions. Polynomial Wigner-Ville Distribution (PWVD) [42], [43] is designed to localize higher order polynomial phase signals. But for multi-component signals it also suffers from cross-terms. In fact, in PWVD there exist non-oscillating cross terms, which cannot be removed by smoothing. Therefore other approaches are needed to remove them. In [42] LPWVD, which is a combination of L-Class WV (LWVD)

and PWVD is developed to solve this problem. In both PWVD and LPWD, the order of transformation needs to be arranged according to the polynomial order of the polynomial phase signal. It was shown that [43] the sixth-order PWVD achieves the delta function concentration for polynomial FM signals of up to the cubic order.

Among many other TF methods the reassigned spectrum [44], [45] is the one which achieves the best localization. With this method cross term-free high resolution TF distribution is obtained in two steps. In the first step, cross terms are removed by a proper smoothing method, such as spectrogram or smoothed pseudo WV method, in the second stage each time and frequency point on the TF plane is moved to a new location according to the center of gravity of neighboring region. In this way, the TF localization or resolution enhancement is obtained. An example is the reassignment of the spectrogram. The spectrogram is given by

$$S_x^h(t,f) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} W_x(s,u) W_h(s-t, u-f) ds du, \tag{2.10}$$

where $W_h(t,f)$ is the WV distribution of window function $h$. For reassignment during computation of spectrogram one also needs to compute,

$$\hat{t}_x(t,f) = \frac{1}{S_x^h(t,f)} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} s W_x(s,u) W_h(s-t, u-f) ds du, \tag{2.11}$$

and

$$\hat{f}_x(t,f) = \frac{1}{S_x^h(t,f)} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} u W_x(s,u) W_h(s-t, u-f) ds du \tag{2.12}$$

The spectrogram value is then moved from the point $(t,f)$ where it has been computed to this centroid $(\hat{t}_x(t,f), \hat{f}_x(t,f))$ leading to

$$\hat{S}_x^h(t,f) := \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} S_x^h(s,u) \delta(t - \hat{t}_x(s,u), f - \hat{f}_x(s,u)) ds du \tag{2.13}$$

The computation cost of reassigned spectrogram by (2.11) and (2.12) is very high. Because for each time and frequency point we need to compute a 2D convolution. Kodera [46, 47] has shown that reassigned time and frequency values

are actually related to the phase of STFT, which is ignored in definition of spectrogram. Therefore (2.11) and (2.12) can be obtained from

$$\hat{t}_x(t,f) = \frac{t}{2} - \frac{1}{2\pi}\frac{\partial\phi(t,f)}{\partial f}, \tag{2.14}$$

and

$$\hat{f}_x(t,f) = \frac{f}{2} + \frac{1}{2\pi}\frac{\partial\phi(t,f)}{\partial t} \tag{2.15}$$

where $\phi(t,f)$ is the phase of STFT. These quantities can be interpreted as the local instantaneous frequency (IF) and group delay (GD) of the analyzed signal, as filtered within the TF domain defined by the TF window $h$ centered in $(t,f)$. From this result it is seen that the reassignment method favors the energy concentrations in the vicinity of local IFs and GDs. Flandrin and Auger [48] have shown that $\hat{t}_x(t,f)$ and $\hat{f}_x(t,f)$ in (2.14) and (2.15) can be equivalently and efficiently computed by

$$\hat{t}_x(t,f) = \frac{t}{2} - Re\left\{\frac{STFT_x^{th}(t,f)}{STFT_x^{h}(t,f)}\right\}, \tag{2.16}$$

and

$$\hat{f}_x(t,f) = \frac{f}{2} + Im\left\{\frac{STFT_x^{dh}(t,f)}{STFT_x^{h}(t,f)}\right\} \tag{2.17}$$

where $Re\{.\}$ and $Im\{.\}$ stand for real and imaginary part respectively. $STFT_x^{th}$ is the STFT of signal $x$ computed by window $h_1(t) = th(t)$ and $STFT_x^{th}$ is the STFT of signal computed by window $h_2(t) = \partial h(t)/\partial t$. Compared to the standard spectrogram, its reassigned version can thus be computed with a moderate increase in the computational cost, since three STFTs are evaluated instead of one. With similar approach the reassignment spectrum for Smoothed Pseudo Wigner Distribution (SPWD) can also be computed efficiently [45]

Moving the value of a distribution to a new location away from where its computed increases the readability. On the other hand this may lead to an over localized TF distribution which may not be desired in all applications. For instance, the reassigned TF distribution of a sinusoidal signal at frequency $f_0$ approaches to an impulse in the TF plane around the frequency $f_0$ [48]. Therefore, the reassigned distribution tends to get away from a valid distribution and violates the uncertainty principle

Another drawback of this method is that relocation of energy at different TF points to the same location amplifies the amplitudes of stronger components in the TF plane much more than the weaker components. Therefore the reassignment method decreases the relative strength of the weaker components.

## 2.2 Compressive Sensing

Shannon/Nyquist sampling theorem [49] dictates that a band limited signal should be sampled at a rate or frequency larger than twice its highest frequency for a perfect reconstruction. The equidistant samples taken at this sampling frequency are then used to reconstruct the original continuous signal by low pass filtering (sinc interpolation). But unfortunately in many signal prepossessing applications there are so many samples that the storage or transmission of these samples makes compression a necessity. That is why after the signal is sampled it is usually transformed to some other domain (e.g., Fourier, wavelet, discrete cosine transform domains), in which it has a simple representation. This simple representation is obtained by getting rid of the negligibly small coefficients in the transform domain so that to have small number of coefficients compared to original signal samples. For example a signal which consists of two pure sinusoidal tones having frequencies of $f_1 = 100Hz$ and $f_2 = 300Hz$ will require a sampling rate of at least $2|f_1 - f_2|Hz = 400Hz$. This means, for one second of the signal segment, we need 400 samples. But we know that the same signal is represented in Fourier domain with just two impulses. In this respect two coefficients in Fourier domain are sufficient and the rest is redundant.

It would be nice to combine these two stages, that is sampling and transformation, so that to get the reduced coefficients directly. But there are two problems with this desired approach. First, not all the signals have a simple or sparse representation in Fourier domain. But they may have simple or sparse representation in some other domain. Therefore the domain in which the signal is sparse should be known in advance. Second, the positions of non-zero coefficients in transform domain are not known and depends on the signal content.

The compressive sensing (CS) [27, 50, 51] is the name given to the method which tries to provide a solution to this problem. The method uses a sufficient number of random linear measurements, which are far less than the number of samples dictated by Shannon/Nyquist sampling theorem, to reconstruct the original signal.

Given a finite-length, one-dimensional, discrete-time signal $x[n]$ $n = 0, 1, \cdots, N-1$ we can represent it with a vector $\boldsymbol{x} \in R^N$. Such a vector can be represented in any transform domain as,

$$\boldsymbol{x} = \sum_{i=1}^{N} s_i \boldsymbol{\psi}_i \tag{2.18}$$

or equivalently in vector form as

$$\boldsymbol{x} = \boldsymbol{\Psi} \boldsymbol{s}, \tag{2.19}$$

where $\boldsymbol{s} = [s_1 \ s_2, \ \cdots, \ s_N]^T$ is a vector containing the transform domain coefficients, $s_i$ obtained by

$$s_i = < \boldsymbol{x}, \psi_i > = \psi_i^T \boldsymbol{x} \quad i = 1, 2, \cdots, N, \tag{2.20}$$

$\psi_i$ is the $i^{th}$ basis vector and $\boldsymbol{\Psi}$ is the transformation matrix given by

$$\boldsymbol{\Psi} = [\psi_1 \ \psi_2 \ \cdots \ \psi_N], \tag{2.21}$$

A signal or vector $\boldsymbol{x}$ is named as $K - sparse$ if it is a linear combination of only $K$ basis vectors; that is, only $K$ of the $s_i$ coefficients in (2.18) are nonzero and $(N - K)$ are zero. This is very useful when $K << N$. In practice this is usually the case where the signal $\boldsymbol{x}$ has just a few large coefficients and with many small ones which can be ignored without causing an observable deformation from the original. In this respect such a signal is compressible.

In CS, $M$ linear measurements are taken from the signal $\boldsymbol{x}$. Then during the reconstruction process, these $M$ measurements are used together with the side information that the signal to be reconstructed is known to be $K - sparse$ in some domain. In a general measurement process with $M$ linear measurements, the inner product between $\boldsymbol{x}$ and a collection of vectors $\{\phi_j\}_{j=1}^{M}$ is computed as

$$y_i = < \boldsymbol{x}, \phi_j > \quad j = 1, 2, \cdots, M \tag{2.22}$$

Putting the measurements into $\boldsymbol{y} = [y_1\ y_2\ \cdots\ y_M]^T$ and using the measurement vectors as row of a matrix $\boldsymbol{\Phi}$ given by

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\phi}_1^T \\ \boldsymbol{\phi}_2^T \\ \cdots \\ \boldsymbol{\phi}_M^T \end{bmatrix}, \tag{2.23}$$

the measurement process can also be represented in matrix notation as

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x}, \tag{2.24}$$

Using the transform relation in (2.19) the measurement in (2.24) can also be expressed as

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{s} = \boldsymbol{\Theta}\boldsymbol{s}, \tag{2.25}$$

where $\boldsymbol{s} \in R^N$ is the transform domain coefficients, $\boldsymbol{y} \in R^M$ is the measurement vector and $\boldsymbol{\Theta} \in R^{M \times N}$ is the overall measurement matrix. The important thing here is that the measurement matrix $\boldsymbol{\Phi}$, is not signal dependent. It is fixed.

Therefore the overall problem consist of designing a stable measurement matrix $\boldsymbol{\Theta}$ and a reconstruction algorithm to reconstruct the original signal $\boldsymbol{x}$ from these $M$ measurements. Since $M < N$, the problem is ill-conditioned. But, the side information that the signal $\boldsymbol{x}$ is $K - sparse$ is used in the following problem definition to find a solution.

$$\begin{aligned} &\hat{\boldsymbol{s}} = \min \|\boldsymbol{s}\|_0 \\ &\text{subject to} \quad \boldsymbol{\Theta}\boldsymbol{s} = \boldsymbol{y} \end{aligned} \tag{2.26}$$

However this problem is a NP-complete optimization problem, and it is not easy to find the solution. If certain conditions such as Restricted Isometry Property (RIP) [49,51] hold for the measurement matrix $\boldsymbol{\Phi}$ , then the $l_0$ norm minimization problem (2.26) can be approximated by the following $l_1$ norm minimization

$$\begin{aligned} &\hat{\boldsymbol{s}} = \min \|\boldsymbol{s}\|_1 \\ &\text{subject to} \quad \boldsymbol{\Theta}\boldsymbol{s} = \boldsymbol{y} \end{aligned} \tag{2.27}$$

It is shown in [25, 50] that constructing matrix from random numbers, which are i.i.d Gaussian random variables, and choosing the number of measurements as

$cKlog(N/K) < MN$ satisfies the RIP conditions. Adding other constraints to the problem in (2.27) may allow even smaller number of measurements than this lower bound. Therefore the performance of the reconstruction is measured in terms of how $M$ is comparable to $K$.

There are various solution methods to problem in (2.27). The most famous one is the Basis Pursuit [52,53]. In many cases, the problem is converted to following unconstrained form

$$\hat{\boldsymbol{s}} = \min_{s} \|\boldsymbol{\Theta s} - \boldsymbol{y}\|_2^2 \; + \; \lambda \|\boldsymbol{s}\|_1 \tag{2.28}$$

or to the following constrained form

$$\begin{aligned} &\hat{\boldsymbol{s}} = \min \|\boldsymbol{s}\|_1 \\ &\text{subject to} \quad \|\boldsymbol{\Theta s} - \boldsymbol{y}\| < \epsilon \end{aligned} \tag{2.29}$$

and a solution is obtained with convex optimization techniques [54–57]. In this thesis the projection onto convex sets (POCS) is used to find the solution. With the proper definition of convex and closed sets the solution is obtained with alternating projections. Signal reconstruction from available data or information using POCS was used in many problems. One example is the resolution enhancement [58], another one is the reconstruction using Fractional Fourier transform domain samples [59].

Sparsity is a side information which is used in CS problem. This side information is used as a regularizer during optimization process. Most of the CS reconstruction algorithms in literature use the $l_p$ norm based regularization schemes where $p \in [0, 1]$. Other kinds of side information can also be used during reconstruction or optimization process. One example is the total variation (TV) function [60,61]. Knowing that the signal $x[n] \quad n = 0, 1, \cdots, N-1$ to be reconstructed has small TV value given by,

$$TV(\boldsymbol{x}) = \|\boldsymbol{x}\|_{TV} = \sum_{i=1}^{N-1} |x_i - x_{i-1}| \tag{2.30}$$

will allow the CS problem in (2.26) to be written as

$$\begin{aligned} &\hat{\boldsymbol{s}} = \min \|\boldsymbol{x}\|_{TV}, \\ &\text{subject to} \quad \boldsymbol{\Phi x} = \boldsymbol{y}, \end{aligned} \tag{2.31}$$

Notice that in TV based optimization problem the constraint related to the measurements is expressed as $\boldsymbol{\Phi x} = \boldsymbol{y}$ rather than as $\boldsymbol{\Theta s} = \boldsymbol{y}$. The TV norm is more appropriate for image processing applications [62, 63]. The reason why the TV norm is more appropriate for CS reconstruction of images is as follows. The transitions between the pixels of a natural image are smooth, therefore the underlying gradient of an image should be sparse. As the $l_p$ norm based regularization results in sparse signal reconstruction, the TV norm based regularization results in signals with sparse gradients. But one difficulty with TV norm is that the function is non-differentiable. Therefore the sub-gradient needs to be used where gradient is not defined.

Minimizing the total variation (TV) corresponds to a kind of smoothing or low-pass filtering in Fourier domain. Therefore a more generalized version of variation which is called as filtered variation (FV) [64] can also be used in (2.31). FV framework has some advantages over the TV framework [65]. If the user has prior knowledge about the frequency content of the signal, it becomes possible to design custom filters for that specific band. By defining different FV constraints for different bands, better reconstruction results can be obtained.

# Chapter 3

# Cross term-free Time-frequency Distribution Reconstruction via Lifted Projections

## 3.1 Sparsity-based Time-frequency Distribution Reconstruction

Most practical time-varying signals are in the form of weighted trajectories on the TF plane. In this respect, although they are neither sparse in time nor in the frequency domain, they are sparse in the joint TF plane. A multi-component [23] amplitude and frequency modulated (AM-FM) signal, which is expressed as follows:

$$x(t) = \sum_{k=1}^{L} a_k(t) e^{j\phi_k(t)} \tag{3.1}$$

is an example of signals, which are sparse in TF plane. In this expression $a_k(t)$ and $\phi_k(t)$ are the amplitude and phase functions of the $k^th$ signal component. The TF distribution of the $k^th$ component can be expressed as:

$$P_k(t,f) = a_k^2(t) \frac{1}{2\pi} \delta\left(f - \frac{d\phi_k(t)}{dt}\right) \tag{3.2}$$

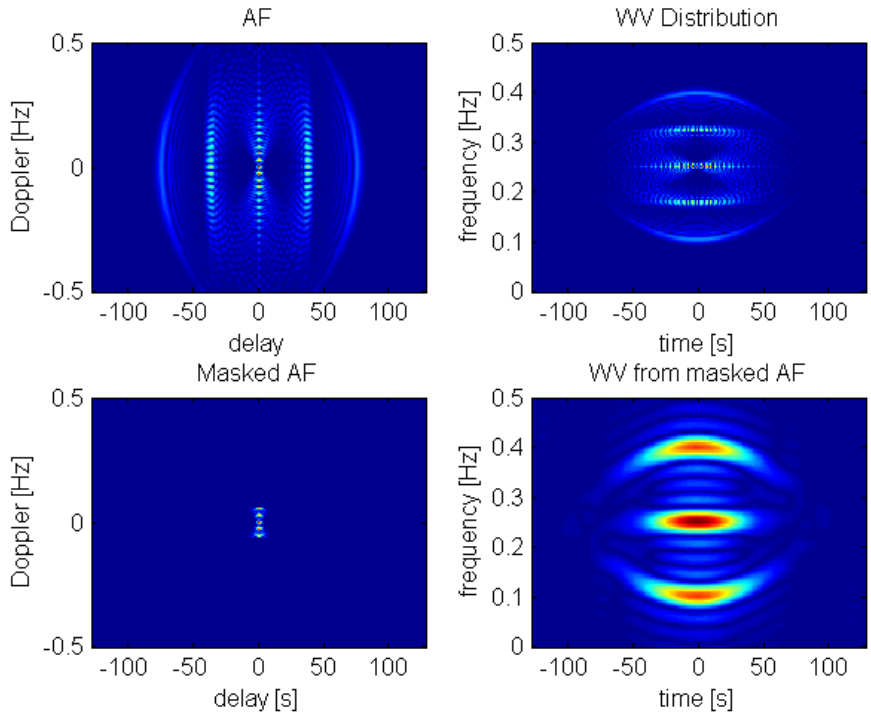


Figure 3.1: Effect of shaping the ambiguity function on the WV distribution: top-left: Ambiguity Function of the time-varying signal: top-right WV distribution: bottom-left: masked Ambiguity Function; bottom-right: WV distribution corresponding to the masked Ambiguity Function. The horizontal and the vertical axes show the time and the normalized frequency, respectively.

This is a trajectory on the TF plane with $\frac{d\phi_k(t)}{dt}$ being the instantaneous frequency (IF) function and $\delta(f)$ being the Dirac-delta function. Though not all the time-varying signals can be expressed in this form, most practical ones are sparse as in (3.2). In other words they are localized in a small area of the TF plane. The WV distribution is the 2-D Fourier transform (FT) of the AF and the values of AF around origin are due to auto-terms of a multi-component signal. Therefore masking the AF with a filter around origin and computing the 2D FT may reduce the cross-terms in WV distribution. But this approach also reduces the resolution as shown in Figure 3.1 The signal has three components or auto-terms in this example. However the WV distribution has five components (top-right). After masking the AF around the origin the three main components are clearly visible in Figure 3.1 (bottom right). Although the original WV distribution has high

resolution the three reconstructed components appear with a reduced resolution.

Due to uncertainty principle [66], [67], perfect localization can not be obtained in both TF and AF domains at the same time. Therefore, there is a trade off between the WF domain resolution and cross-terms. In order to reduce the cross-terms of the TF distribution as much as possible a set of optimization problems are proposed by Flandrin and Borgnat [28] as follows:

$$\begin{aligned} &\boldsymbol{P}^* = \text{argmin}_{\boldsymbol{P}} \, \|\boldsymbol{P}\|_1 \\ &\text{subject to} \quad \mathcal{F}^{-1}\{\boldsymbol{P}\} = \boldsymbol{A}_x[k,l] \qquad k,l \in \Omega \end{aligned} \tag{3.3}$$

where $\boldsymbol{P}$ and $\boldsymbol{A}_x$ are matrices of size $N \times N$ obtained by discretization $P(t,f)$ and $A_x(\tau,\theta)$, respectively, and $N$ is the length of the discrete-time time-varying signal $x$. The $l_1 - norm$ is defined as $\|\boldsymbol{P}\|_1 = \sum_{i=1}^{N}\sum_{j=1}^{N}|P_{ij}|$. The set $\Omega$ defines the filter mask around the origin in AF domain, and $k$ and $l$ are the discrete indexes corresponding to delay and Doppler parameters, respectively. It is established in CS theory that minimization of $l_1 - norm$ of $P$ provides sparsity in the WV domain [28].

The second optimization problem is a relaxed version of (3.3) :

$$\begin{aligned} &\boldsymbol{P}^* = \text{argmin}_{\boldsymbol{P}} \, \|\boldsymbol{P}\|_1 \\ &\text{subject to} \quad \|\mathcal{F}^{-1}\{\boldsymbol{P}\} - \boldsymbol{A}_x[k,l]\|_2^2 \leq \epsilon \qquad k,l \in \Omega \end{aligned} \tag{3.4}$$

where the parameter $\epsilon$ is a user defined upper-bound on the error variance between the inverse Fourier transform of the WV distribution $\boldsymbol{P}$ and the ambiguity function $\boldsymbol{A}_x$ over the filter mask $\Omega$. Obviously, the problem (3.4) is equivalent to (3.3) when $\epsilon = 0$.

The third problem is a regularized optimization problem

$$\boldsymbol{P}^* = \underset{\boldsymbol{P}}{\text{argmin}} \quad \lambda\|\boldsymbol{P}\|_1 + \frac{1}{2}\|\mathcal{F}^{-1}\{\boldsymbol{P}\} - \boldsymbol{A}_x[k,l]\|_2^2 \qquad k,l \in \Omega, \tag{3.5}$$

where the regularization parameter $\lambda$ is also a user-defined parameter adjusting the trade-off between the $l_1 - norm$ minimization and the error between the actual and estimated ambiguity functions. A large $\lambda$ value corresponds to a sparse WV distribution in the TF plane but this may correspond to a large deviation from

the actual ambiguity function. It is shown that optimization problems (3.4) and (3.5) are actually equivalent to each other [28], [68]. It is always possible to find a $\lambda$ value corresponding to each $\epsilon$ value.

In Figure 3.2, a reconstructed solution obtained by minimizing (3.3) is shown. The signal is the same as the signal in Figure 3.1. A circular mask $\Phi_\Omega$ with radius $r_0 = N/16$ around the origin is applied to the AF as in Figure 3.1. The circular mask is given by

$$\Phi_\Omega[k,l] = \begin{cases} 1 & \sqrt{k^2+l^2} \leq r_0 \\ 0 & else \end{cases} . \tag{3.6}$$

The TF distribution in Figure 3.2 was obtained using the $l_1$-MAGIC TOOLBOX [29]. The 3D plot of the WV distribution is shown in the figure at bottom. The solution has high resolution and cross terms are removed but the reconstructed solution is too sparse to be called as a TF distribution as stated in [28]. This is because the estimated distribution is not smooth at all. It is discontinuous and spiky.

Instead of solving the minimization problem (3.3) which has a strict constraint in the AF domain, the minimization problem (3.4) with relaxed constraints, are solved in [28] to obtain an acceptable result. In this modified problem, the parameter $\epsilon > 0$ needs to be properly defined in advance. Therefore, the choice of the regularization parameter $\lambda$ (or equivalently the upper bound $\epsilon > 0$) is left as an open problem in [28].

Among many other TF representations the reassigned spectrum [44, 45] results in a good TF localization. The TF, in Figure 3.3 was obtained with Reassigned Smoothed Pseudo WV (RSPWV) using the Time-Frequency Toolbox [1]. The 3D plot of the result is also shown at the bottom.

The reassignment spectrum produces a good localization around the IF law, as shown in Figure 3.3. This is similar to the result obtained by $l_1$-MAGIC TOOLBOX, but it has a spiky nature. In this respect it deviates from the physical meaning of the signal being analyzed. However, it is still the best method in terms of TF localization.

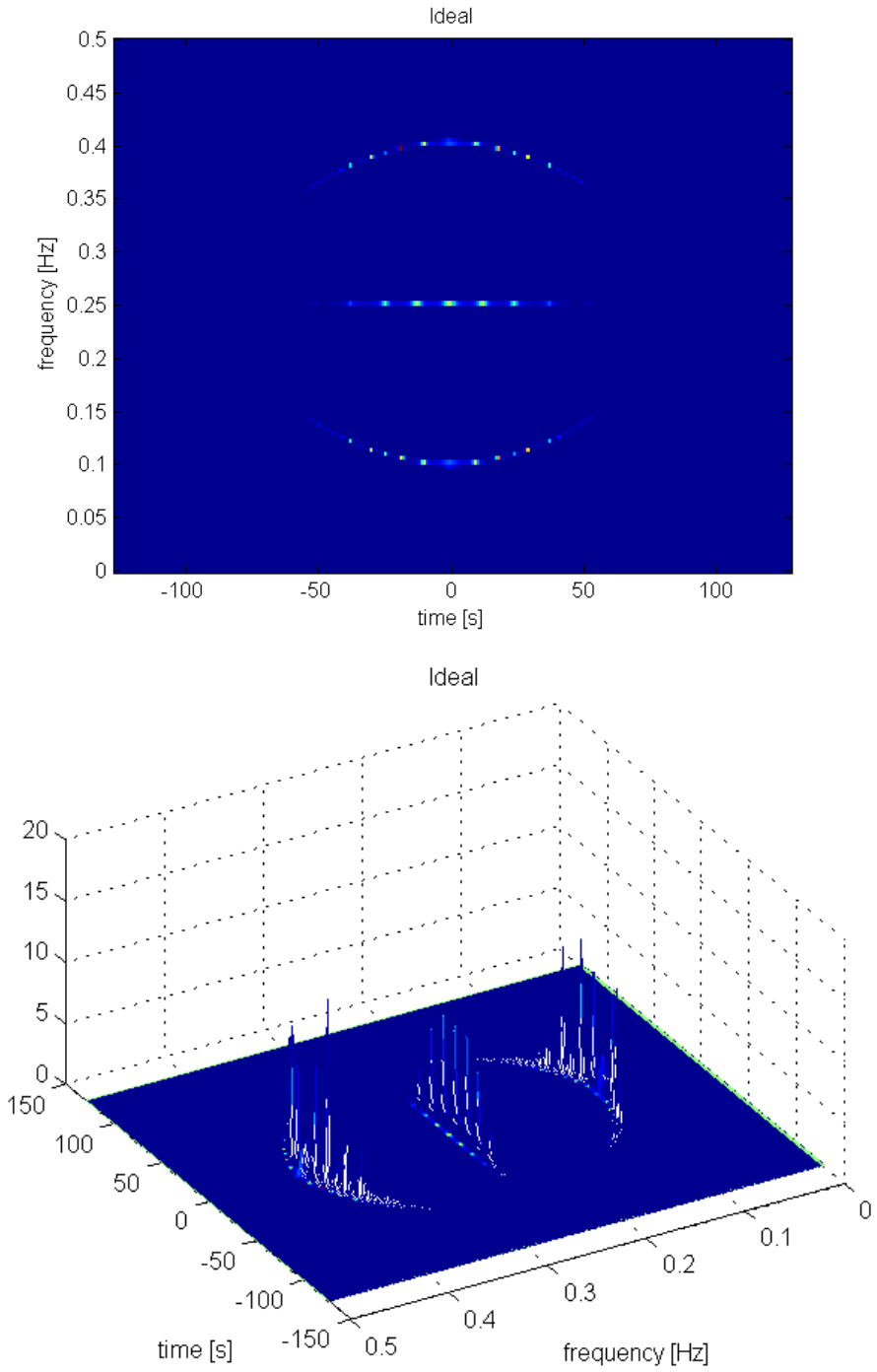


Figure 3.2: TF distribution obtained by the minimization of (3.3) using $l_1$-MAGIC TOOLBOX (top) and its 3D plot. The frequency is normalized according to the sampling frequency.

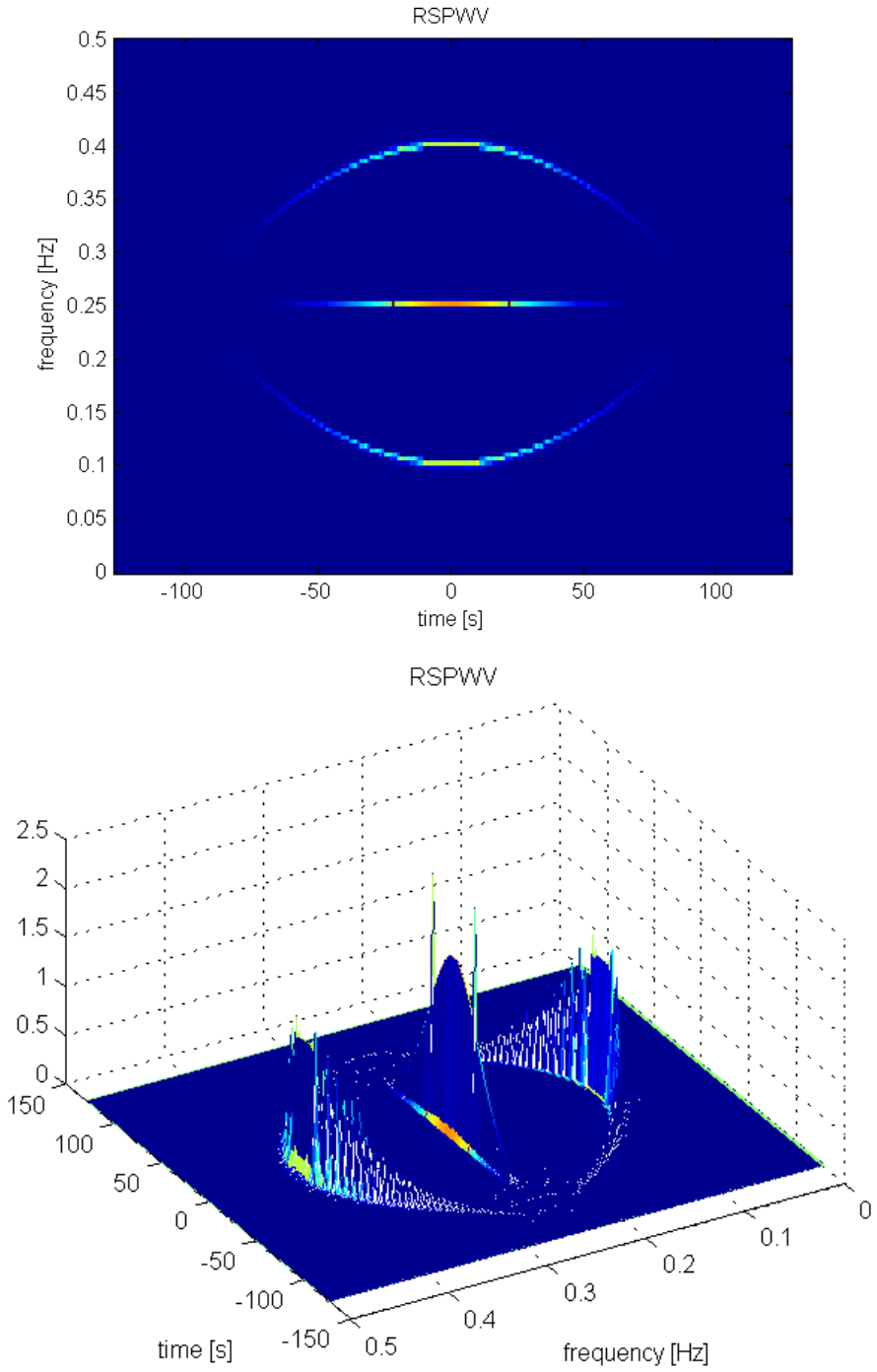


Figure 3.3: Reassigned Smoothed Pseudo WV (RSPWV) distribution and its 3D plot obtained by using the Time-Frequency-Toolbox [1].

We use a lifted Projection Onto Convex Sets POCS method [69, 70], which does not require any regularization parameter or upper bound on the $l_1$-norm of the signal to estimate the TF distribution.

The algorithm is iterative;it iterates back and forth between the Fourier and AF domains. In the AF domain, the masking filter is applied on the current iterate. In the TF domain, an orthogonal projection onto the epigraph set of $l_1$-norm is performed.

## 3.2 Time-frequency Distribution Reconstruction with Lifted POCS

Bregmans (POCS) framework [30,71–73] was successfully applied to many inverse and design problems in signal [58, 59] and image processing [68, 74, 75]. POCS is an iterative signal reconstruction method in which the goal is to find a solution satisfying all the constraint of a given problem in a Hilbert space framework. The solution vector should be in the intersection of all the constraint sets corresponding to the constraints. If the constraint sets happen to be closed and convex the algorithm globally converges regardless of the initial vector. In each step of the algorithm an orthogonal projection onto one of the convex sets is performed. Bregman showed that iterates converge to a vector in the intersection of all the convex sets provided that the intersection of the constraint sets is non-empty. If the sets do not intersect iterations oscillate between members of the sets [76, 77]. This process is graphically illustrated in Figure 3.4 for both intersecting and non intersecting cases. Both $\boldsymbol{x}^*$, and $\boldsymbol{x}_1^*$ and $\boldsymbol{x}_2^*$ are accepted as solutions in inverse problems.

POCS based solution is proposed here based on the cost function ($l_1$-norm) of the TF reconstruction problem defined in (3.4)

$$f(\boldsymbol{P}) = \|\boldsymbol{P}\|_1 = \sum_{i=1}^{N}\sum_{j=1}^{N}|P_{ij}| \tag{3.7}$$

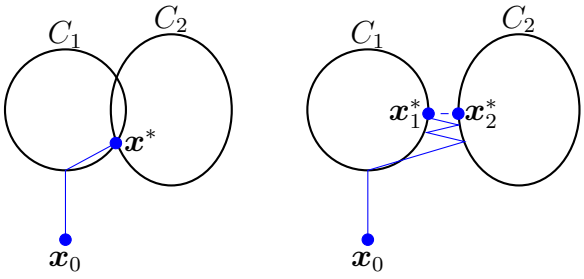


Figure 3.4: Left: POCS iterates converge to a vector in the intersection of convex sets $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$. The vector $\boldsymbol{x}_0$ is the initial vector and $\boldsymbol{x}^*$ is in the intersection of sets $\boldsymbol{C}_1$and $\boldsymbol{C}_2$. Right: Iterates oscillate between the two vectors when the intersection is empty. Vectors $\boldsymbol{x}_1^*$ and $\boldsymbol{x}_2^*$ minimize the distance between the sets $\boldsymbol{C}_1$and $\boldsymbol{C}_2$.

Since the TF distribution $\boldsymbol{P}$ has $N^2$ entries it can be converted to a vector in $\boldsymbol{R}^{N^2}$.

The POCS method, depending on available side information, can be applied to sparsity based reconstruction problem in several ways. One basic approach is to use two convex sets, defined in the following way:

$$C_1 = \{vec(\boldsymbol{P}) \in \boldsymbol{R}^{N^2} \quad | \quad f(\boldsymbol{P}) = \|\boldsymbol{P}\|_1 \leq \epsilon_1\} \tag{3.8}$$

where $\epsilon_1$ is the bound on the $l_1$ norm of the TF distribution to be reconstructed. The set $C_2$ is the set of measurements or the ambiguity domain constraint:

$$C_2 = \{vec(\boldsymbol{P}) \in \boldsymbol{R}^{N^2} \quad | \quad \mathcal{F}^{-1}\{\boldsymbol{P}\} = \boldsymbol{A}_x[k,l] \quad k,l \in \Omega,\} \tag{3.9}$$

The reconstruction with this method, if the number of measurements is below a threshold, produces noisy results [31]. Because the intersection of the sets $C_1$ and $C_2$ does not contain a single point, due to insufficient number of measurements. In this respect the solution will be approximate or close to actual. The error between the noisy solution and the actual will be distributed among all vector entries. This is due to $l_2$ minimization during orthogonal projection operation onto each set. Therefore, in limited measurement case, together with the sets (3.8) and (3.9) for a smooth reconstruction, a third set can be defined in the following way,

$$C_3 = \{vec(\boldsymbol{P}) \in \boldsymbol{R}^{N^2} \quad | \quad \|\boldsymbol{P}\|_{TV} \leq \epsilon_{TV},\} \tag{3.10}$$

where together with $\epsilon_1$, $\epsilon_{TV}$ also needs to be defined. Using these three sets, the POCS method will converge to a solution, if they intersect. Alternatively, the problem can be defined in the following way [31],

$$\begin{aligned} \boldsymbol{P}^* &= \text{argmin}_{\boldsymbol{P}} \|\boldsymbol{P}\|_{TV} \\ \text{subject to} &\left\{ \begin{array}{ll} f(\boldsymbol{P}) = \|\boldsymbol{P}\|_1 \leq \epsilon_1 & \\ \mathcal{F}^{-1}\{\boldsymbol{P}\} = \boldsymbol{A}_x[k,l] & k,l \in \Omega \end{array} \right\} \end{aligned} \tag{3.11}$$

where we only need to know the bound $\epsilon_1$.

In CS type problems, the side information determines the required minimum number of measurements for a successful reconstruction. In POCS method defined with the sets (3.8), (3.9) and (3.10) three types of side information is used to reconstruct signal. These are sparsity, bound on $l_1$ norm and bound on TV norm. In some problems where it is known that the signal energy is distributed among specific regions or bands of transform domain (example, wavelet domain), further reduction in number of measurements can be obtained with additional assumption or bounds on $l_1$ norm of specific bands [31, 64].

There is a difference between original CS problem and the sparsity based TF reconstruction problem. In CS problem, the aim is to reconstruct the signal from a small number of linear measurements with sparsity assumption or other side information available. But in sparsity based TF reconstruction problem, actually we have all the measurements, namely the AF domain coefficients. But, rather than random, we intentionally select a specific set of those coefficients which allow cross-term free reconstruction. Otherwise, the TF will be reconstructed including the cross terms. On the other hand, we do not have any side information, except for sparsity, namely the bounds on $l_1$ norm or any other regularization constraint.

With only the information at hand and appropriate convex set definitions, a Lifted POCS method is developed to solve TF reconstruction problem. In lifted POCS approach we increase the dimension of the vectors by one. In $\boldsymbol{R}^{N^2+1}$ any vector on the graph of $l_1$-norm can be represented as follows

$$\boldsymbol{w} = [vec(\boldsymbol{P})^T \quad f(\boldsymbol{P})]^T, \tag{3.12}$$

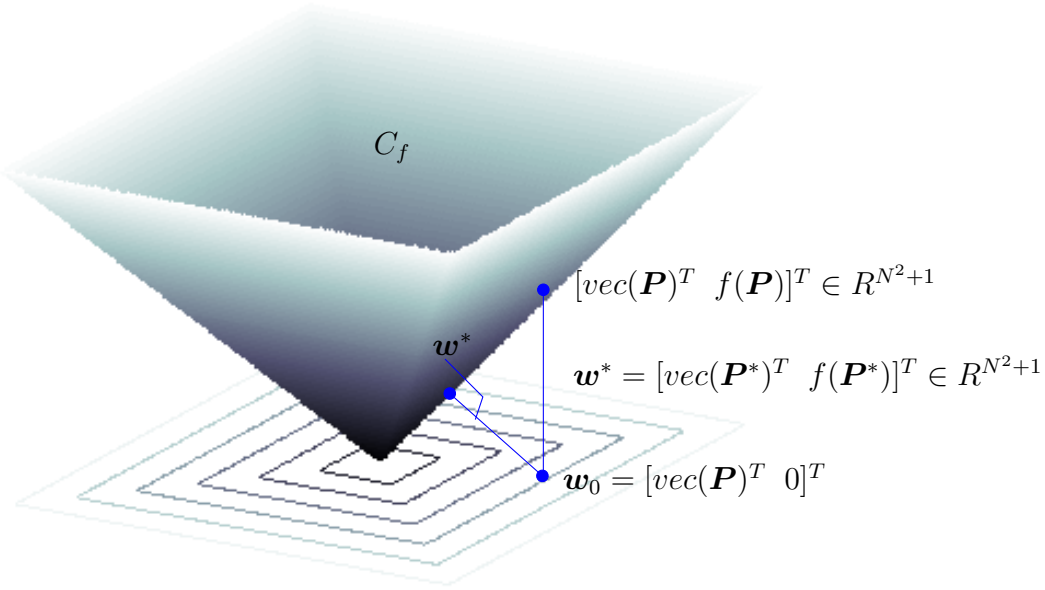


Figure 3.5: Graphical representation of the epigraph set of $l_1$ norm $C_f$ and the projection of the vector $\boldsymbol{P}_0$ onto the set $C_f$

where $vec(\boldsymbol{P}) \in \boldsymbol{R}^{N^2}$ is the vector form of the TF distribution matrix $\boldsymbol{P}$ and the last entry represents the $l_1$-norm of the TF distribution $\boldsymbol{P}$. For the TF reconstruction problem, the epigraph set $C_f$ of the $l_1$-norm is defined as follows:

$$C_f = \{\boldsymbol{w} = [vec(\boldsymbol{P})^T \quad v]^T \in \boldsymbol{R}^{N^2+1} \quad | \quad f(\boldsymbol{P}) = \|\boldsymbol{P}\|_1 \leq v\} \tag{3.13}$$

where $\boldsymbol{w}$ is an arbitrary vector in lifted domain $\boldsymbol{R}^{N^2+1}$ , and $v$ is the last element of the vector $\boldsymbol{w}$. The epigraph set $C_f$ contains all the vectors above the graph of $l_1$-norm in the lifted domain $R^{N^2} + 1$ [73]. The epigraph set of a function is graphically illustrated in Figure 3.5. Since the $l_1$-norm is a convex function the epigraph set is a convex set in the vector space $\boldsymbol{R}^{N^2+1}$. The set $C_f$ represents our TF domain constraint on the solution of the TF distribution estimation problem.

The second convex set is simply based on the AF domain information expressed in lifted domain. It is the set of TF distributions whose 2D inverse FT is equal to the $\boldsymbol{A}_x[k,l]$ on the filter mask $\Omega$. It is defined as follows.

$$C_{AF} = \{\boldsymbol{w} = [vec(\boldsymbol{P})^T \; v]^T \in \boldsymbol{R}^{N^2+1} \quad | \quad \mathcal{F}^{-1}\{\boldsymbol{P}\} = \boldsymbol{A}_x[k,l] \quad k,l \in \Omega\} \tag{3.14}$$

It can be shown that $C_{AF}$ is also a closed and convex set.

Both the sets $C_f$ and $C_{AF}$ are defined in lifted domain $\boldsymbol{R}^{N^2+1}$. It may not

be possible to know, if the sets $C_f$ and $C_{AF}$ intersect or not a priori. This depends on the values of the ambiguity function. But we can easily understand if they intersect or not during the implementation of the POCS algorithm. If the iterates converge to a single solution they intersect. If they oscillate between the two solutions then it means that the sets $C_f$ and $C_{AF}$ do not intersect.

## 3.3 Projection onto the sets $C_f$ and $C_{AF}$ and the iterative algorithm

Next, the orthogonal projection operations onto the sets $C_f$ and $C_{AF}$ will be described.

Given an initial TF distribution $\boldsymbol{P}_0$ we construct a corresponding vector in $\boldsymbol{R}^{N^2+1}$ by padding a zero at the very end as follows: $\boldsymbol{w}_0 = [vec(\boldsymbol{P}_0)^T \quad 0]^T \in \boldsymbol{R}^{N^2+1}]$ , whose orthogonal projection $\boldsymbol{w}_1$ onto $C_f$ is defined as follows:

$$\boldsymbol{w}_1 = \min_{\boldsymbol{w} \in C_f} \|\boldsymbol{w} - \boldsymbol{w}_0\|_2^2 \tag{3.15}$$

The vector $\boldsymbol{w}_1$ is the vector closest vector in $C_f$ to $\boldsymbol{w}_0$. The solution TF distribution matrix $\boldsymbol{P}_1$ is obtained from the first $N \times N$ entries of $\boldsymbol{w} = [vec(\boldsymbol{P}_1)^T \quad f(\boldsymbol{P}_1)]^T$. The last entry of $\boldsymbol{w}_1$ is $f(\boldsymbol{P}_1)$ because the projection should be on the boundary of the convex set which is the "graph" $f(\boldsymbol{P}_1)$. If $\boldsymbol{w}_0$ is inside the $C_f$ its projection is itself by definition. The projection of $\boldsymbol{w}_0$ onto epigraph set $C_f$ can be also defined as follows:

$$\boldsymbol{w}_1 = [vec(\boldsymbol{P}_1)^T \quad f(\boldsymbol{P}_1)]^T = \quad \min_{\boldsymbol{P}} \|vec(\boldsymbol{P}) - vec(\boldsymbol{P}_0)\|_2^2 + f^2(\boldsymbol{P}) \tag{3.16}$$

where the first term is obtained from the first $N^2$ entries and the second term is obtained from the last entries of $\boldsymbol{w}$ and $\boldsymbol{w}_0$, respectively. Notice that square of the $l_1$-norm $f^2(\boldsymbol{P})$ is different from the $l_2$-norm. The solution of the minimization problem (3.16) is explained in Appendix A.1.

Next, the vector $\boldsymbol{w}_1$ is projected onto $C_{AF}$ producing the next iterate $\boldsymbol{w}_2$ . The corresponding TF matrix $\boldsymbol{P}_2$ satisfies $\mathcal{F}^{-1}\{\boldsymbol{P}_2\} = \boldsymbol{A}_x[k,l] \qquad k,l \in \Omega$. This

projection corresponds to the AF domain constraint. It is implemented very easily using the 2D inverse Fourier Transform. The ambiguity function corresponding to $\boldsymbol{P}_1$ is computed as follows:

$$\boldsymbol{A}_1 = \mathcal{F}^{-1}\{\boldsymbol{P}_1\} \tag{3.17}$$

The ambiguity function $\boldsymbol{A}_2$ is defined using the actual $\boldsymbol{A}_x$ values in the mask $\Omega$:

$$\boldsymbol{A}_2[k,l] = \boldsymbol{A}_x[k,l] \qquad k,l \in \Omega \tag{3.18}$$

the remaining entries of $\boldsymbol{A}_2$ come from $\boldsymbol{A}_1$:

$$\boldsymbol{A}_2[k,l] = \boldsymbol{A}_1[k,l] \qquad k,l \notin \Omega \tag{3.19}$$

Next $\boldsymbol{P}_2$ is obtained by computing 2D FT of $\boldsymbol{A}_2$.

In the second round of POCS iterations $\boldsymbol{P}_2$ or equivalently $\boldsymbol{w}_2 = [vect(\boldsymbol{P}_2)^T \quad f(\boldsymbol{P}_1)]^T$ is constructed where first $N$ entries are taken from $\boldsymbol{P}_2$ and $(N+1)^{th}$ entry is taken from previous projection onto $C_f$ because we have not changed it during projection onto $C_{AF}$. Then $\boldsymbol{w}_2$ is projected back onto $C_f$ to obtain $\boldsymbol{P}_3$. After this projection operation the constraint (3.18) is probably no longer valid. Therefore $\boldsymbol{P}_3$ is projected back onto $C_{AF}$ to obtain $\boldsymbol{P}_4$ and so on. The lifted POCS iterations continue in this manner. The pseudo-code for the lifted POCS algorithm is shown in Algorithm 1. Assuming that the intersection of $C_f$ and $C_{AF}$ is non-empty, the iterations will converge to a point in intersection set. Or, they oscillate between $C_f$ and $C_{AF}$ as shown in Figure 3.4. Both cases are fine with us because we look for a compromise solution for the TF distribution.

**Algorithm 1** The pseudo-code for Lifted POCS with real AF coefficients

**function** $\boldsymbol{P}$=LPOCSR($\boldsymbol{x}$)
$N = length(\boldsymbol{x})$; ;$\Omega = circ(N/16)$
$\boldsymbol{A}_x$:=AF($\boldsymbol{x}$) $\boldsymbol{A}_0$:=mask($\boldsymbol{A}_x$,$\Omega$)
$\boldsymbol{P}_0 := \mathcal{F}\{\boldsymbol{A}_0\}$ ; $\boldsymbol{w}_0 := [vec(\boldsymbol{P}_0)^T \quad 0]^T$
$i = 1$ ; $\epsilon = 10^{-5}$
**while** $err \geq \epsilon$ **do**
$\boldsymbol{w}_i = \min_{\boldsymbol{w} \in C_f} \|\boldsymbol{w} - \boldsymbol{w}_{i-1}\|_2^2 = [vec(\boldsymbol{P}_i)^T \quad w_{i,n+1}]^T$
$\boldsymbol{A}_i = \mathcal{F}^{-1}\{\boldsymbol{P}_i\}$
$\boldsymbol{A}_i|_\Omega := \boldsymbol{A}_x|_\Omega$
$\boldsymbol{P}_i := \mathcal{F}\{\boldsymbol{A}_i\}$
$\boldsymbol{w}_i := [vec(\boldsymbol{P}_i)^T \quad w_{i,n+1}]^T$
$err = \|vec(\boldsymbol{P}_i) - vec(\boldsymbol{P}_{i-1})\|_2 / \|vec(\boldsymbol{P}_{i-1})\|$
$i = i + 1$
**end while**
**end function**

The method proposed here also provides globally convergent solutions for other convex cost functions such as total-variation (TV) [60], filtered variation (FV) [64], $l_1$, and entropic function which are widely used in signal and image processing problems because all convex cost functions can be represented as closed and convex sets in a lifted vector space.

## 3.4 Experimental Results

In order to test the effectiveness of the lifted POCS method introduced in Section 3.2, TF distributions for several example signals are estimated. The time and Fourier domain representation of the example signals are given in Appendix B. The examples used in [28] are also used here. Reconstruction results are shown in Figures 3.6 - 3.13. In all the examples the set $\Omega$ is chosen as a circular mask around the origin in the ambiguity domain as given in (3.6). The radius of the

mask is selected as $r_0 = N/16$, where $N$ is the length of the discrete-time signal as in [28]. The results obtained using Wigner-Ville, Spectrogram, Smoothed Pseudo Wigner-Ville (SPWV), Reassigned SPWV (RSPWV) [1] and TF reconstruction using the masked AF and $l_1$-MAGIC TOOLBOX (interior point methods) as in [28] are shown in Figures 3.6 - 3.13, respectively. For the purpose of comparison, the desired ideal TF model of the signals is also included in Figures. TF model is nothing but the TF constructed from IF law of the signal components scaled by their power as in (3.2). Though not all examples are polynomial phase signals and have time-varying amplitude, the LPWVD [42] with order 6 was also used to obtain related TF distribution. The related MATLAB code was obtained from Y. Wang [42]. In order to get, on the average, good results, for all the example signals at hand, the SPWV time smoothing window length was set to an odd integer closest to $N/10$, and the length of frequency smoothing filter at time domain was set to an odd integer closest to $N/4$ , with $N$ being signal length. In this way any parameter adaptation to the signal was avoided.

The convergence of the Lifted POCS method is monitored with the help of normalized error defined by,

$$err = \frac{\|vect(\boldsymbol{P}_i) - vect(\boldsymbol{P}_{i-1})\|_2}{\|vect(\boldsymbol{P}_{i-1})\|_2} \tag{3.20}$$

The $l_1$-norm of the TF distribution versus the number of iterations is shown in Figure 3.14 for the example Signal 1.

Reconstruction results in Figures 3.6 - 3.13 show that the solutions obtained with $l_1$-MAGIC TOOLBOX is too sparse. As pointed out by Borgnat and Flandrin [28] they cannot be accepted as a TF representation of the signal. The reassigned spectrum RSPWV has a good localization and better smoothness than the results with $l_1$-MAGIC TOOLBOX but it also has a spiky nature as shown in Figure 3.3,3.9,3.11 and Figure 3.13. On the other hand, from the same figures it is observed that the lifted POCS method generates better and acceptable results without adjusting any parameters during the optimization process. In this respect the LPOCS method provides a good compromise between localization and smoothness which is a physical property of the original signal. Both SPWV and LPOCS have good resolution and smoothness based on visual comparison.

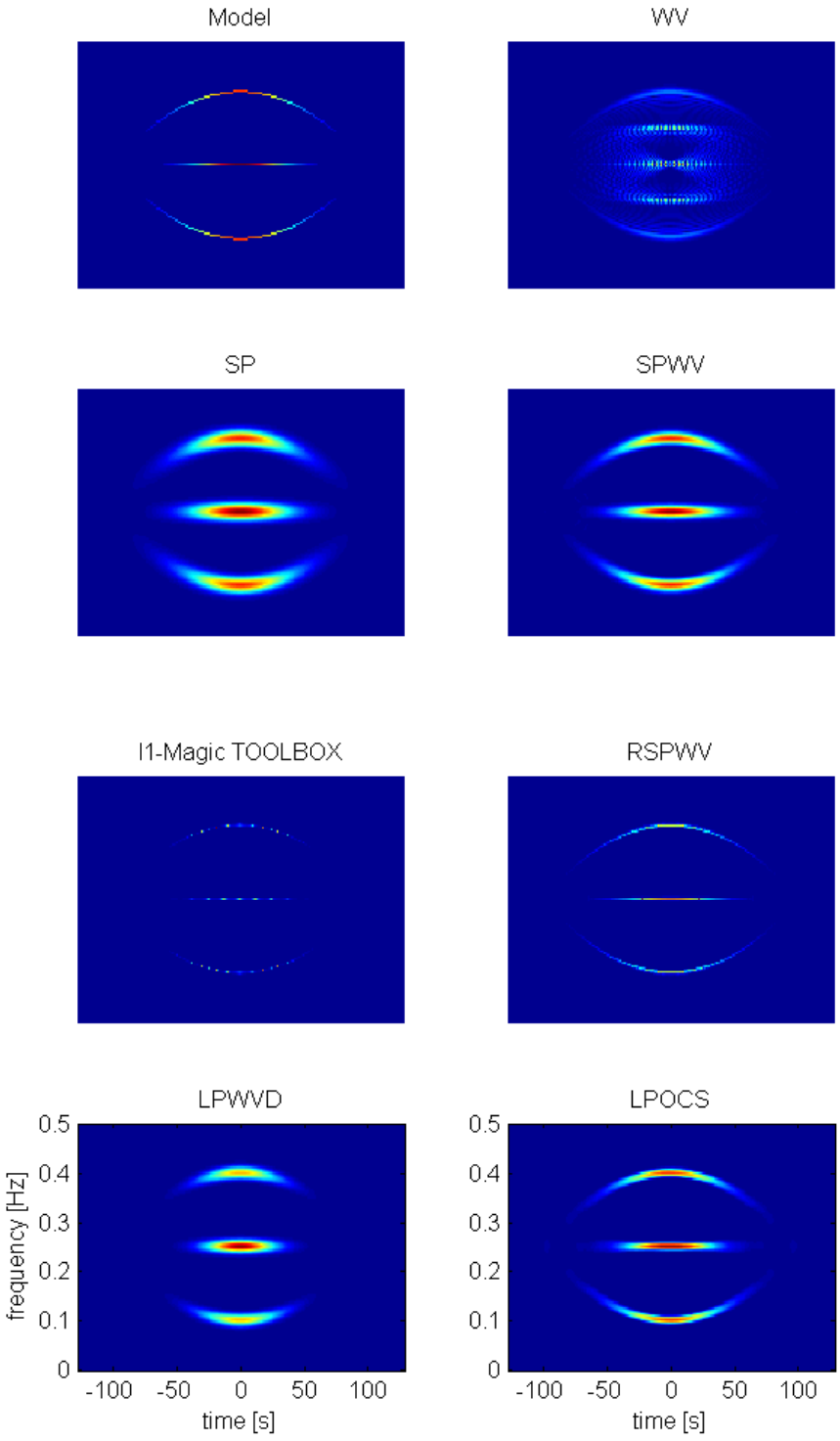


Figure 3.6: Signal 1: the TF reconstruction using, left column: the ideal model, Spectrogram, $l_1$-MAGIC TOOLBOX, L-Class Polynomial WV distribution (LP-WVD), right column: WV, the Smoothed Pseudo WV (SPWV), Reassigned SPWV (RSPWV), lifted POCS.

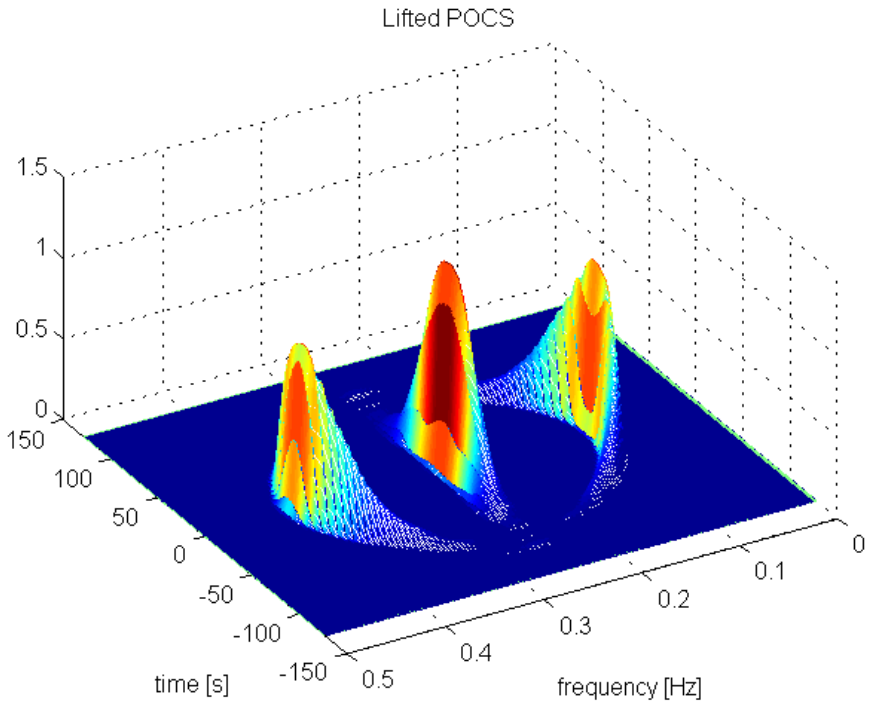


Figure 3.7: Signal 1: 3D plot of the TF distribution obtained by lifted POCS.

But the resolution of LPOCS is better than SPWV. SPWV additionally requires the time and frequency window lengths to be adapted to the signal for good resolution. In Figure 3.15, a reconstructed TF example obtained from a noisy signal is shown. The time-varying signal in Figure 3.8 was corrupted by additive zero-mean white Gaussian noise. The SNR value is 10dB. Signal auto-terms are clearly reconstructed and cross-terms are suppressed by the lifted POCS method in Figure 3.15 (bottom right). The result is comparable to the reassigned spectrum.

In Figure 3.16 a signal example from Frequency Hopping /M-ary Frequency-Shift-Keyed (FH/MFSK) communication is shown. It is shown that [78] using a cross term-free TF representation the parameters of FH/MFSK, which include hopping frequencies, hopping rate, hopping sequence and modulation type, can be estimated without making any assumption about the alphabet of hopping frequencies or the synchronization. It is observed that the LPOCS method clearly

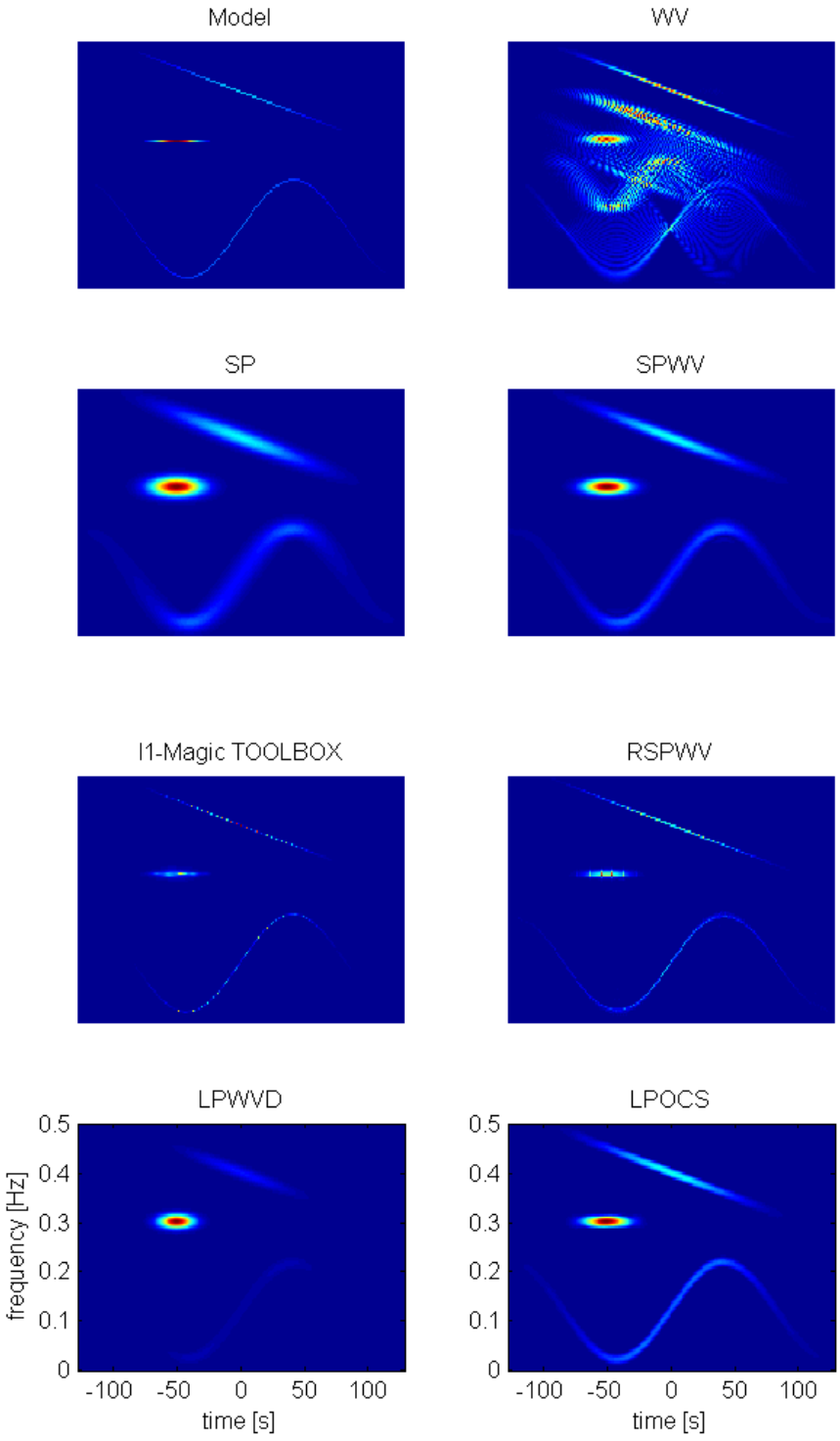


Figure 3.8: Signal 2: The TF reconstruction using, left column: the ideal model, Spectrogram, $l_1$-MAGIC TOOLBOX, L-Class Polynomial WV distribution (LP-WVD), right column: WV, the Smoothed Pseudo WV (SPWV), Reassigned SPWV (RSPWV), lifted POCS.

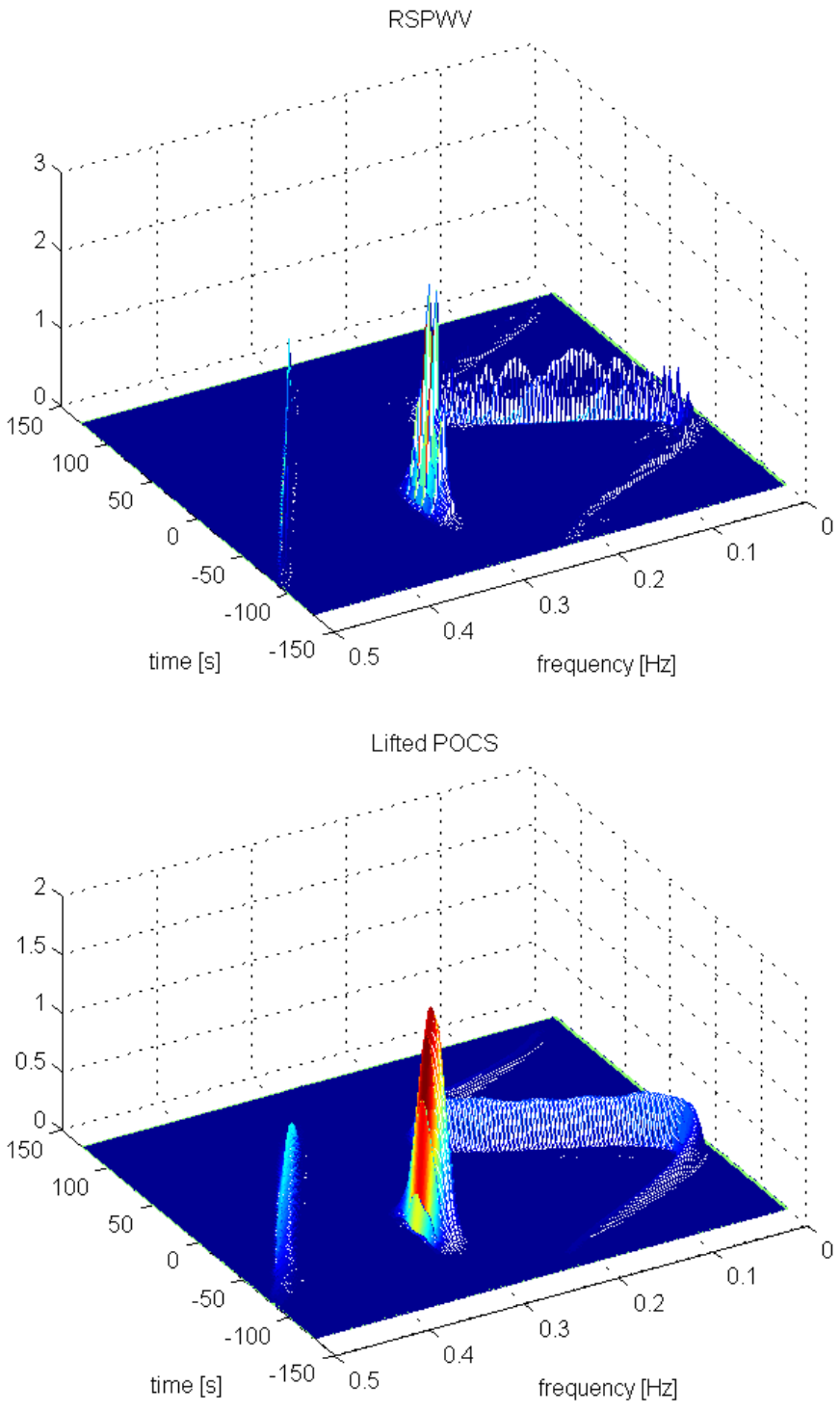


Figure 3.9: Signal 2: 3D plot of the TF distributions obtained by reassigned SPWV (RSPWV) and Lifted POCS (bottom).

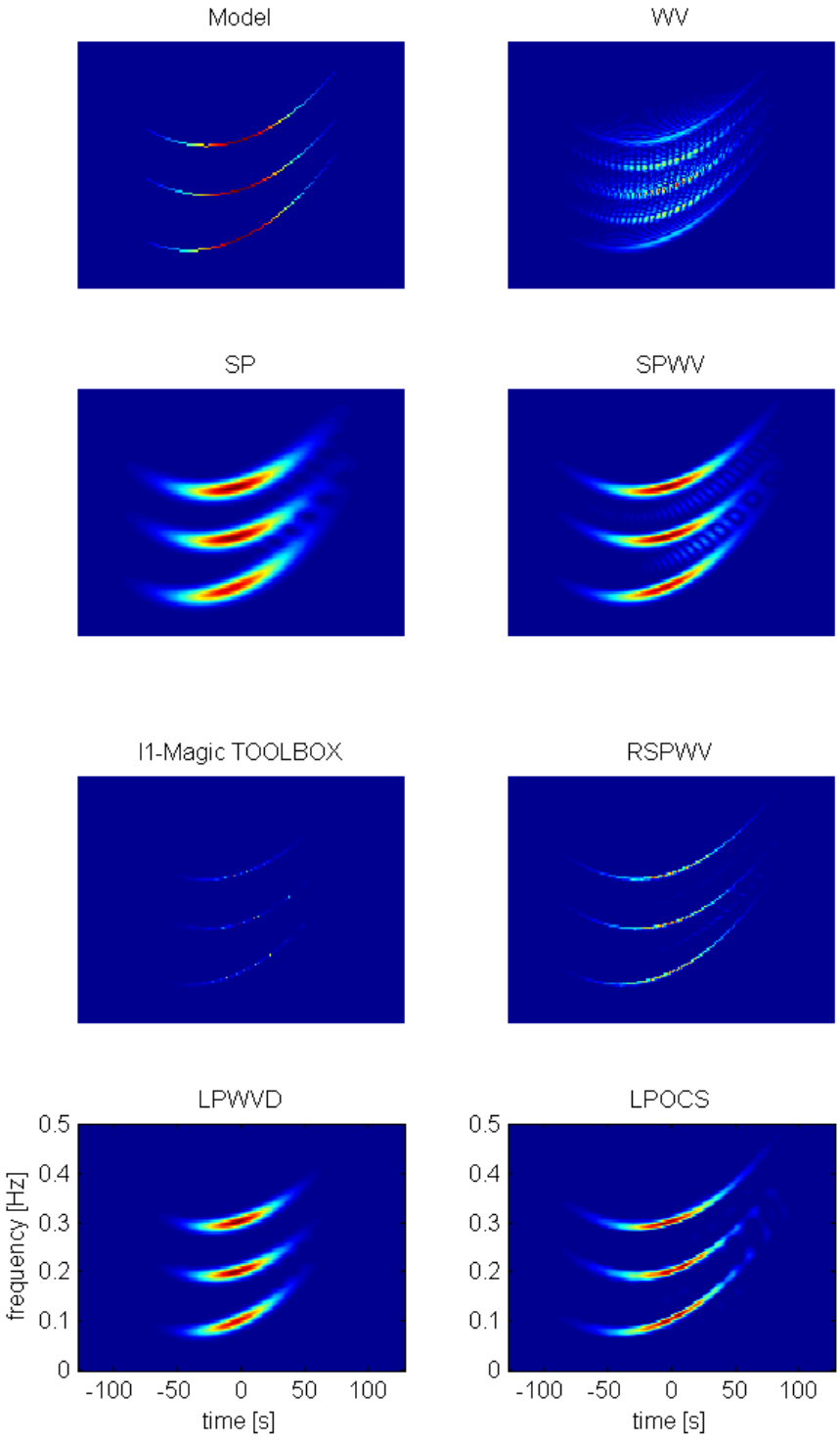


Figure 3.10: Signal 3: the TF reconstruction using, left column: the ideal model, Spectrogram, $l_1$-MAGIC TOOLBOX, L-Class Polynomial WV distribution (LP-WVD), right column: WV, the Smoothed Pseudo WV (SPWV), Reassigned SPWV (RSPWV), lifted POCS.

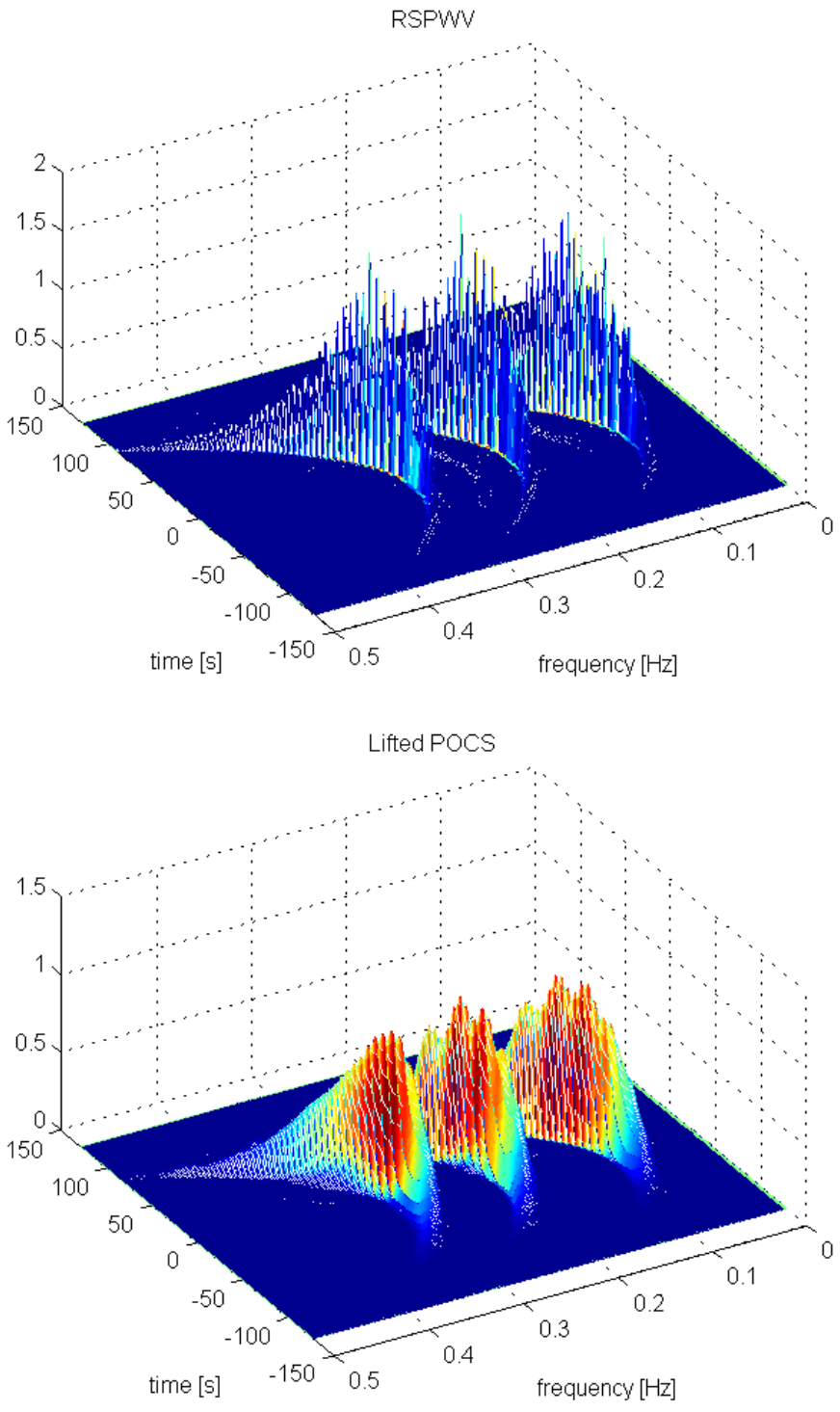


Figure 3.11: Signal 3: 3D plot of the TF distributions obtained by Reassigned SPWV (RSPWV) and Lifted POCS (bottom).

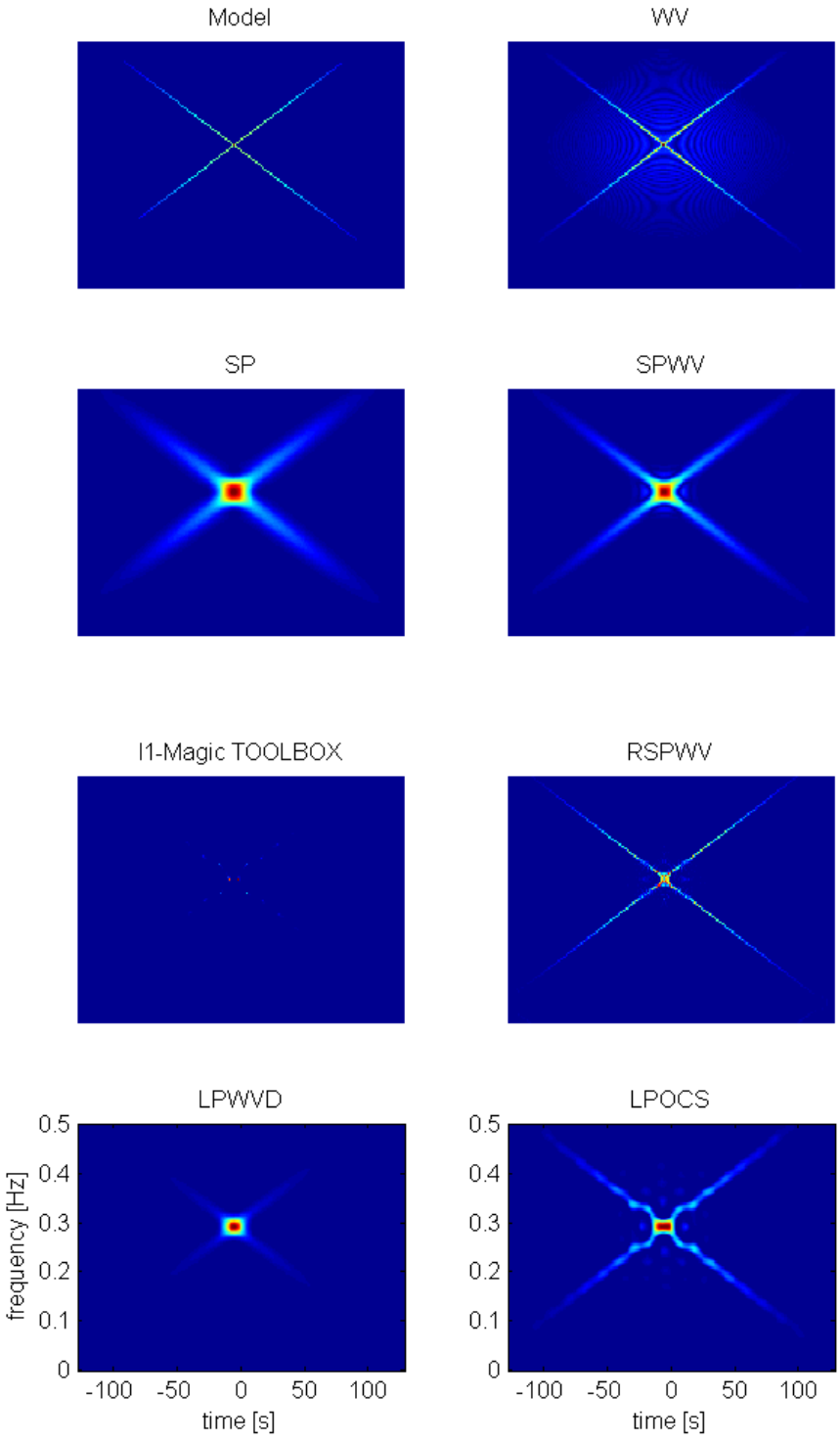


Figure 3.12: Signal 4: the TF reconstruction using, left column: the ideal model, Spectrogram, $l_1$-MAGIC TOOLBOX, L-Class Polynomial WV distribution (LP-WVD), right column: WV, the Smoothed Pseudo WV (SPWV), Reassigned SPWV (RSPWV), Lifted POCS.

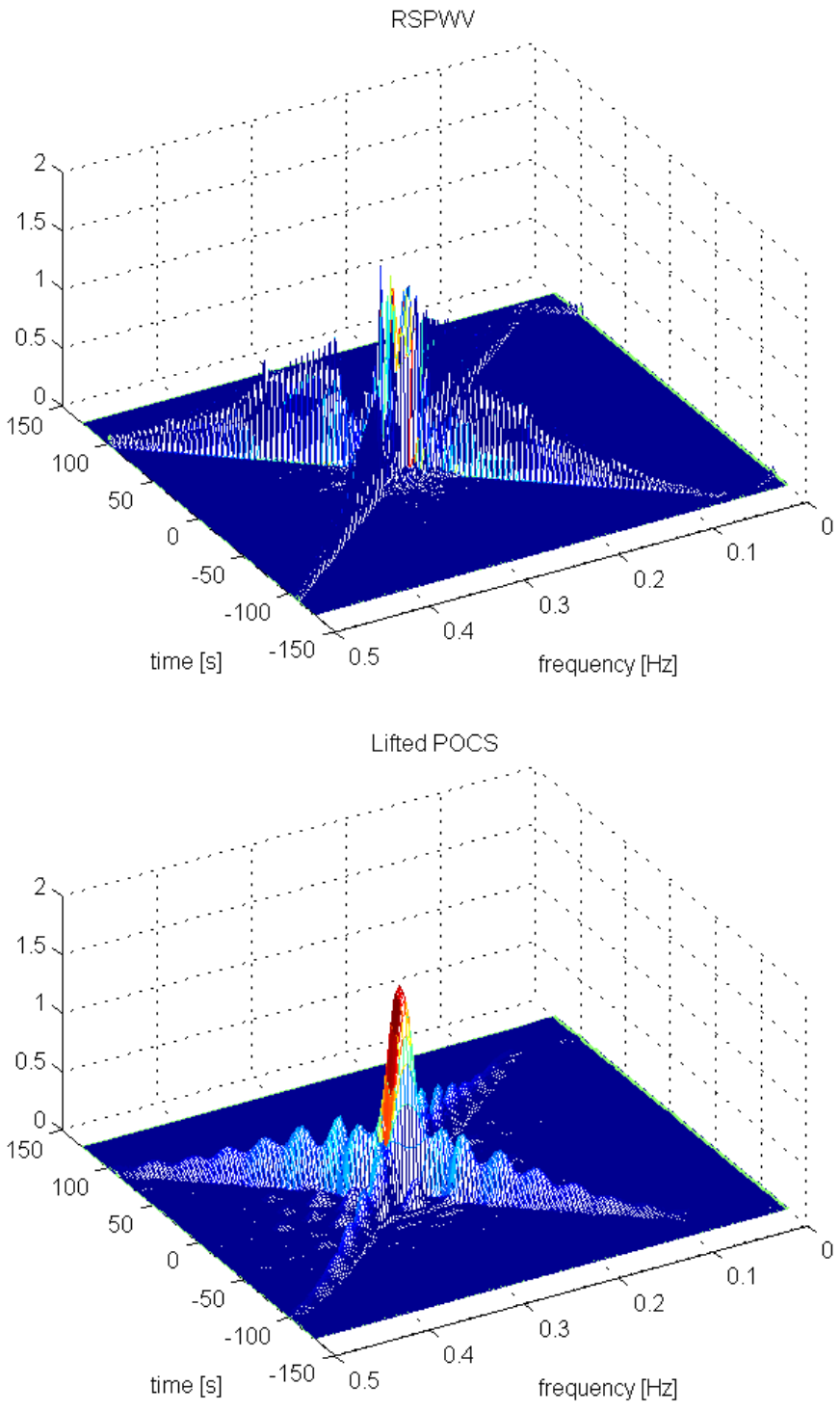


Figure 3.13: Signal 4: 3D plot of the TF distributions obtained by Reassigned SPWV (RSPWV) and Lifted POCS (bottom).

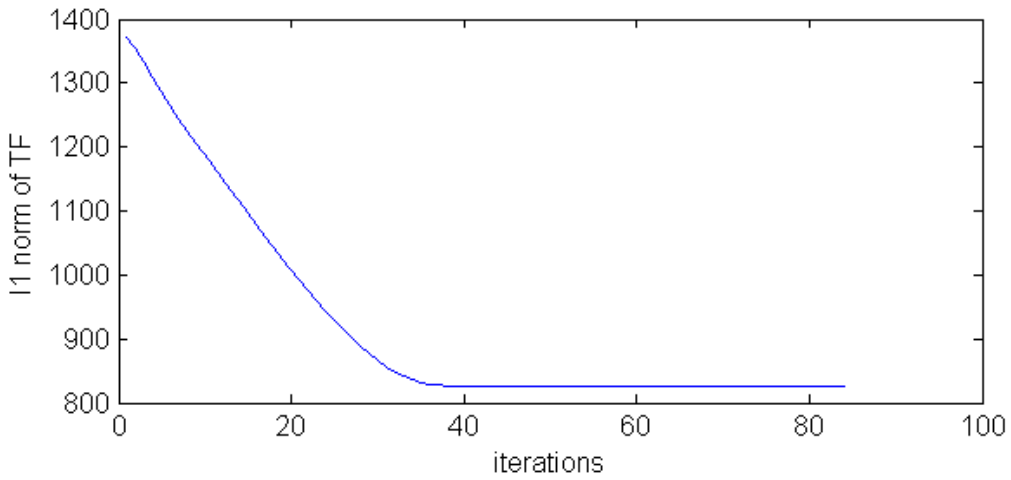


Figure 3.14: The convergence plot of the lifted POCS iterations for Signal 1: The plot shows the $l_1$-norm of the TF distribution versus the number of iterations.

reveals the hopping frequencies and the hopping rate without adjusting any parameters. It provides better localization than LPWVD and it is not spiky, either.

In Figure 3.17, TF of a short segment from a Dolphin's click signal is shown. It is known that this acoustic signal should have three FM components starting at 0.1Hz, 0.18Hz and 0.29Hz (normalized frequency) corresponding to actual 1100Hz,1980HZ and 3190 Hz respectively at first half of the observation duration. Only the spectrogram and the LPOCS solutions reveal these three components clearly and the LPOCS has a better resolution.

In order to measure the localization of each TF distribution in a quantitative way we use $l_1$-norm as a measure. Rényi entropy [79] is also a preferred method for measuring the localization. Rényi entropy is given by

$$R_P^{\alpha} = \frac{1}{1-\alpha} log_2\{\sum_{n=1}^{N}\sum_{m=1}^{N} P^{\alpha}[n,m]\} \tag{3.21}$$

where $P[n,m]$ is the TF distribution and $\alpha$ is the order of measure. Rényi entropy allows TF distribution to take negative values. The value of Rényi entropy is expressed in terms of bits. The lower the Rényi measure the better the localization. A Rényi entropy of order three was shown to be a good measure for

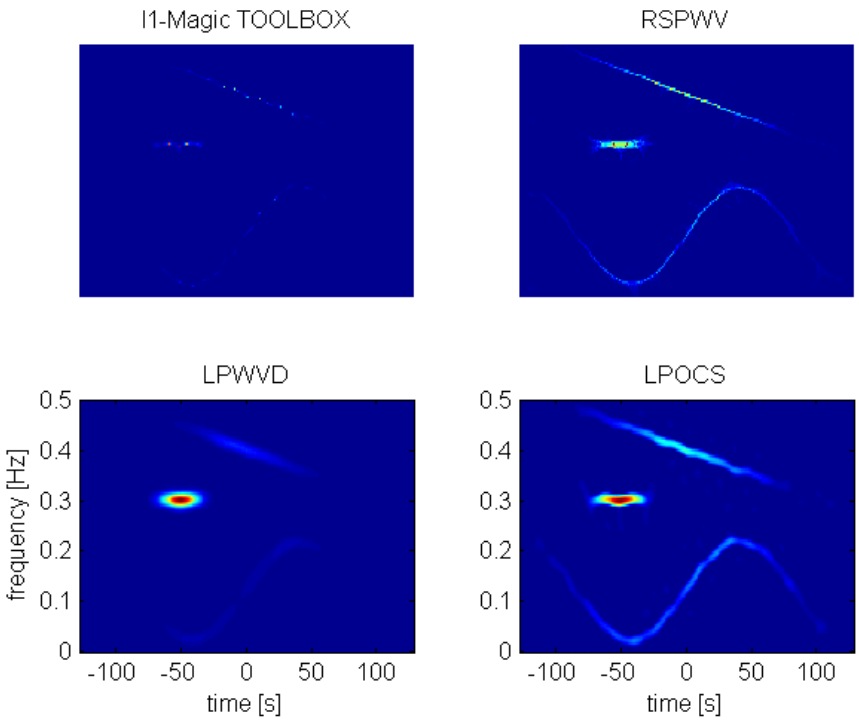


Figure 3.15: The signal in Figure 3.8 is corrupted by additive zero-mean white Gaussian noise. The $SNR$ value is 10 dB. The TF reconstruction result obtained by Lifted POCS method (bottom right) is comparable to the Reassigned Smoothed Pseudo WV (top right). The frequency is the normalized frequency.

localization [79].

Localization alone is not sufficient for a good comparison. We also need to know how similar the TF result is to the model TF we desire. There are various similarity measures depending on application. $l_2$ distance, $l_1$ distance, Kullback-Leibler divergence [80], Pearson correlation [81], earth mover's distance (EMD) [82] are some examples. EMD is a measure of the distance between two probability distributions over a region. It is the minimum cost of turning one distribution into the other. EMD is more suitable for content based image retrieval and texture comparison applications. Its computation requires an optimization problem to be solved. That is why the histogram of the images are used for the distance computation. Kullback-Leibler divergence is a measure of information divergence between the two distribution functions. Since both EMD and Kullback-Leibler

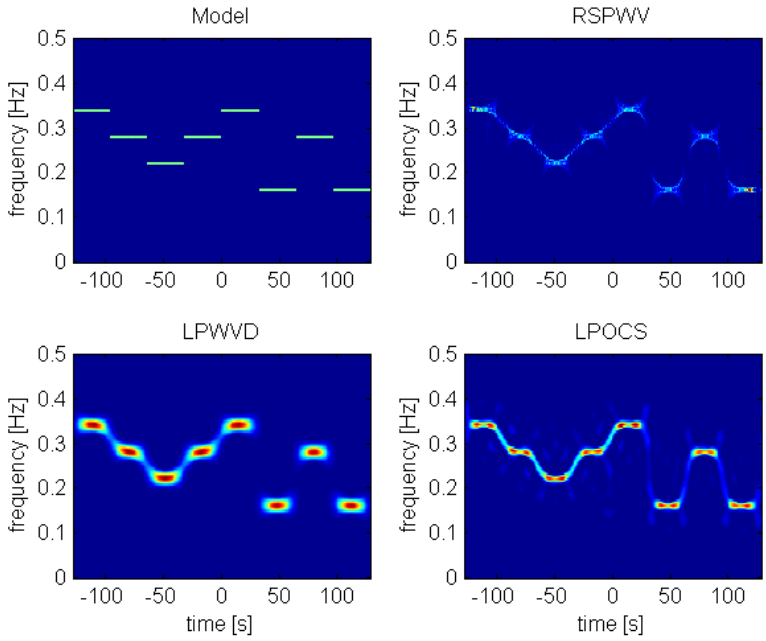


Figure 3.16: Signal 5: TFD of a Frequency Hopping MFSK signal. First row: the ideal model and Reassigned SPWV(RSPWV). Bottom row: L-Class Polynomial WV Distribution (LPWVD) and Lifted POCS. The frequency is normalized.

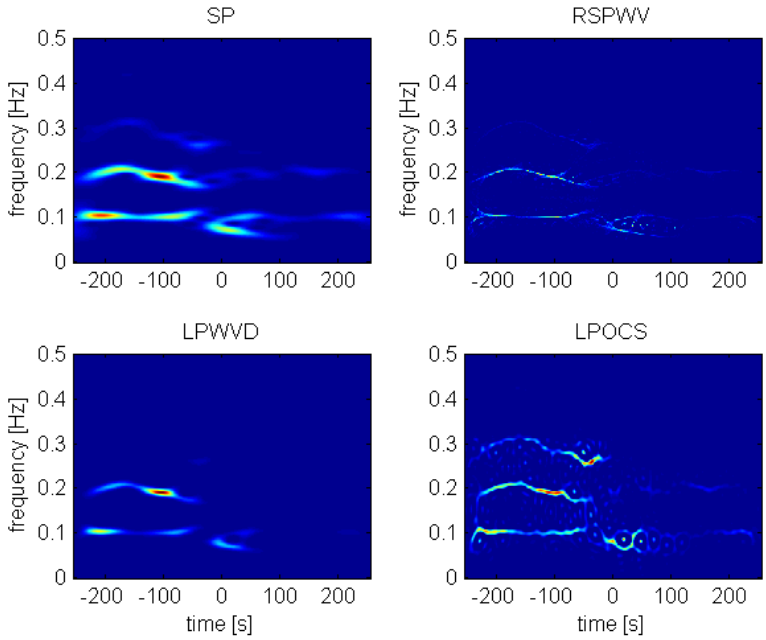


Figure 3.17: Signal 6. TFD of a dolphin's click-signal segment. Top-row: spectrogram (SP) and Reassigned SPWV (RSPWV). Bottom-row: L-Class Polynomial WV Distribution (LPWVD) and Lifted POCS. The frequency is normalized frequency.

divergence are applied to the positive functions and we have WV distribution with negative values, in this work the Pearson correlation is used as the similarity measure. The Pearson correlation coefficient between solution TF $\boldsymbol{P}$ and model TF $\boldsymbol{P}_{model}$ is given by

$$pcor = \frac{\widetilde{vec}(\boldsymbol{P})^T \widetilde{vec}(\boldsymbol{P}_{model})}{\|\widetilde{vec}(\boldsymbol{P})\|_2 \|\widetilde{vec}(\boldsymbol{P}_{model})\|_2} \tag{3.22}$$

where $\widetilde{vec}(\boldsymbol{P})$ represents the vector form of $\boldsymbol{P}$ with mean value subtracted. The Pearson correlation is widely used in the sciences as a measure of the degree of linear dependence between two variables. It measures the shape similarity rather than any exact norm difference and takes the values between -1 and 1. The value of 1 indicates total positive correlation, the value 0 corresponds to no correlation, and -1 shows total negative correlation. In other words, the value of 1 indicates exact shape match. Therefore together with Rényi entropy the Pearson correlation of the solution TF and the model TF are also used as a measure of similarity.

For a meaningful comparison we first should look at the final solution in terms of acceptability as a TF distribution related to the signal. Therefore we should first check how similar the result to the desired model. Then we consider the localization. From Table 3.1 we observe that LPOCS method and the RPSWV methods are better than all the other methods in terms of the similarity measure. When we compare the localization property of the methods the TF obtained by $l_1$-Magic TOOLBOX has the highest localization. The second one is the RPSWV and the third one is LPOCS. But as we emphasized in Figure 3.2, Figure 3.3, 3.9, 3.11 and Figure 3.13 the TF solutions obtained RPSWV and $l_1$-Magic TOOLBOX method, are spiky and they do not correspond to the physical reality of the actual signals [28]. In fact, from Table 3.2 we observe that, $l_1$-Magic TOOLBOX provides over localized results which have lower Rényi measure than the actual model TF in some cases. In Signal 6 shown in Figure 3.17, we observe that RPSWV is weak to produce the spectral lines clearly. The LPOCS method has good results both in terms of localization, similarity and physical interpretation. Furthermore, LPOCS method does not require any parameter adjustment nor parameter selection. Our overall assessment is that LPOCS is superior to the RSPWV. However the computational cost of RSPWV is lower than LPOCS.

Table 3.1: Pearson correlation coefficient between TF distributions and the model TF for tested signals. A higher value shows better similarity to the model.

| | **Pearson correlation coefficient** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Signal** | **Model** | **WV** | **SP** | **SPWV** | **$l_1$-Magic** | **LPWVD** | **RSPWV** | **LPOCS** |
| Signal 1 | 1 | 0,35 | 0,41 | 0,49 | 0,49 | 0,47 | **0,73** | 0,59 |
| Signal 2 | 1 | 0,24 | 0,38 | 0,43 | 0,29 | 0,40 | **0,52** | 0,50 |
| Signal 3 | 1 | 0,18 | 0,39 | 0,45 | 0,31 | 0,44 | 0,47 | **0,55** |
| Signal 4 | 1 | **0,45** | 0,34 | 0,40 | 0,09 | 0,28 | 0,42 | 0,40 |
| Signal 5 | 1 | 0,11 | 0,34 | 0,35 | 0,20 | 0,38 | 0,34 | **0,44** |
| Signal 6 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A |

Table 3.2: Rényi entropy of all the TF distributions for tested signals. A lower value indicates better localization.

| | **Rényi entropy** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Signal** | **Model** | **WV** | **SP** | **SPWV** | **$l_1$-Magic** | **LPWVD** | **RSPWV** | **LPOCS** |
| Signal 1 | 8,13 | 9,01 | 11,51 | 10,97 | **7,18** | 10,44 | 8,47 | 10,14 |
| Signal 2 | **7,46** | 9,56 | 11,18 | 10,66 | 8,06 | 8,84 | 8,74 | 9,99 |
| Signal 3 | 8,24 | 9,20 | 11,77 | 11,18 | **7,59** | 10,69 | 8,82 | 10,35 |
| Signal 4 | 8,04 | 8,51 | 11,01 | 10,41 | **5,45** | 8,46 | 8,57 | 9,65 |
| Signal 5 | 7,99 | 9,77 | 11,88 | 11,23 | **6,53** | 10,86 | 9,45 | 10,23 |
| Signal 6 | N/A | 10,99 | 13,11 | 12,39 | 9,53 | 11,28 | 10,73 | 11,70 |

In both the proposed lifted POCS method and in [28] the cardinality of the set $\Omega$ is very low compared to the actual TF signal ($N \times N$). In all the signal examples tried in this section the size of the $\Omega$ set is selected as a circle with radius $r_0 = N/16$. This is necessary to remove the cross-terms [28]. When the sets $C_f$ and $C_{AF}$ intersect there may be many solutions satisfying the constraints specified by the sets $C_f$ and $C_{AF}$. In this case the solution depends on the initial vector. In all the examples iterations start with a 2-D distribution obtained from the actual AF by a masking window with radius $r_0 = N/16$. In Signal 1, the initial starting distribution for the iterative lifted POCS approach is shown in the bottom-left plot of Figure 3.1. In other signal examples the related masked AF is used as the initial estimate. They are all relatively smooth VW distributions. When the

sets $C_f$ and $C_{AF}$ do not intersect the iterations converge to either one of the two unique distributions as graphically illustrated in Figure 3.4.

Different mask sizes in $C_{AF}$ ranging from $r_0 = N/12$ to $N/24$ were also tried. This range of masks successfully removed cross terms in all of the above cases. Therefore the choice of $r_0$ is not very critical to estimate a cross-term free VW distribution. However, the question of optimal $r_0$ value or the shape of the mask $\Omega$ for a given time-varying signal remains as an open problem. In the next section a method is proposed to solve this problem.

The computational cost of the lifted POCS method and the method in [28] are higher than the classical VW and other AF shaping or smoothing based methods. This is because the optimization problems posed by both methods are solved in an iterative manner. Obviously, estimated VW distributions are better in terms of cross-terms compared to classical methods. Considering the good localization achieved in the TF plane without any regularization parameter, the lifted POCS method is a promising approach for the TF distribution estimation.

# Chapter 4

# Cross term-free TF Reconstruction using Partial AF Information

## 4.1 Introduction

Similar to any POCS method, the solution vector, or the TFD in our case, is in the intersection of the convex sets in Lifted POCS method [30], [71]. The POCS iterations, starting from an initial vector, will converge, provided that the intersection is non empty. Therefore, the solution in general depends on two things: (i) the intersection situation of the convex sets and (ii) the initial vector. If the intersection contains a single vector then the solution is unique and independent of the initial point. If the intersection contains more than one vector then the solution will be the closest point in intersection to the initial vector according to Euclidean norm. If the intersection is empty then the POCS method oscillates between vectors of the sets which are the closest vectors between the sets. In this regard the solution should be selected from one of these vectors depending on some acceptability criteria. Both scenarios are graphically illustrated in Figure 3.4.

In [28], the solution to the constrained $l_1$ minimization problem

$$\begin{aligned} &\boldsymbol{P}^* = \text{argmin}_{\boldsymbol{P}} \, \|\boldsymbol{P}\|_1 \\ &\text{subject to} \quad \mathcal{F}^{-1}\{\boldsymbol{P}\} = \boldsymbol{A}_x[k,l] \qquad k,l \in \Omega \end{aligned} \tag{4.1}$$

or its relaxed version was found via interior points methods. In Section 3.2 the solution to a localized TF with AF domain constraints was obtained using lifted projection based POCS method. The constraints were expressed with the following closed and convex sets:

$$C_f = \{\boldsymbol{w} = [vec(\boldsymbol{P})^T \quad f(\boldsymbol{P})]^T \in \boldsymbol{R}^{N^2+1} \quad | \quad f(\boldsymbol{P}) = \|\boldsymbol{P}\|_1 \leq v\} \tag{4.2}$$

and

$$C_{AF} = \{\boldsymbol{w} = [vec(\boldsymbol{P})^T \quad v]^T \in \boldsymbol{R}^{N^2+1} \mid \quad \mathcal{F}^{-1}\{\boldsymbol{P}\} = \boldsymbol{A}_x[k,l] \quad k,l \in \Omega\} \tag{4.3}$$

In the POCS method, whether the sets $C_f$ and $C_{AF}$ intersect or not and the size of intersection depends on $C_{AF}$. The definition of epigraph set $C_f$ does not require any parameter selection. But the AF domain set $C_{AF}$ is dependent on the subset $\Omega$. In Section 3.2, the shape of $\Omega$ was selected as a circle around origin in AF domain for the tested signal examples and the radius of the circle was fixed to $r_0 = N/16$, where $N$ being the discrete signal length. With this selection of $\Omega$, the results, comparable to Reassigned Smoothed Pseudo WV (RSPWV) and better than the others, were obtained in terms of overall similarity and localization. Similar results were obtained by ranging $r_0$ between $r_0 = N/24$ and $r_0 = N/12$. But even better results can be obtained by a proper selection of the AF domain constraint set. In this respect the selection of type and size of $\Omega$ is an open problem as in [28].

In this section two alternative convex set definitions are used in AF domain, for the aim of both having better results or further relaxing the dependency on the selection of the set.

## 4.2 Reconstruction with Real Part of AF Coefficients

The first AF domain set is defined in the following way:

$$C_{AF} = \{\boldsymbol{w} = [vec(\boldsymbol{P})^T\ v]^T \in \boldsymbol{R}^{N^2+1} \mid \quad Re\{\mathcal{F}^{-1}\{\boldsymbol{P}\}\} = Re\{\boldsymbol{A}_x[k,l]\} \quad k,l \in \Omega\} \tag{4.4}$$

The motivation behind this type of selection is that, in AF domain we have the complex valued coefficients and they are represented by two numbers, that is, real and imaginary parts. Therefore instead of forcing the reconstructed TF distribution to have all coefficients in a given AF area to be equal to the original $\boldsymbol{A}_x$ we may force only one part, real or imaginary, and let the other to be determined during optimization process. In other words, for the same AF domain area as in original lifted POCS method, we allow a further freedom during optimization process. In this way, the amplitude or phase of reconstructed AF, that is $\mathcal{F}^{-1}\{\boldsymbol{P}\}$ in $\Omega$, is allowed to vary rather than being equal to $\boldsymbol{A}_x[k,l]$. But still it will have some correlation with $\boldsymbol{A}_x[k,l]$, as they have the same real part. It can be trivially shown that this is also a convex and closed set. The pseudo-code for this Lifted POCS algorithm with real part of AF coefficients is given in Algorithm 2. The AF domain set area is set to circular area with the same radius $r_0 = N/16$ as the original lifted POCS in 3.2.

**Algorithm 2** The pseudo-code for Lifted POCS with real AF coefficients

**function** $\boldsymbol{P}$=LPOCSR($\boldsymbol{x}$)
$N = length(\boldsymbol{x})$; ;$\Omega = circ(N/16)$
$\boldsymbol{A}_x$:=AF($\boldsymbol{x}$) $\boldsymbol{A}_0$:=mask($\boldsymbol{A}_x$,$\Omega$)
$\boldsymbol{P}_0 := \mathcal{F}\{\boldsymbol{A}_0\}$ ; $\boldsymbol{w}_0 := [vec(\boldsymbol{P}_0)^T \quad 0]^T$
$i = 1$ ; $\epsilon = 10^{-5}$
**while** $err \geq \epsilon$ **do**
$\boldsymbol{w}_i = \min_{\boldsymbol{w} \in C_f} \|\boldsymbol{w} - \boldsymbol{w}_{i-1}\|_2^2 = [vec(\boldsymbol{P}_i)^T \quad w_{i,n+1}]^T$
$\boldsymbol{A}_i = \mathcal{F}^{-1}\{\boldsymbol{P}_i\}$
$Re\{\boldsymbol{A}_i|_\Omega\} := Re\{\boldsymbol{A}_x|_\Omega\}$
$\boldsymbol{P}_i := \mathcal{F}\{\boldsymbol{A}_i\}$
$\boldsymbol{w}_i := [vec(\boldsymbol{P}_i)^T \quad w_{i,n+1}]^T$
$err = \|vec(\boldsymbol{P}_i) - vec(\boldsymbol{P}_{i-1})\|_2 / \|vec(\boldsymbol{P}_{i-1})\|$
$i = i + 1$
**end while**
**end function**

The reconstructed TF results with this new set for Signal 1 and 2 are shown in Figures 4.5 and 4.4. From these figures, we observe that there is a slight improvement in localization compared to original Lifted POCS method. In order to see the improvement quantitatively, the localization and similarity of the TF representations with this new set were computed using (3.21) and (3.22) and the results are listed in Tables 4.1 and 4.2. From these tables we see that selecting only the real parts of the $\boldsymbol{A}_x$ in selected area $\Omega$ as the constraint set, improves both the localization and similarity. In this respect the similarity of the TF distribution obtained with this method is better than the Reassigned Smoothed Pseudo WV (RSPWV).

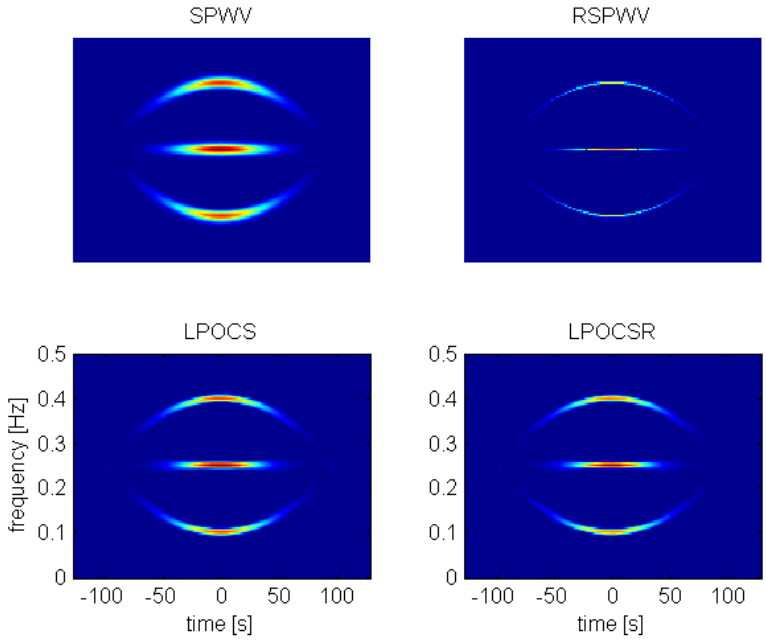


Figure 4.1: Signal 1: the TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right).

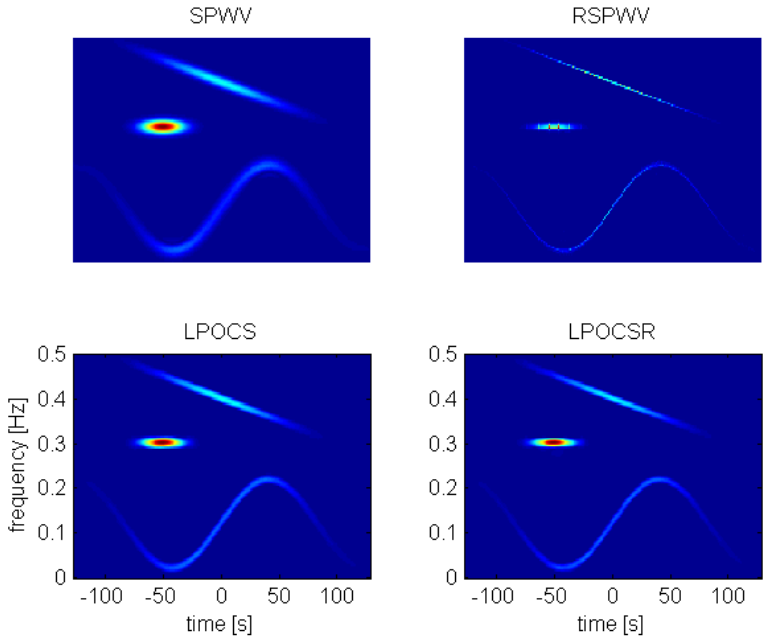


Figure 4.2: Signal 2: the TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right).

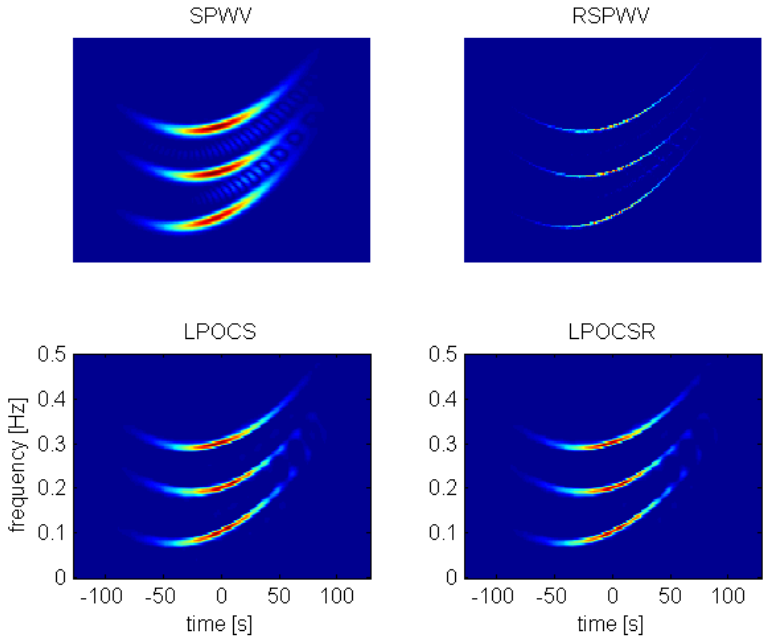


Figure 4.3: Signal 3: the TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right).

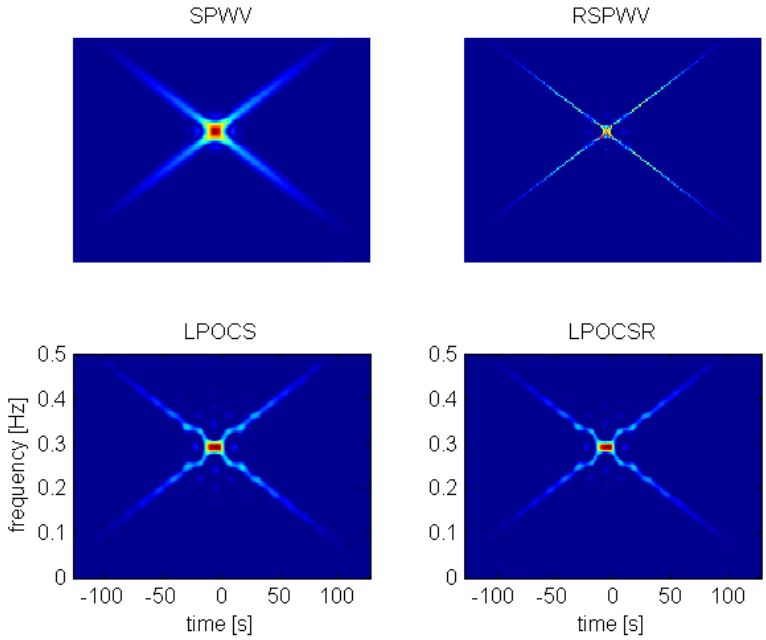


Figure 4.4: Signal 4: the TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right).

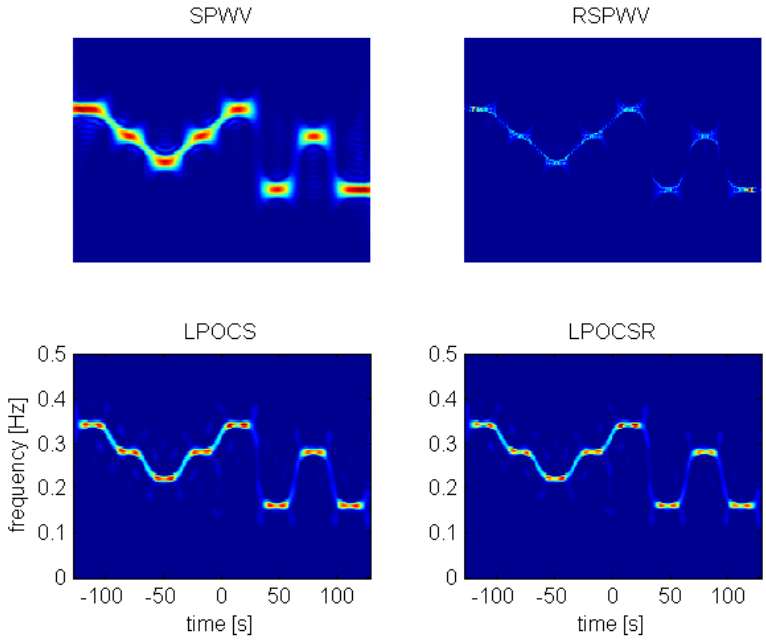


Figure 4.5: Signal 5: TFD of a Frequency Hopping MFSK signal. The TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right).

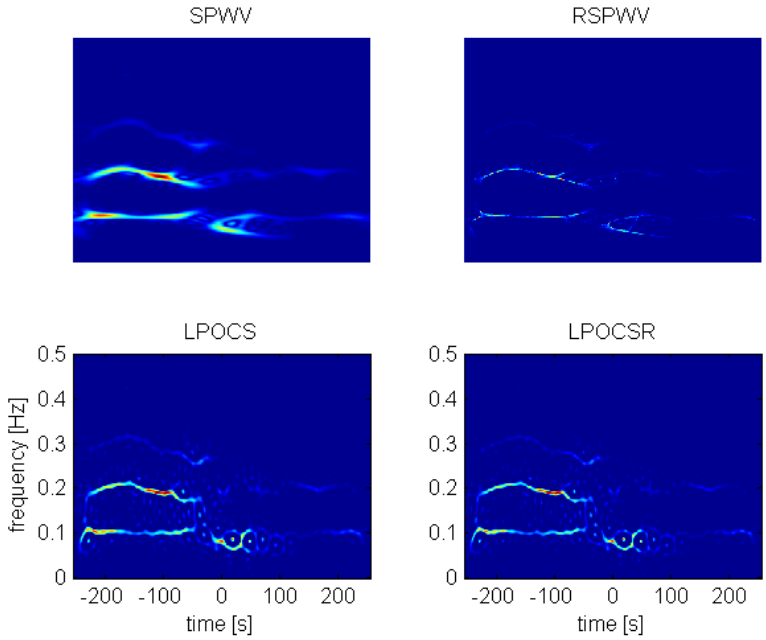


Figure 4.6: Signal 6: TFD of a dolphin's click-signal. The TF reconstruction using, Smoothed Pseudo WV (SPWV) (top left), the reassigned Smoothed Pseudo WV (RSPWV) (top right), Lifted POCS (bottom left) and Lifted POCS with real AF coefficients (bottom right).

Table 4.1: Pearson correlation coefficient between TF distributions and the model TF for tested signals. A higher value shows better similarity to the model.

| | Pearson correlation coefficient | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Signal** | **Model** | **WV** | **SP** | **SPWV** | **$l_1$-Magic** | **RSPWV** | **LPOCS** | **LPOCSR** |
| Signal 1 | 1 | 0,35 | 0,41 | 0,49 | 0,49 | **0,73** | 0,59 | 0,62 |
| Signal 2 | 1 | 0,24 | 0,38 | 0,43 | 0,29 | **0,52** | 0,50 | **0,52** |
| Signal 3 | 1 | 0,18 | 0,39 | 0,45 | 0,31 | 0,47 | 0,55 | **0,58** |
| Signal 4 | 1 | **0,45** | 0,34 | 0,40 | 0,09 | 0,42 | 0,40 | 0,40 |
| Signal 5 | 1 | 0,11 | 0,34 | 0,35 | 0,20 | 0,34 | **0,44** | **0,44** |
| Signal 6 | N/A | N/A | N/A | N/A | N/A | N/A | N/A | N/A |

Table 4.2: Rényi entropy of all the TF distributions for tested signals. A lower value indicates better localization.

| | Rényi entropy | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Signal** | **Model** | **WV** | **SP** | **SPWV** | **$l_1$-Magic** | **RSPWV** | **LPOCS** | **LPOCSR** |
| Signal 1 | 8,13 | 9,01 | 11,51 | 10,97 | **7,18** | 8,47 | 10,14 | 9,92 |
| Signal 2 | **7,46** | 9,56 | 11,18 | 10,66 | 8,06 | 8,74 | 9,99 | 9,73 |
| Signal 3 | 8,24 | 9,20 | 11,77 | 11,18 | **7,59** | 8,82 | 10,35 | 10,12 |
| Signal 4 | 8,04 | 8,51 | 11,01 | 10,41 | **5,45** | 8,57 | 9,65 | 9,32 |
| Signal 5 | 7,99 | 9,77 | 11,88 | 11,23 | **6,53** | 9,45 | 10,23 | 10,01 |
| Signal 6 | N/A | 10,99 | 13,11 | 12,39 | 9,53 | 10,73 | 11,70 | 11,45 |

## 4.3 Reconstruction with only the Phase of AF Coefficients

The second AF domain set is defined in the following way:

$$C_{AF_\phi} = \left\{ \boldsymbol{w} = [vec(\boldsymbol{P})^T \; v]^T \in \boldsymbol{R}^{N^2+1} \;\middle|\; \begin{array}{l} \phi(\mathcal{F}^{-1}\{\boldsymbol{P}\}) = \phi(\boldsymbol{A}_x), \\ \mathcal{F}^{-1}\{\boldsymbol{P}\} = \boldsymbol{A}_x[k,l] \quad k,l = 0 \end{array} \right\} \tag{4.5}$$

The definition in (4.5) is simply the set of TF distributions whose corresponding ambiguity function has its phases and value at origin (DC term) equal to $\boldsymbol{A}_x$, where $\phi(.)$ represents the phase term. It can be shown that $C_{AF_\phi}$, is a closed and convex set. Both the sets $C_f$ and $C_{AF_\phi}$ are defined in lifted domain $\boldsymbol{R}^{N^2+1}$.

The motivation behind this selection of the AF domain set is similar to that of using real part of AF coefficients explained in Section 4.2. The phases and DC term of the AF to be reconstructed will be equal to that of the signal but the magnitudes will be determined during the coarse of optimization process based on localization requirement. The phase information is very important and in many signal processing applications allows the signal reconstruction without magnitude knowledge [83]. With this selection we will get rid of the AF domain set type and size selection problem.

The lifted POCS algorithm iterates in the following way: Given an initial TF distribution $\boldsymbol{P}_0$ we construct a corresponding vector in $\boldsymbol{R}^{n^2+1}$ by padding a zero at the very end as follows: $\boldsymbol{w}_0 = [vect(\boldsymbol{P}_0)^T \quad 0]^T \in \boldsymbol{R}^{N^2+1}]$ , whose orthogonal projection $\boldsymbol{w}_1$ onto $C_f$ is defined as follows:

$$\boldsymbol{w}_1 = \min_{\boldsymbol{w} \in C_f} \|\boldsymbol{w} - \boldsymbol{w}_0\|_2^2 \tag{4.6}$$

The solution of the minimization problem (4.6) is explained in Appendix A.1. Next, the vector $\boldsymbol{w}_1$ is projected onto $C_{AF_\phi}$ producing the next iterate $\boldsymbol{w}_2$ . The corresponding TF matrix $\boldsymbol{P}_2$ satisfies $\phi(\mathcal{F}^{-1}\{\boldsymbol{P}_2\}) = \phi(\boldsymbol{A}_x)$ and $\mathcal{F}^{-1}\{\boldsymbol{P}_2\} = \boldsymbol{A}_x[k,l] for k,l = 0$. This projection corresponds to the AF domain constraint. It is implemented very easily using the 2D inverse Fourier Transform. The ambiguity function corresponding to $\boldsymbol{P}_1$ is computed as follows:

$$\boldsymbol{A}_1 = \mathcal{F}^{-1}\{\boldsymbol{P}_1\}, \tag{4.7}$$

The ambiguity function $\boldsymbol{A}_2$ is defined using the actual phase values and DC term of $\boldsymbol{A}_x$:

$$\phi(\boldsymbol{A}_2) = \phi(\boldsymbol{A}_x) \tag{4.8}$$

and the value at origin (DC term) is also set to that of $\boldsymbol{A}_x$:

$$\boldsymbol{A}_2[0,0] = \boldsymbol{A}_x[0,0]. \tag{4.9}$$

Next $\boldsymbol{P}_2$ is obtained by computing the 2D FT of $\boldsymbol{A}_2$. In the second round of POCS iterations $\boldsymbol{P}_2$ or equivalently $\boldsymbol{w}_2 = [vect(\boldsymbol{P}_2)^T \quad f(\boldsymbol{P}_1)]^T$ is constructed where first $N$ entries are taken from $\boldsymbol{P}_2$ and $(N+1)^{th}$ entry is taken from previous projection onto $C_f$ because we have not changed it during projection onto $C_{AF}$. Then $\boldsymbol{w}_2$ is

projected back onto $C_f$ to obtain $\boldsymbol{P}_3$. The lifted POCS iterations continue until a satisfactory level of convergence is achieved.

Assuming that the intersection of $C_f$ and $C_{AF\phi}$ is non-empty, the iterations will converge to a point in intersection set. Or, they oscillate between $C_f$ and $C_{AF\phi}$. Both cases are fine with us because we look for a compromise solution for the TF distribution.

In Figures 4.7 - 4.11 results using this method are shown. From the figures, we observe that, the phase only method, except for the Signal 4 successfully reconstruct a high resolution TFD comparable to LPOCS method presented in 3.2 without specifying the constraint set size $\Omega$. For the Signal 4 it was observed that using an impulse at origin of TF plane given by,

$$\boldsymbol{P}_0[n,m] = \delta[n,m] = \begin{cases} 1 & n,m = 0 \\ 0 & else \end{cases} \tag{4.10}$$

as the initial TF, produces a better result. The result with this initial is shown in Figure 4.12. The result is highly localized and does not have any cross terms. From this result it obvious that a proper initial TF selection strategy is needed for better results. The possible reason that the example Signal 4 has failed with the standard initialization can be attributed to its frequency content. The signal has two chirps which sweep the entire spectrum. In cases where the selection of initial has an important effect on the solution, the Fourier transform of the signal can be analyzed. Based on this analysis a proper initial TF can be selected. If the signal is a full-band signal in Fourier domain then $P_0[n,m]$ should be selected as a narrow mask.It is selected as an impulse for Signal 4. Both time domain and Fourier domain representation of all the tested signal examples are given in Appendix B.

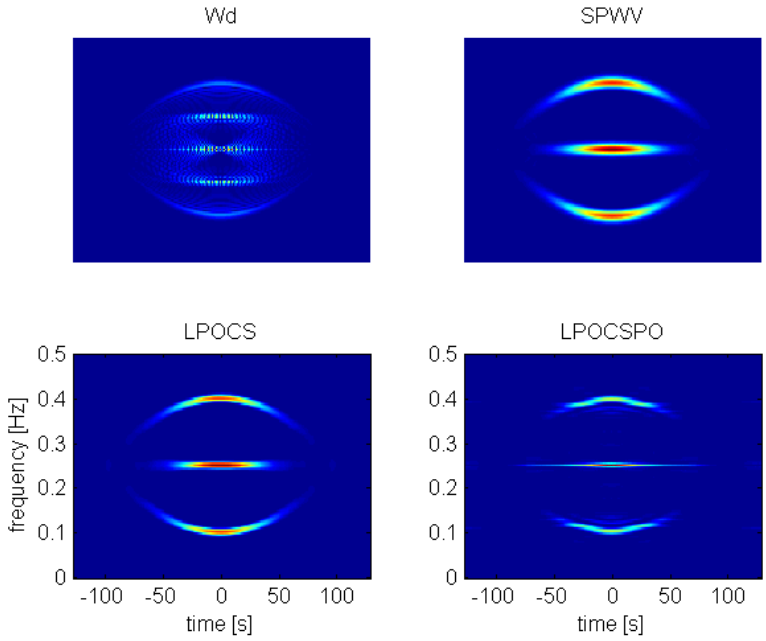


Figure 4.7: Signal 1: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right).

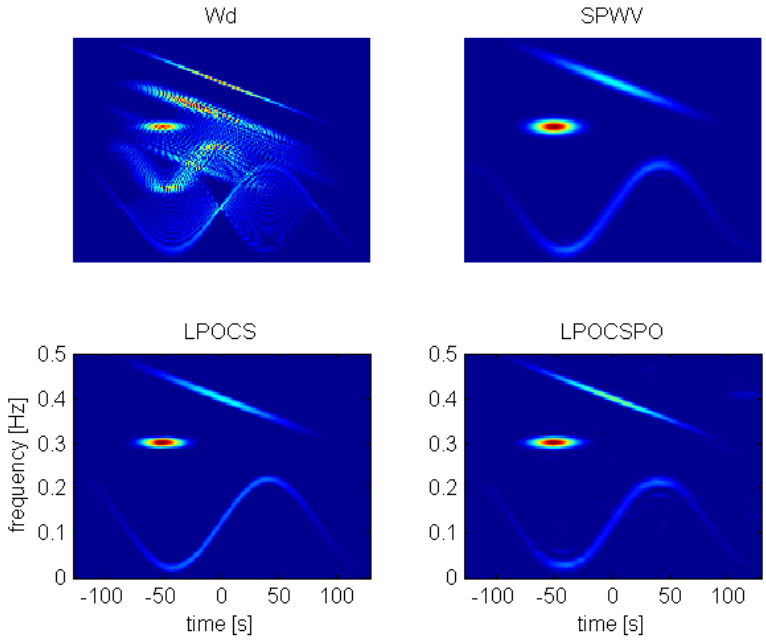


Figure 4.8: Signal 2: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right).

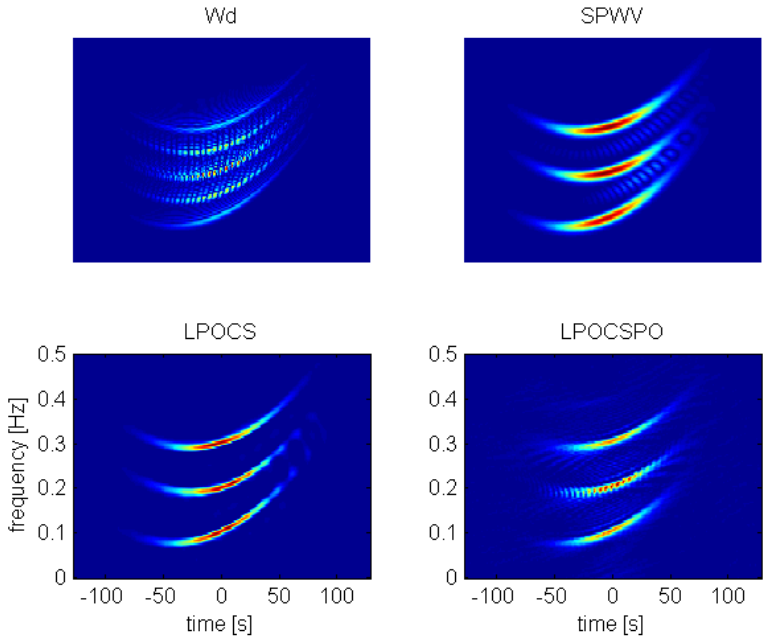


Figure 4.9: Signal 3: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right).

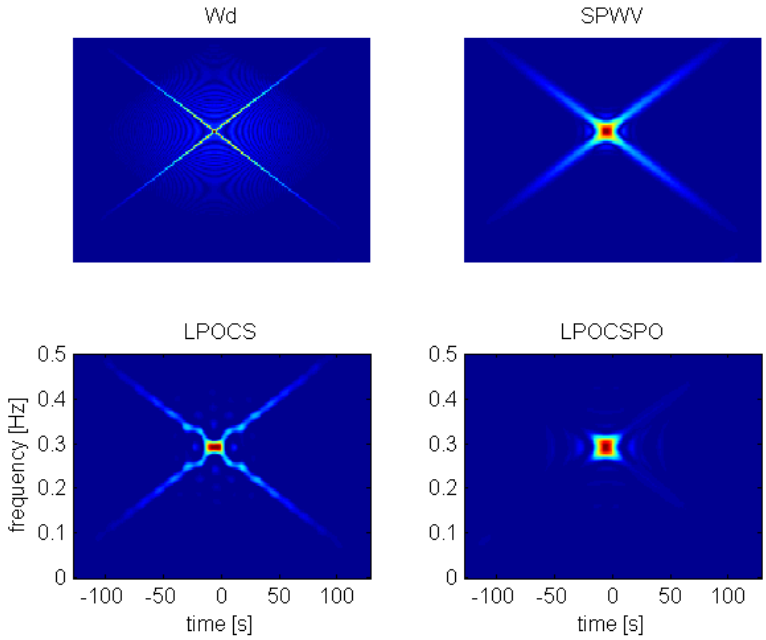


Figure 4.10: Signal 4: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right).

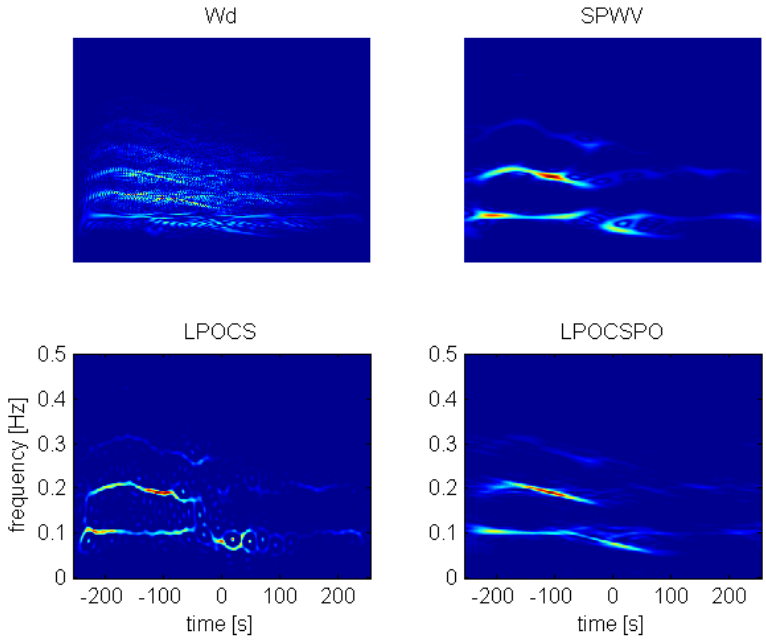


Figure 4.11: Signal 6: the TF reconstruction using, WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right).

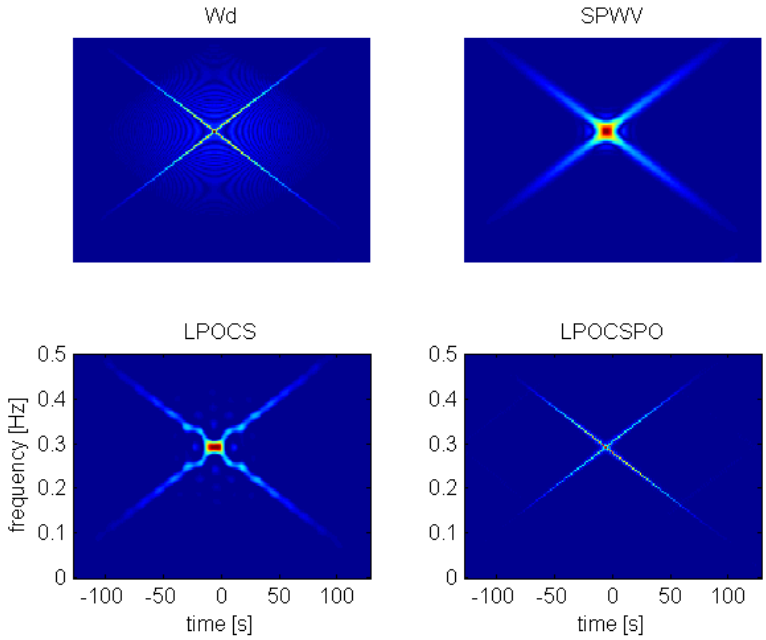


Figure 4.12: Signal 4: the TF reconstruction with initial TF as an impulse given by 4.10 at origin of TF plane: WV (top left), the Smoothed Pseudo WV (SPWV) (top right), lifted POCS (bottom left) and Lifted POCS with only the phase of AF coefficients (bottom right).

# Chapter 5

# Smoothing Kernel Estimation by Projection onto the Epigraph Set of $l_1$ norm

## 5.1 Kernel Design with Optimization

Although Wigner-Ville (WV) distribution has many pleasing properties [21], its bilinear or quadratic definition gives rise to spurious structures or so-called cross terms in TF plane. These spurious structures are result of interaction among different component of a multi-component signal. Even for a mono component signal this interaction happens between parts of the signal having nonlinear IF function. That is why the cross terms are classified as outer cross terms or interferences, which result from interaction of different components (auto-components) and the inner cross terms which appear due to interaction of different part of the component itself.

Based on the analysis of the cross terms [84], [85] the following observations are obtained. The cross terms might have a peak value as high as twice that of the auto-components, they are formed at mid-time and mid-frequency of the

auto-components and they are highly oscillatory. The frequency of oscillations increases with the increasing distance in time and frequency. Based on these observations, there have been many studies to remove the cross terms via some sort of smoothing. Traditionally the cross term removal from a signal is based on masking the Ambiguity Function (AF), which is given by:

$$P_x(t,f) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} A_x(\tau,\theta)\Phi(\tau,\theta)e^{-j\theta t - j2\pi f\tau}d\theta d\tau, \tag{5.1}$$

where $A_x(\tau,\theta)$ is the ambiguity function (AF) of the signal $x$ and $\Phi(\tau,\theta)$ is the mask or low-pass filter that has the smoothing effect in TF plane. The distributions obtained by smoothing the WD are studied under the name of Cohen's bilinear class of timefrequency distributions. $\Phi(\tau,\theta)$ is the kernel of Cohen's class TF distribution.

The low-pass smoothing kernel $\Phi(\tau,\theta)$ is designed with the objective of passing the autoterms which are centered at the origin of the AF plane and suppressing the crossterms which are located away from the origin. Initially fixed kernels were designed to address the problem such as Choi and Williams [86], Papandreou and Boudreaux-Bartels [87]. The Gaussian kernel, which is one of the examples to these kind of fixed kernels represented in polar coordinates, is defined as,

$$\Phi^p(r,\phi) = exp(\frac{-r^2}{2\sigma(\phi)^2}) \tag{5.2}$$

where $r$ and $\phi$ are the parameters of polar coordinates and the standard deviation $\sigma(\phi) = \sigma_0$ is usually chosen constant.

The circular kernel,which was also used in Section 3.2 to obtain an initial TF distribution, is defined as

$$\Phi^p(r,\phi) = \begin{cases} 1 & r \le r_0 \\ 0 & else \end{cases} \tag{5.3}$$

Figure 5.1 shows the result of smoothing with a circular kernel having radius $r_0 = N/16$ and a Gaussian kernel as in (5.2) having $\sigma(\phi) = N/16$. With both circular and Gaussian kernel the cross terms are substantially removed but at the cost of reduced resolution. The Gaussian kernel has a better result compared

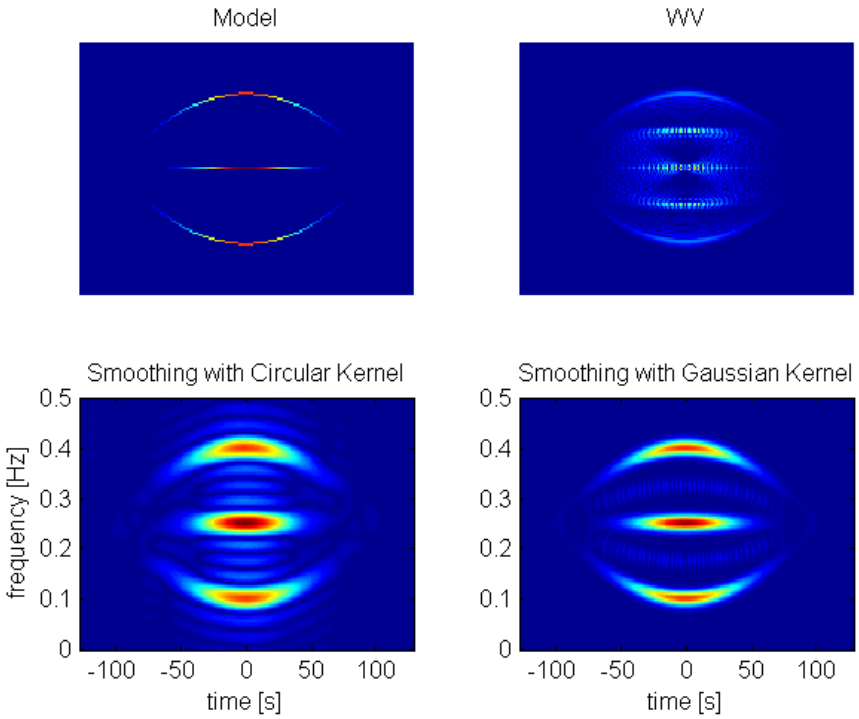


Figure 5.1: Smoothing the WV distribution with a circular kernel (bottom left $r_0 = N/16$) and with Gaussian kernel (bottom right,$\sigma(\phi) = N/16$)

to the circular one. But we do not know the best shape or type of the kernel. With some shape better results can be obtained and this shape will be signal dependent.

In order to adapt the kernel to the signal signal dependent kernels were designed by Jones and Baraniuk [88]- [89] by solving the following optimization problem

$$\max_{\Phi^p} \int_0^{2\pi} \int_0^{\infty} |A_x^p(r,\phi)\Phi^p(r,\phi)|^2 r dr d\phi, \tag{5.4}$$

$$\text{subject to } \frac{1}{4\pi^2} \int_0^{2\pi} \int_0^{\infty} |\Phi^p(r,\phi)|^2 r dr d\phi$$

$$= \frac{1}{4\pi^2} \int_0^{2\pi} \sigma^2(\phi) d\phi \leq \alpha, \quad \alpha \geq 0, \tag{5.5}$$

where, $A_x^p(r,\phi) = A_x(rcos\theta, rsin\theta)$ is the polar representation of the $A_x(\tau,\theta)$. In

this optimization equation 5.4 tries to adjust the passband of the lowpass kernel to cover the autoterms. On the other hand, 5.5 limits the volume of the kernel to $\alpha$ to exclude the crossterms, which are located away from the origin.

TF distributions computed with a signal dependent kernel usually obtains better cross term suppression and higher resolution compared to TF distributions with fixed kernels. However, as pointed out in [89] the design of a single kernel for a signal with multi-components, may cause some compromises. Because no matter how it is adapted, single kernel can not be optimum for all components locally. Therefore, the kernel also needs to be adapted to the local feature of the signal. A locally adaptive method which applies the smoothing on each slice of Wigner distribution, is developed by Özdemir and Arıkan [90]. The method uses the relation between Radon ambiguity function [91] and fractional Fourier transform [38, 39]. In the next section, a signal dependent kernel is estimated with an initial coarse TF estimate and lifted projection onto epigraph set of $l_1$ cost function.

## 5.2 Kernel Estimation by Projection onto the Epigraph Set of $l_1$ Norm

As discussed in Section 3.2, if we filter AF with a fixed circular mask or kernel around the origin and inverse transform the result we will get a TF with cross terms removed or attenuated. The size of mask should be small enough for a complete cross term removal. But the resolution will be reduced as a result of low-pass filtering or smoothing. Similar result will be obtained with Gaussian kernel given in (5.2). Figure 5.1 shows the result of smoothing with a circular kernel having radius $r_0 = N/16$ and Gaussian kernel as in (5.2) having $\sigma(\phi) = N/16$. Although the Gaussian kernel produces better results, in terms of cross term removal, both are away from the model we desire. Therefore, a kernel matched to the local features of the signal needs to be found. One way is to find the solution to the optimization problem in (5.4). In this chapter a new efficient

approach, will be presented to estimate the signal dependent kernel. In many signal processing applications an initial rough processing reveals some important features of the analyzed signal, and this will ease the subsequent processing. In fact, the method explained in Section 3.2 also uses an initial rough masking with a circular kernel and then the subsequent $l_1$ minimization in TF plane further improves the result. In the following sections this approach will be utilized for kernel estimation. In other words the kernel will be estimated with an initial coarse masking and a subsequent projection onto epigraph set of $l_1$ cost function.

The fixed kernel TF distributions shown in Figure 5.1 are, in a sense, noisy and smeared versions of the model TF we are looking for. Therefore, considering localization and spurious structures, they still do not obey the constraints and need to be regulated. The way of regularization is to project these TF distributions onto some set which contains members having desired constraints. In fact the initial masking with this kernel is also a projection. That is, the projection onto set of TF distributions whose AF values outside the mask are zero. But the problem with this projection is that we need to define the type (e.g. ,circular, Gaussian etc.) and the size $r$ or $\sigma$ of the kernel. Although the standard deviation $\sigma$ in optimization problem (5.4) is allowed to vary radially, the total size of the kernel is set to be confined to a predefined region by (5.5) and the shape at each angle $\phi$ is set to be Gaussian. The main benefit of the projection onto the epigraph set of $l_1$ norm, which was used in Section 3.2 is obvious at this point. It allows the definition of the set without any predefined parameters.

The new kernel design idea is similar to de-noising applications,in which the noisy signal is projected onto a set of signals having desired properties or constraints. For example, a smooth signal contaminated with white noise is projected onto the set of signals whose Total Variation (TV) is below some predefined value [61]. But the TV threshold needs to be defined [61]. In [69] a de-noising method is proposed, which does not require any predefined threshold, using projection onto the epigraph set of TV cost function.

The same approach will be used here. Our aim is to get a localized solution. Therefore, the projection onto epigraph set of $l_1$ cost function will be used.

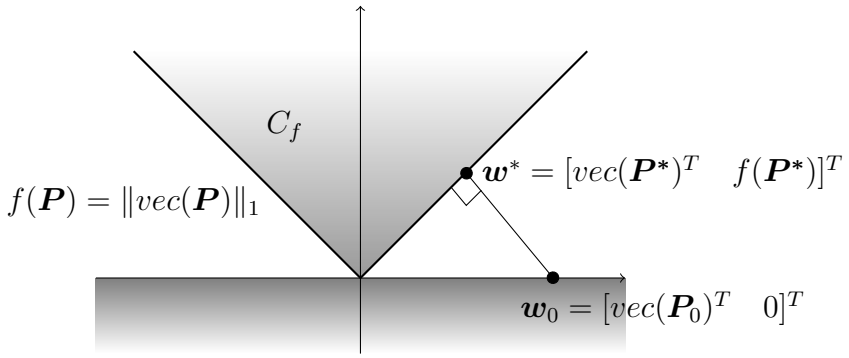


Figure 5.2: Graphical representation of the de-noising process using projection onto the epigraph set of $l_1$ cost function, where $vec(\boldsymbol{P}) \in \boldsymbol{R}^{N^2}$ and $\boldsymbol{w} = [vec(\boldsymbol{P})^T \quad v]^T \in \boldsymbol{R}^{N^2+1}$ in the lifted domain

Let us start with an initial TF distribution $\boldsymbol{P}_0$ that we obtained using a fixed kernel,which is defined as follows:

$$\boldsymbol{P}_0 = \mathcal{F}\{\boldsymbol{A}_x \bullet \boldsymbol{\Phi}_0\}, \tag{5.6}$$

where $\bullet$ is the entry-wise or Hadamard product, which is explained in Appendix C, and $\boldsymbol{\Phi}_0$ is the initial fixed kernel, then the projected TF distribution onto epigraph set of a cost function will be obtained by solving the following optimization problem:

$$\boldsymbol{P}^* = \min_{\boldsymbol{P} \in C_f} \|vec(\boldsymbol{P}) - vec(\boldsymbol{P}_0)\|_2^2. \tag{5.7}$$

The projection is illustrated in Figure 5.2. The projection operation in 5.7 is explained for a general convex cost function in Appendix A.1. The epigraph set of $l_1$ cost function is given by,

$$C_f = \{\boldsymbol{w} = [vec(\boldsymbol{P})^T \quad v]^T \in \boldsymbol{R}^{N^2+1} \quad | \quad f(\boldsymbol{P}) = \|vec(\boldsymbol{P})\|_1 \leq v\} \tag{5.8}$$

The projection $\boldsymbol{P}^*$ in (5.7) will give us a localized TF distribution corresponding to $\boldsymbol{P}_0$. The projection results for two example signals are shown in Figure 5.3 and 5.4 receptively. From these figures it is obvious that, the results are over localized and do not represent the actual signal. But still the projection results in Figures 5.3 and 5.4 contain some information related to the original signal. What is obtained is the most localized part of the signal. If we get the AF function corresponding to the epigraph projection with inverse Fourier transform

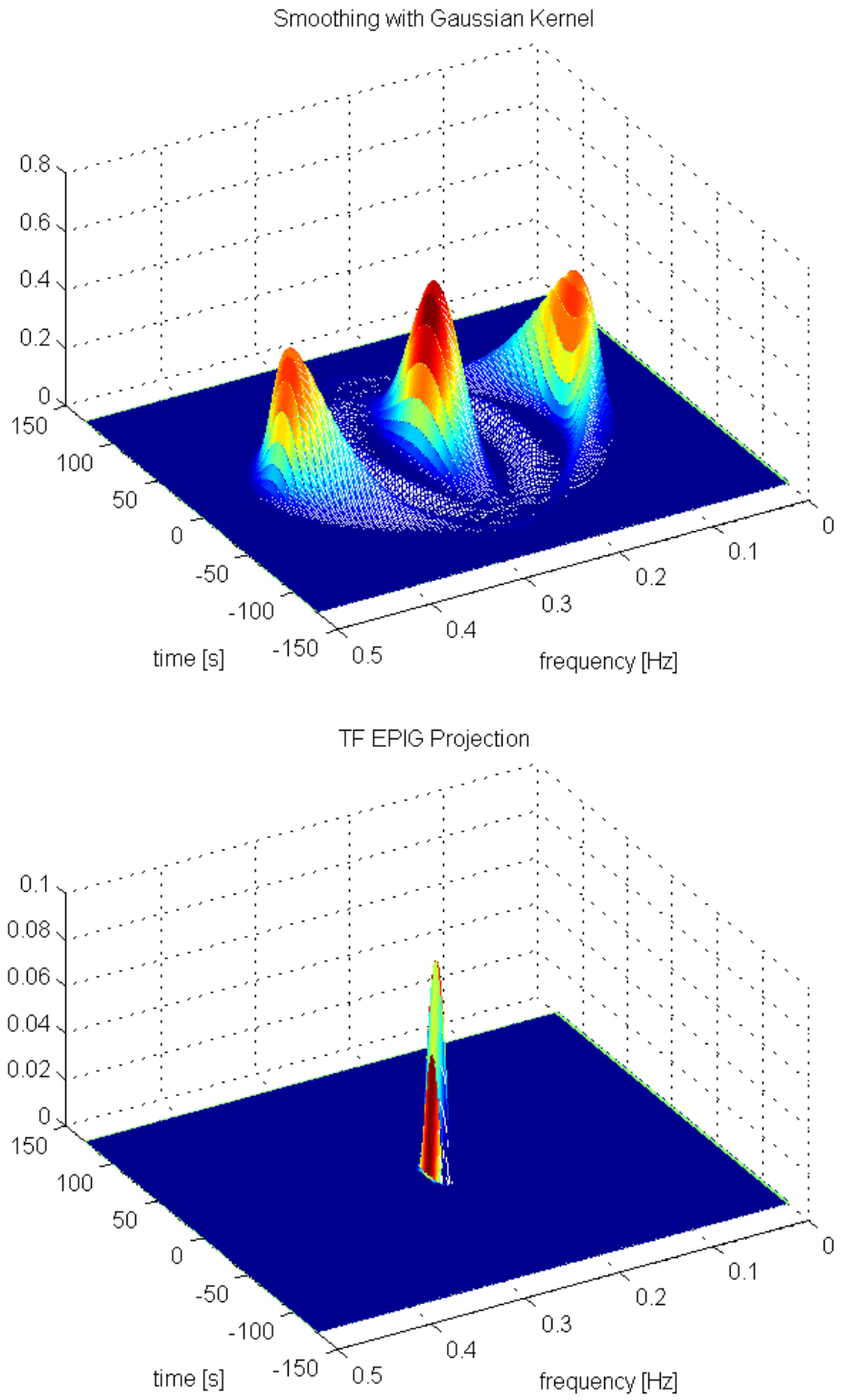


Figure 5.3: Projecting the initial smoothed TF with Gaussian kernel onto the epigraph set of $l_1$ function results in an over localized solution which is not acceptable as the TF distribution corresponding to the original signal.

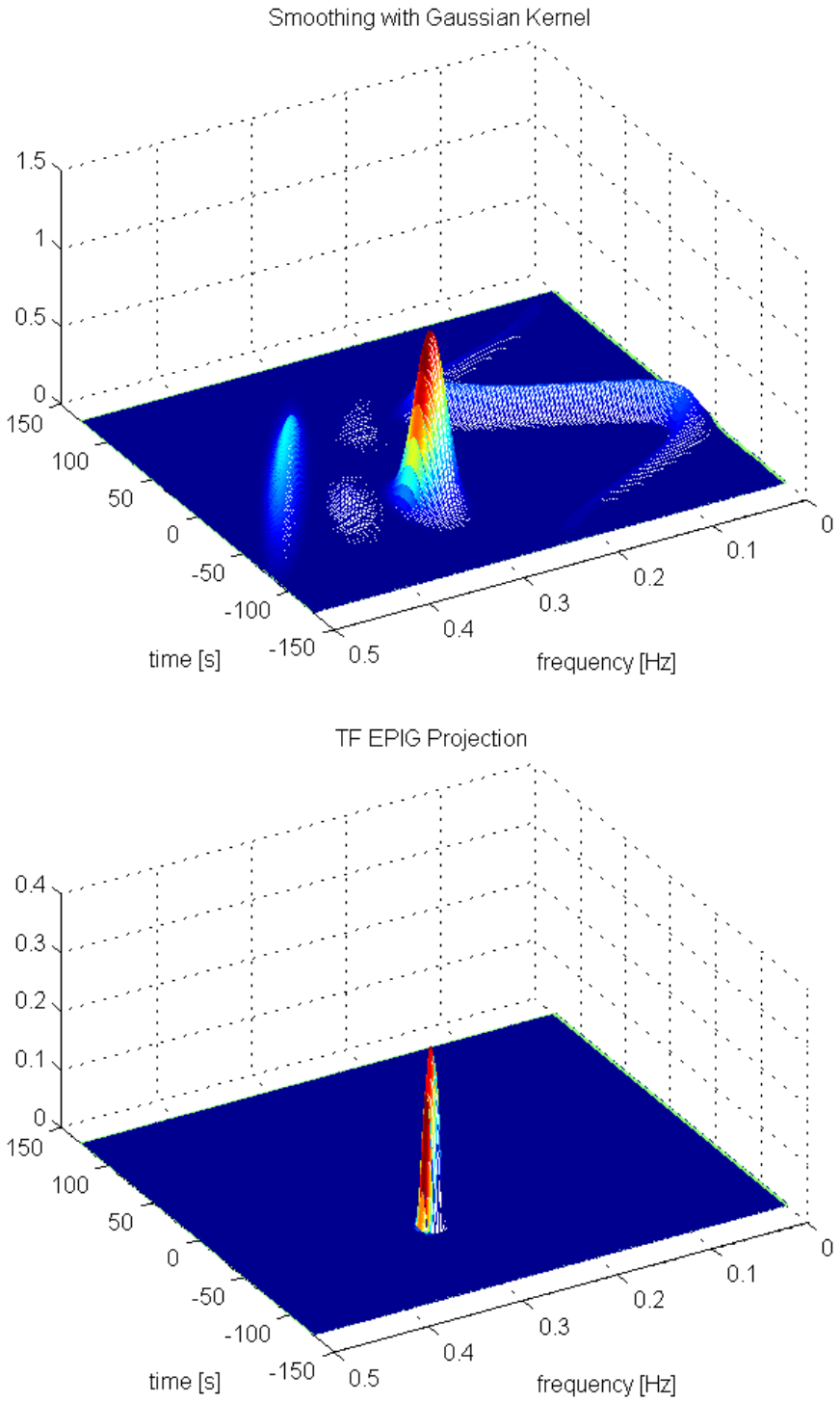


Figure 5.4: Projecting the initial smoothed TF with Gaussian kernel onto the epigraph set of $l_1$ function results in an over localized solution which is not acceptable as the TF distribution corresponding to the original signal.

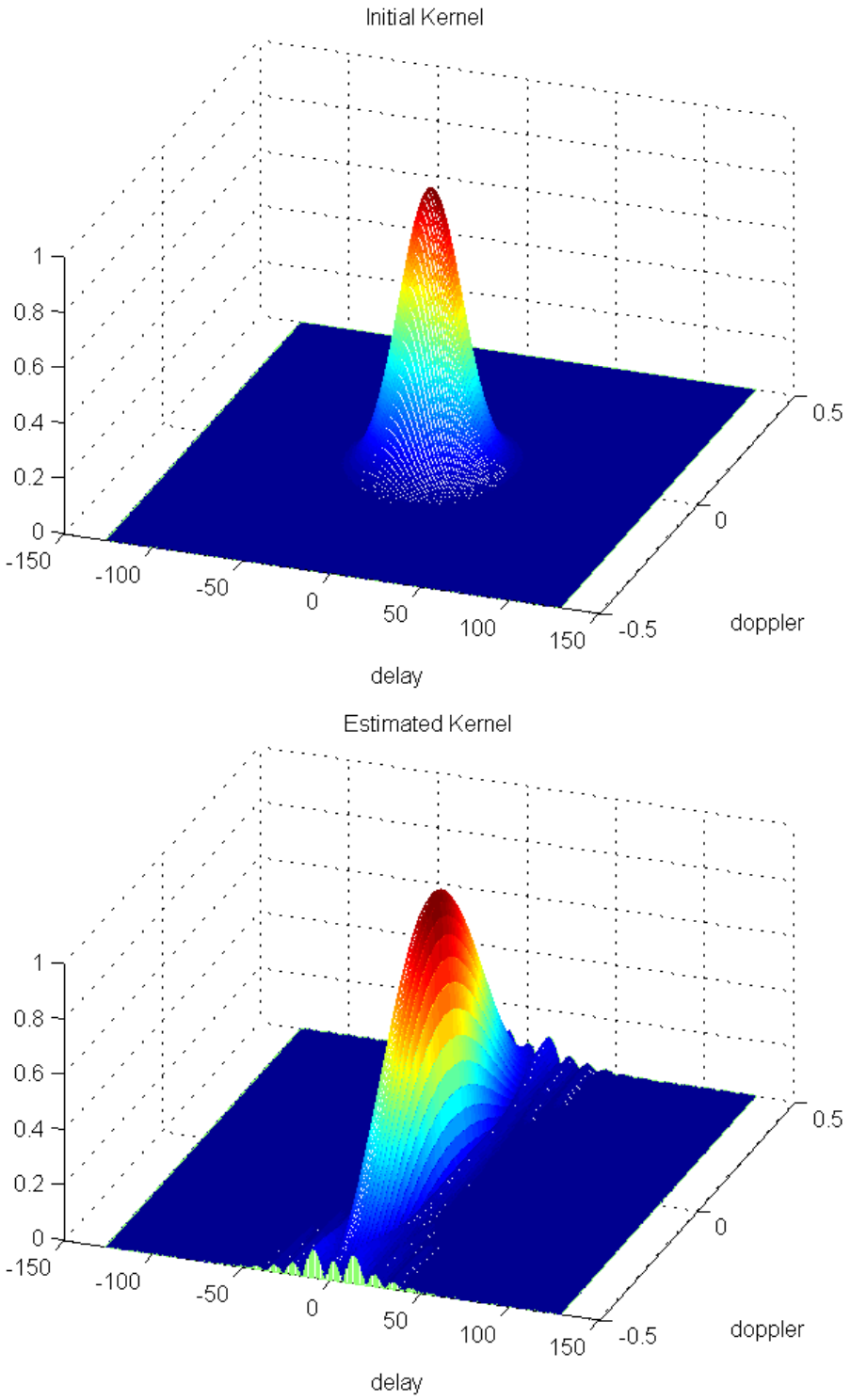


Figure 5.5: The initial Gaussian kernel with $\sigma = N/16$ (top), $N$ being the signal length, and the estimated kernel from initial smoothed TF and epigraph projection (bottom). While the support of initial kernel is circular or has the same $\sigma$ in all directions, the estimated one is aligned in Doppler direction which is in accordance with signal layout in TF plane shown in Figure 5.3 (top)

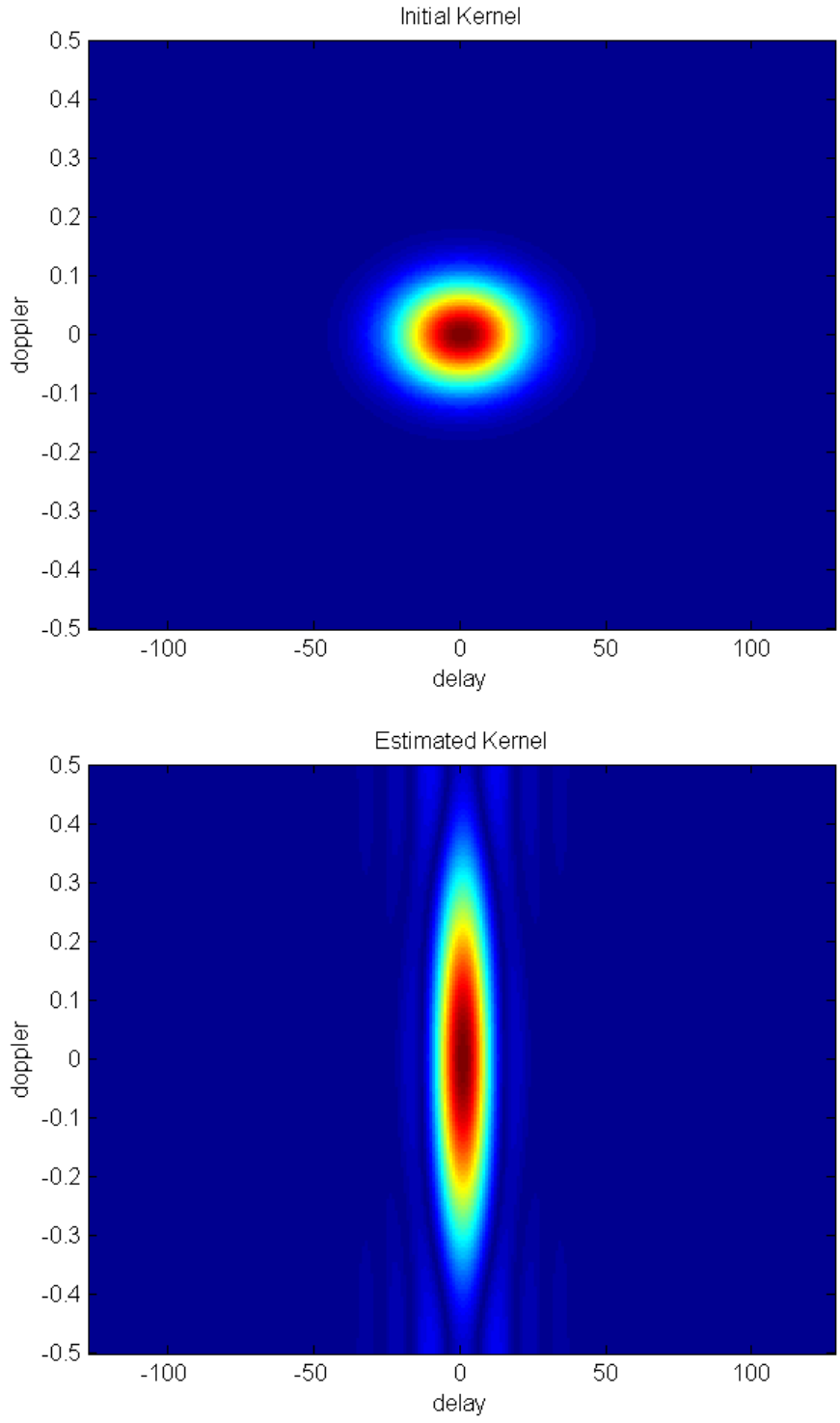


Figure 5.6: The initial Gaussian kernel with $\sigma = N/16$ (top), $N$ being the signal length, and the estimated kernel from initial smoothed TF and epigraph projection (bottom). While the support of initial kernel is circular or has the same $\sigma$ in all directions, the estimated one is aligned in Doppler direction which is in accordance with signal layout in TF plane shown in Figure 5.3 (top)

and normalize with its maximum magnitude we will see that it gives us a signal dependent kernel estimate. In Figure 5.5 the 3D plot of the initial fixed Gaussian kernel with $\sigma = N/16$ ,where $N$ being the signal length, and the kernel, estimated from initial smoothed TF distribution and epigraph projection, is shown. The kernel is estimated from the example signal shown in the top part of Figure 5.3. The top view of the kernels are shown in Figure 5.6. While the support of initial kernel is circular or has the fixed $\sigma$ in all directions, the estimated one is aligned in Doppler direction which is in accordance with the signal layout in TF plane as shown in the top part of Figure 5.3.

If we use the resultant kernel for masking the original AF we will observe that a localized TF will be obtained with reduced cross-terms. This is better verified with example signals. The TF distributions obtained with this method for two example signals are compared to the other methods in Figures 5.7 and 5.8, respectively. The TF with optimized kernel is the distribution, smoothed with the kernel obtained from optimization problem in (5.4). From these figures, we observe that the method successfully removes the cross-terms and results in high resolution. The results are comparable to the optimized Gaussian kernel given in (5.4). Therefore, the algorithm in Table 5.1 is proposed based on above observation.

One important observation with the estimated kernel is that some spurious structures are observed together with the main kernel lobe in AF plane. These structures are dependent on size of the initial kernel $\mathbf{\Phi}_0$ and the signal. They are shown in the top part of Figure 5.9 and 5.10. These spurious parts are observed for some cases and have an unwanted effect in final smoothed TF with the estimated kernel. Therefore, we need to eliminate these parts. It was observed that, they can be eliminated by selecting the initial kernel $\mathbf{\Phi}_0$ in AF domain with an even smaller area ($r < N/16$). It was also observed that, without changing the initial fixed kernel, they can also be removed by applying a second Gaussian mask to the initial TF distribution $\boldsymbol{P}_0$ obtained by fixed kernel.

The selection of smaller initial kernel in AF domain, will result in an over smoothed and smeared initial $\boldsymbol{P}_0$ in TF plane. This in turn, when projected onto

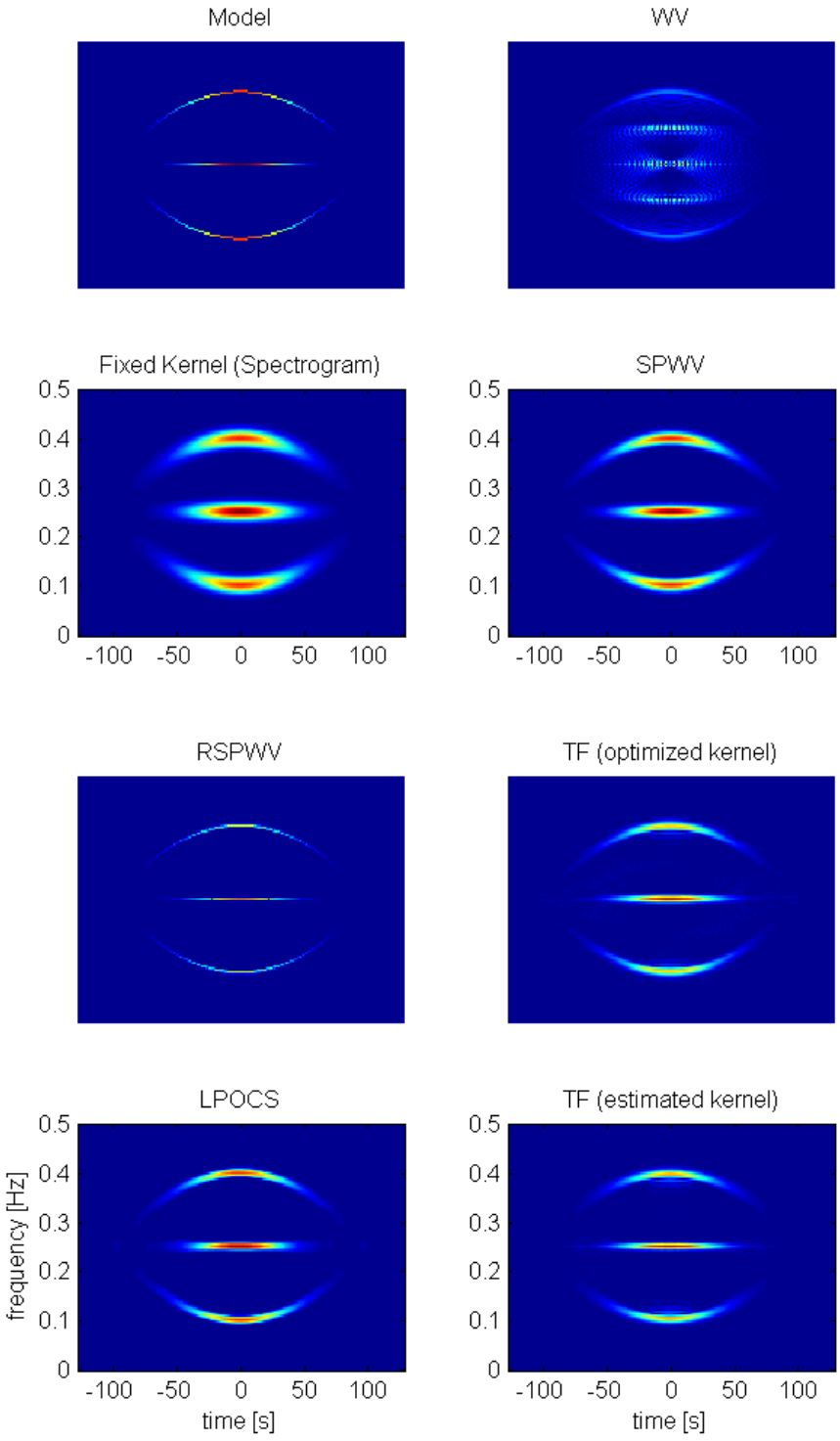


Figure 5.7: Comparison of TF smoothing with estimated kernel to other methods for the Example signal 1: left column: the ideal model, fixed kernel or Spectrogram(SP), reassigned SPWV (RSPWV), lifted POCS, right column: WV distribution, the Smoothed Pseudo WV (SPWV) distribution,TF with optimal kernel($\alpha = 1.4$), TF with the estimated kernel.

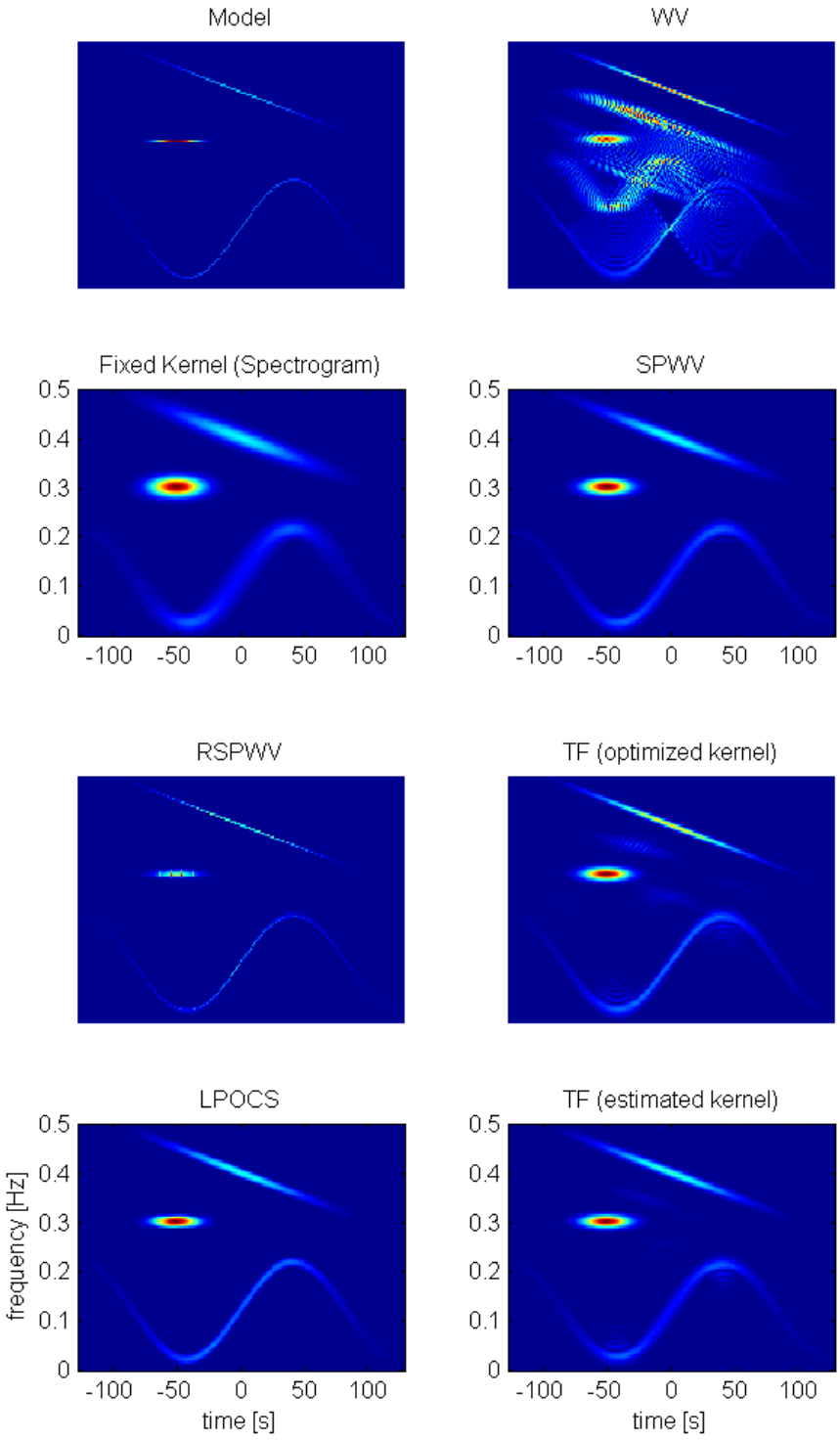


Figure 5.8: Comparison of TF smoothing with estimated kernel to other methods for the Example signal 2: left column: the ideal model, fixed kernel or Spectrogram(SP), reassigned SPWV (RSPWV), lifted POCS, right column: WV distribution, the Smoothed Pseudo WV (SPWV) distribution,TF with optimal kernel($\alpha = 1.4$), TF with the estimated kernel.

Table 5.1: Signal dependent kernel estimation steps and smoothing

| **Step** | **Computation** | **Action** |
|---|---|---|
| 1 | $\boldsymbol{A}_0 = \boldsymbol{A}_x \bullet \boldsymbol{\Phi}_0$ | Initial masking with a fixed kernel $\boldsymbol{\Phi}_0(r,\phi) = exp(\frac{-r^2}{2\sigma_0^2})$ $\sigma_0 = N/16$, in the AF domain |
| 2 | $\boldsymbol{P}_0 = \mathcal{F}\{\boldsymbol{A}_0\}$ | Compute the Fourier transform |
| 3 | $\boldsymbol{P}^* = \min_{\boldsymbol{P}\in C_f} \|vec(\boldsymbol{P}) - vect(\boldsymbol{P}_1)\|_2^2$ | Projection onto epigraph set of $l_1$ cost function |
| 4 | $\boldsymbol{A}^* = \mathcal{F}^{-1}\{\boldsymbol{P}^*\}$ | AF computation |
| 5 | $\tilde{\boldsymbol{\Phi}}_x = \frac{\lvert\boldsymbol{A}^*\rvert}{\max_{k,l}\lvert\boldsymbol{A}^*[k,l]\rvert}$ | Normalization |
| 6 | $\tilde{\boldsymbol{P}}_x = \mathcal{F}\{\tilde{\boldsymbol{\Phi}}_x \bullet \boldsymbol{A}_x\}$ | TF smoothing with estimated kernel |

epigraph set and transformed back into AF domain, will have a similar compactness effect on resultant AF domain kernel. Even though there is a projection in this there-step process and it is a non-linear operation, still the effect of uncertainty principle is observed. This way the spurious structures will be removed.

It was observed that using a second Gaussian mask in TF domain before projection is much more effective and also produces slightly better results. Also, as it was stated, this spurious structures are not observed for all the signals. The results with and without second Gaussian mask in TF plane are shown in Figure 5.9 and their top views are shown in Figure 5.10 for the example Signal 3. This second Gaussian mask is also a fixed one and does not to be fine tuned or adapted to the signal. Its peak is at the center of TF plane with standart deviation $\sigma_p = N/4$. The shape of this Gaussian mask is shown in Figure 5.11. With this TF plane masking, the overall method is listed in Table 5.2. The estimated and the optimized kernels for an example signal are shown in Figure 5.12. Their top views are shown in Figure 5.13. From this figure we observe that the kernel

Table 5.2: Signal dependent kernel estimation steps with a pre-filter

| **Step** | **Computation** | **Action** |
|---|---|---|
| 1 | $\boldsymbol{A}_0 = \boldsymbol{A}_x \bullet \boldsymbol{\Phi}_0$ | Initial masking with a fixed kernel $\boldsymbol{\Phi}_0(r,\phi) = exp(\frac{-r^2}{2\sigma_0^2})$ $\sigma_0 = N/16$, in the AF domain |
| 2 | $\boldsymbol{P}_0 = \mathcal{F}\{\boldsymbol{A}_0\}$ | Compute the Fourier transform |
| 3 | $\boldsymbol{P}_1 = \boldsymbol{G} \bullet \boldsymbol{P}_0$ | Masking the initial TF with a second Gaussian mask shown in Figure 5.11 |
| 4 | $\boldsymbol{P}^* = \min_{\boldsymbol{P} \in C_f} \|vec(\boldsymbol{P}) - vect(\boldsymbol{P}_1)\|_2^2$ | Projection onto epigraph set of $l_1$ cost function |
| 5 | $\boldsymbol{A}^* = \mathcal{F}^{-1}\{\boldsymbol{P}^*\}$ | AF computation |
| 6 | $\tilde{\boldsymbol{\Phi}}_x = \frac{\lvert\boldsymbol{A}^*\rvert}{\max_{k,l} \lvert\boldsymbol{A}^*[k,l]\rvert}$ | Normalization |
| 7 | $\tilde{\boldsymbol{P}}_x = \mathcal{F}\{\tilde{\boldsymbol{\Phi}}_x \bullet \boldsymbol{A}_x\}$ | TF smoothing with estimated kernel |

estimated by using the projection onto the epigraph set of $l_1$ norm at bottom is comparable to the one obtained by solving optimization problem in (5.4) on top.

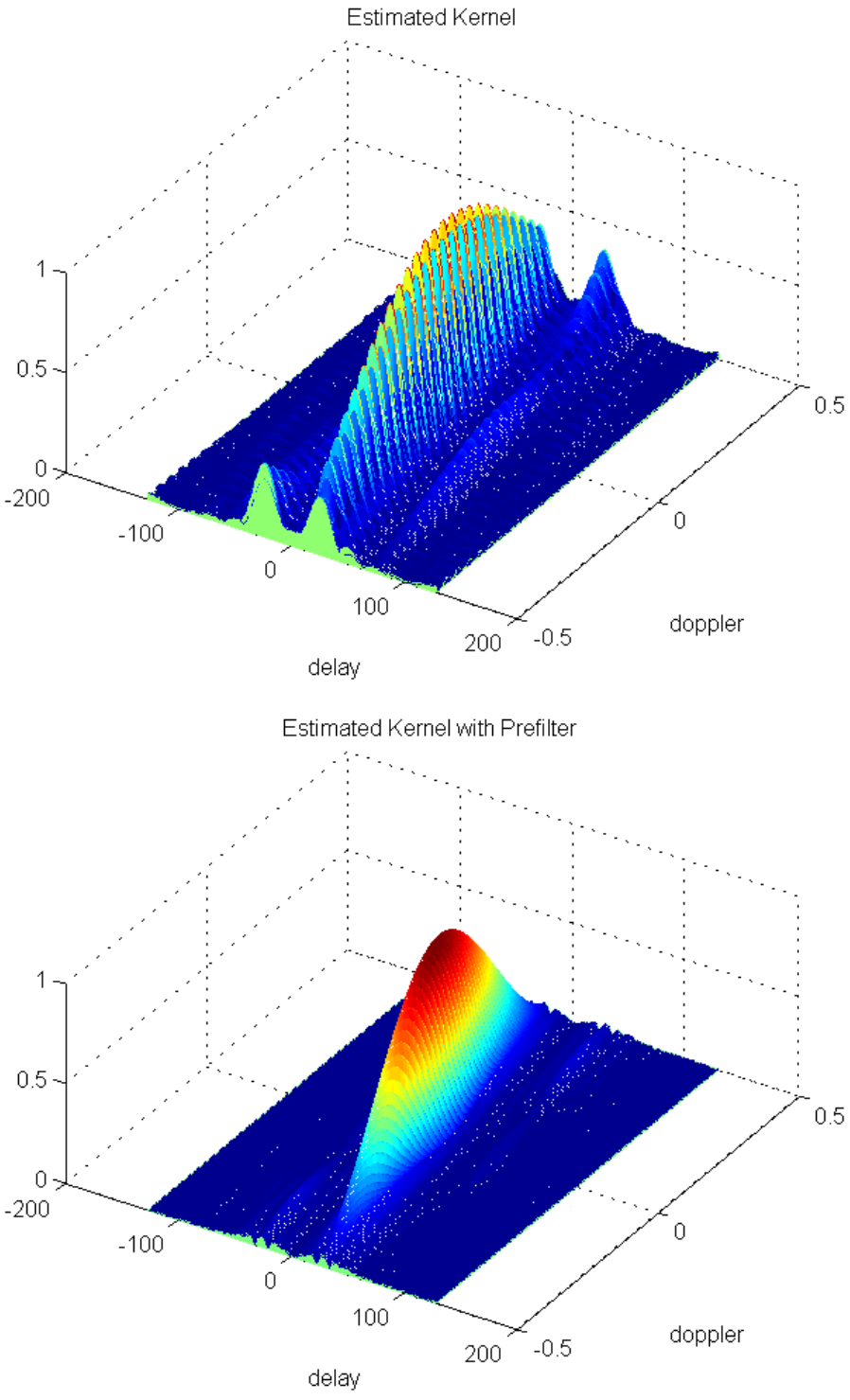


Figure 5.9: The estimated kernel has spurious structures due to initial coarse low-pass filter. In order to remove them a a Gaussian mask is applied to the initial smoothed TF distribution before epigraph projection. The estimated kernel without Gaussian mask is shown on top and the estimation with Gaussian mask is shown at bottom.

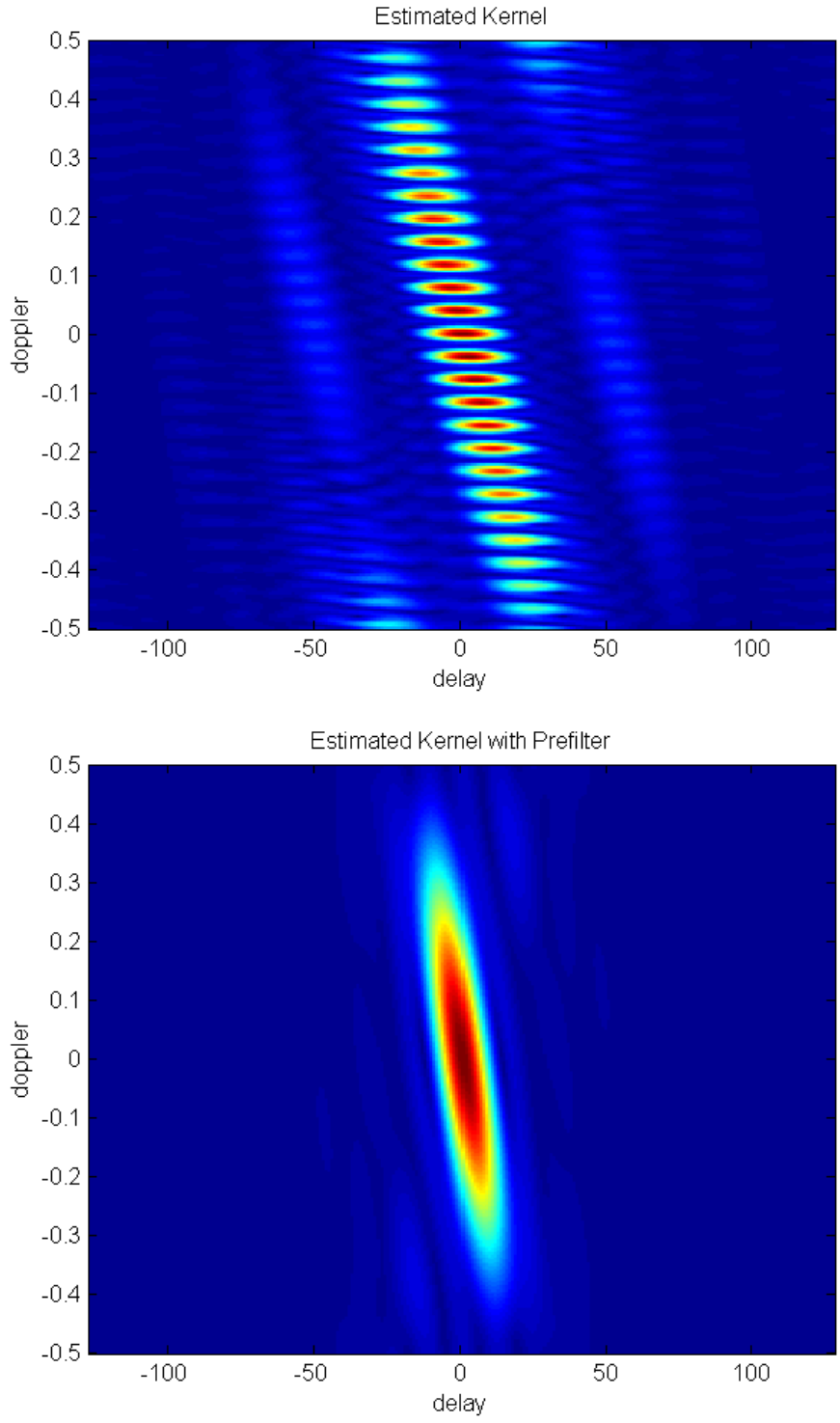


Figure 5.10: The estimated kernel has spurious structures due to initial coarse low-pass filter. In order to remove them a Gaussian mask is applied to the initial smoothed TF distribution before epigraph projection. The estimated kernel without Gaussian mask is shown on top and the estimation with Gaussian mask is shown at bottom.

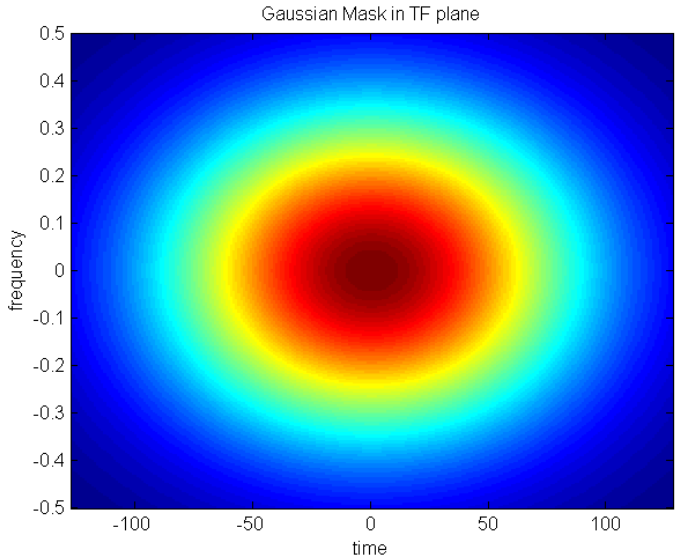


Figure 5.11: The support of the second Gaussian mask applied to initial TF at Step 2 of Table 5.1 (red corresponds to 1 and blue color corresponds to zero).

## 5.3 Simulation Results

Results for four other examples are shown in Figure 5.14 - 5.17. Example Signal 5 in Figure 5.16 is the TF distribution for Frequency Hopping /Mary Frequency-Shift-Keyed (FH/MFSK) signal. Example Signal 6 in Figure 5.17 is the TF of a short segment from a dolphin's click signal. In all tested examples, the initial Gaussian smoothing kernel $\mathbf{\Phi}_0$ was selected with $\sigma = N/16$, where $N$ being the signal length. It was experimentally observed that $\sigma$ can be safely set to any value between $N/36 \leq \sigma \leq N/14$ and does not need to be fine tuned. This range was seen to be sufficient for initial cross-term removal. The second Gaussian mask $\boldsymbol{G}$ in TF plane was selected with $\sigma_p = N/4$. In general, to get good results for all the example signals at hand, the SPWV time smoothing window length was set to an odd integer closest to $N/10$. The length of frequency smoothing filter at time domain was set to an odd integer closest to $N/4$. In this way any parameter adaptation to the signal was avoided. The similarity and the localization features for the method are compared to several other methods discussed in Section 3.2 and the results are given in Tables 5.3 and 5.4, respectively. In these tables we

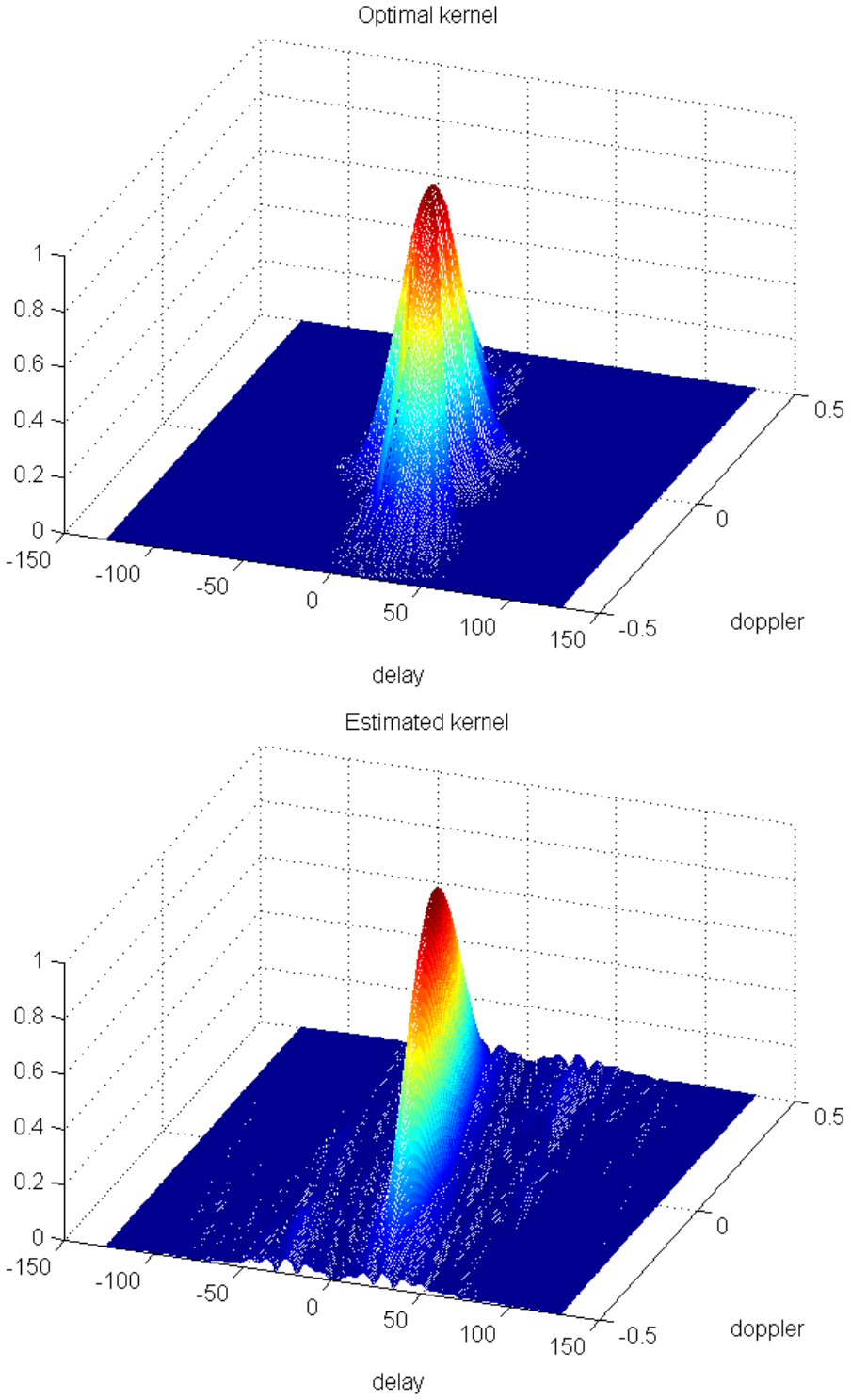


Figure 5.12: The optimized kernel (top) and the estimated kernel (bottom).

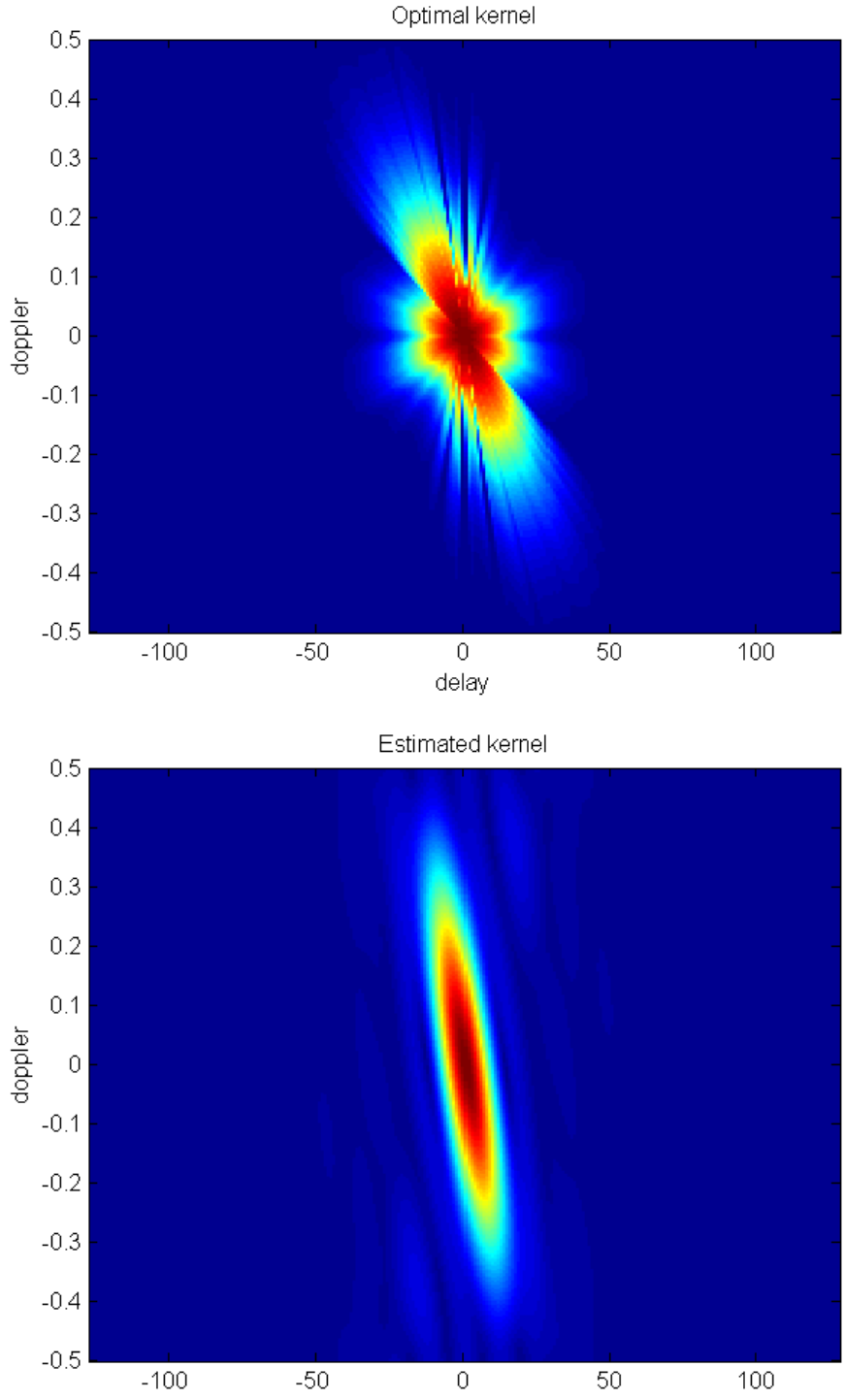


Figure 5.13: The optimized kernel (top) and the estimated kernel (bottom).

also see that the proposed method generates, in general, a localized and cross term free TF distribution. From visual inspection and quantitative comparison it is observed that the method is comparable to the optimized kernel based smoothing obtained by solving the optimization in (5.4) but slightly worse than LPOCS method. As explained in Section 3.2, the RSPWV distribution has better localization but is not the best one in overall assessment in terms of similarity and localization. In this respect LPOCS and RSPWV are comparable. As discussed in Section 5.1, although adapted to the signal, a single kernel can not be optimum for all components of a multi-component signal. This is observed in example Signal 5 in Figure 5.16 both for optimized kernel and estimated kernel. Therefore in this respect the LPOCS is better than all other methods, in terms of localization, cross-terms removal and adaptability to local features. On the other hand, in terms of computational cost the proposed kernel estimation based-method is faster and efficient than LPOCS because it does not require successive projections onto convex sets.

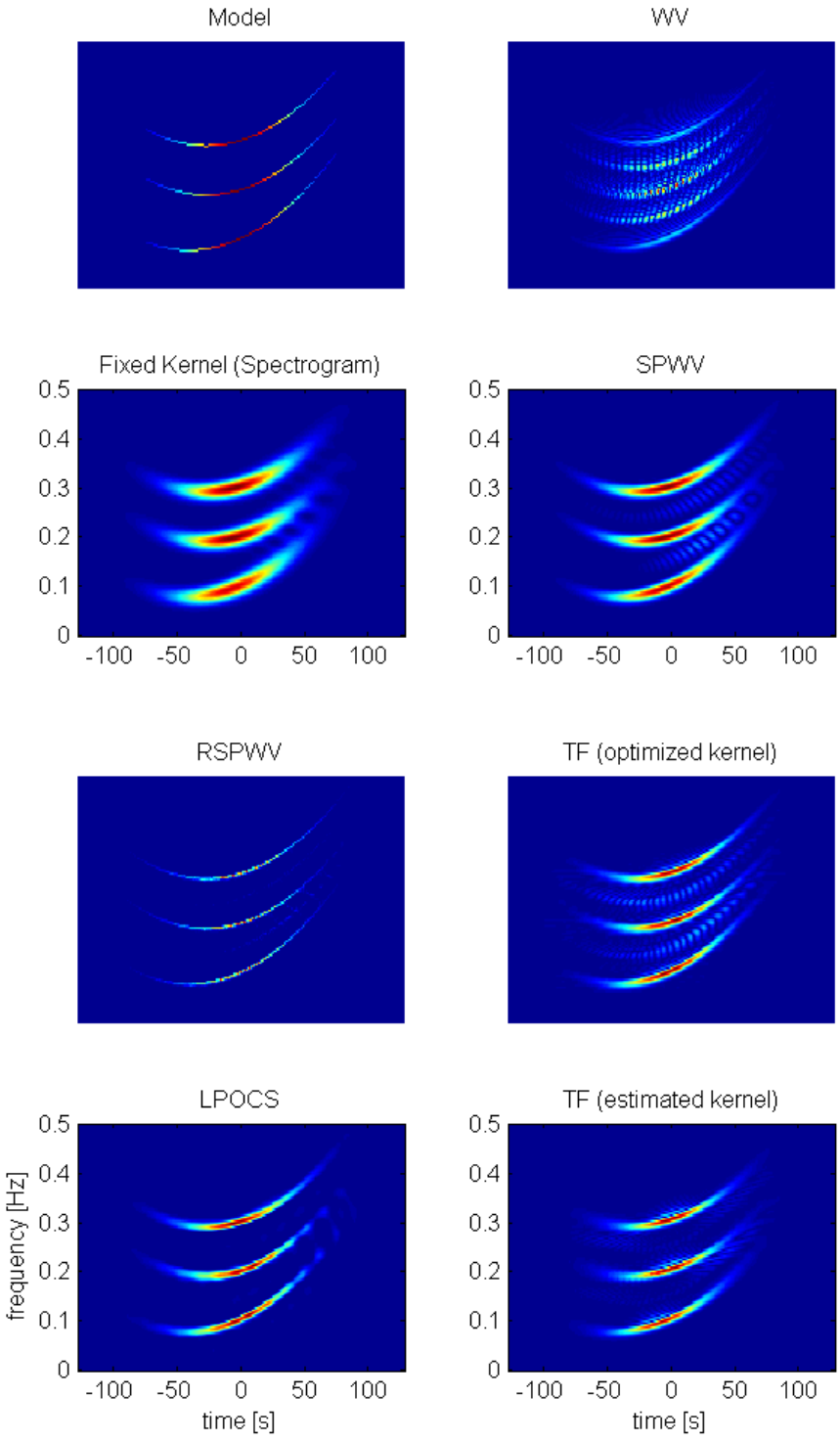


Figure 5.14: Comparison of TF smoothing with estimated kernel to other methods for the Example signal 3: left column: The ideal model, fixed kernel or Spectrogram(SP), reassigned SPWV (RSPWV), lifted POCS, right column: WV distribution, the Smoothed Pseudo WV (SPWV) distribution,TF with optimal kernel($\alpha = 1.4$), TF with the estimated kernel.

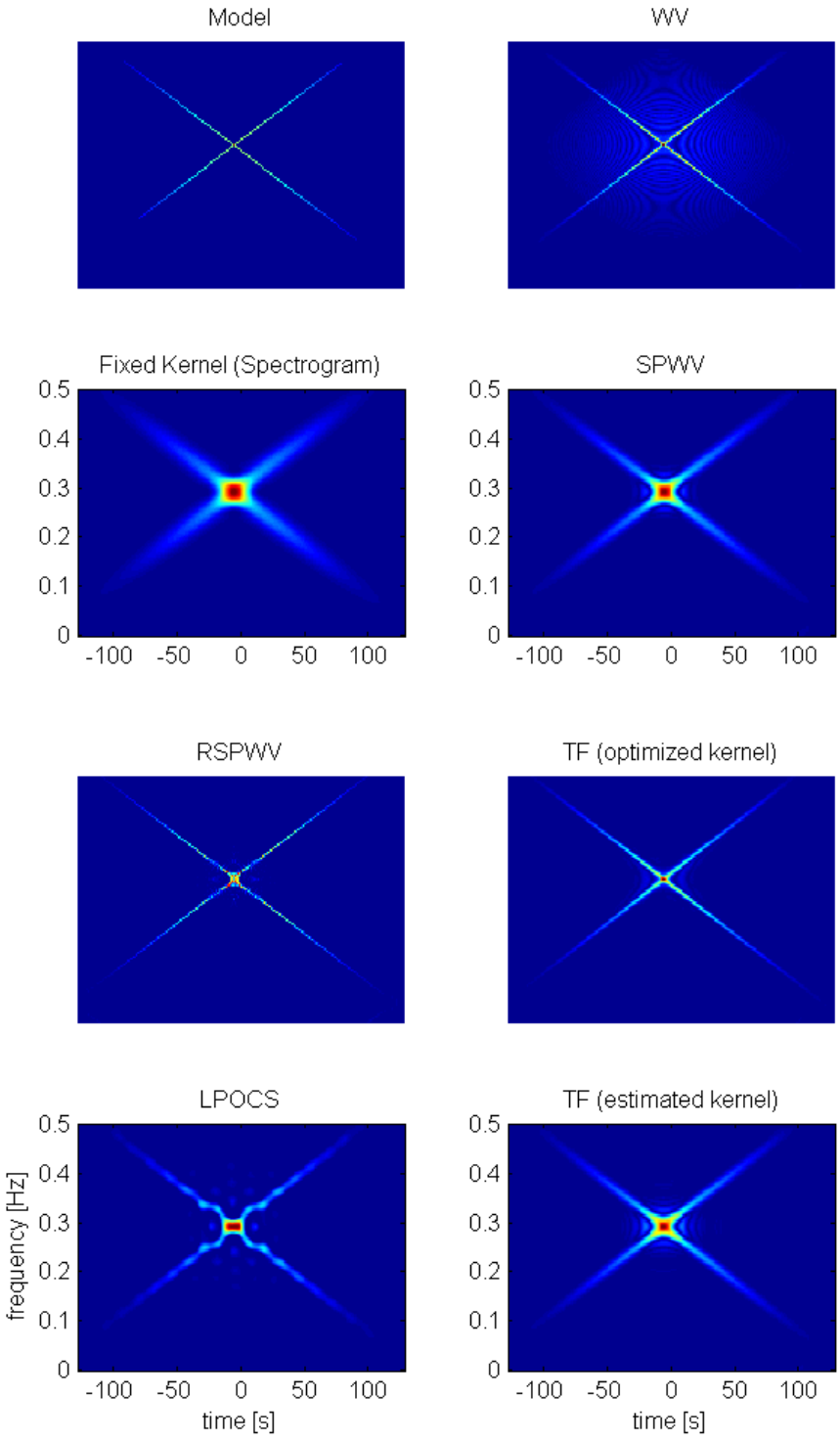


Figure 5.15: Comparison of TF smoothing with estimated kernel to other methods for the Example signal 4: left column: The ideal model, fixed kernel or Spectrogram(SP), reassigned SPWV (RSPWV), lifted POCS, right column: WV distribution, the Smoothed Pseudo WV (SPWV) distribution,TF with optimal kernel($\alpha = 1.4$), TF with the estimated kernel.

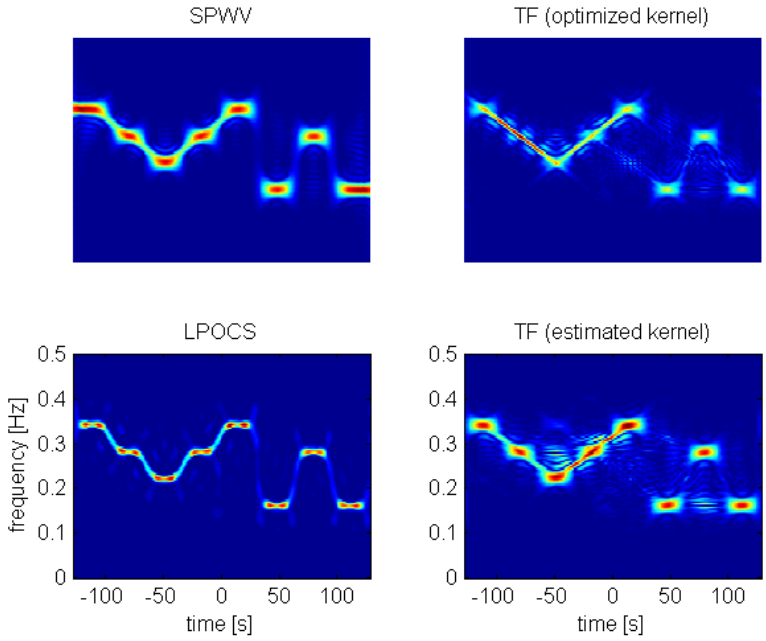


Figure 5.16: Comparison of TF smoothing with estimated kernel to other methods for the Example signal 5: left column: The result with Smoothed Pseudo WV (SPWV), lifted POCS, right column: TF with the optimal kernel, TF with the estimated kernel.

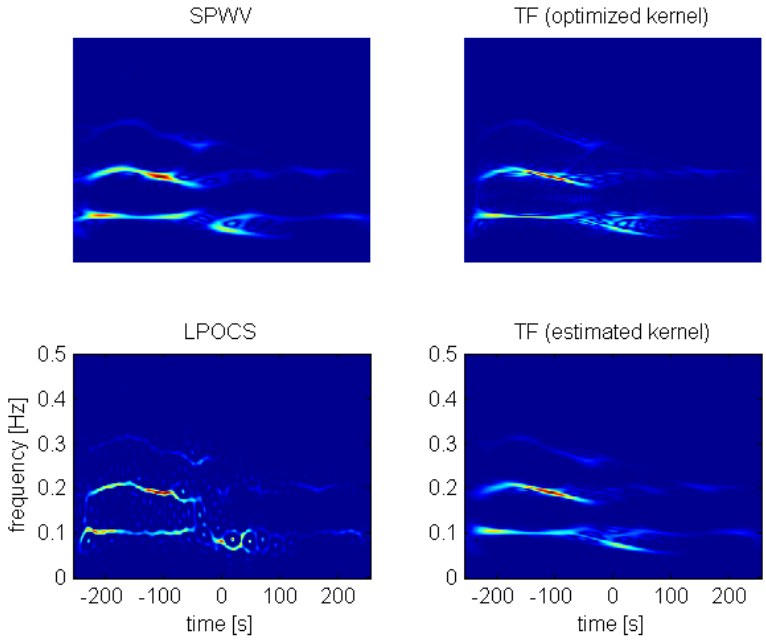


Figure 5.17: Comparison of TF smoothing with estimated kernel to other methods for the Example signal 6: left column: The result with Smoothed Pseudo WV (SPWV), lifted POCS, right column: TF with the optimal kernel, TF with the estimated kernel.

Table 5.3: Pearson correlation coefficient between TF distributions and the model TF for tested signal examples. A higher value shows better similarity to the model.

| | **Pearson correlation coefficient** | | | | |
|---|---|---|---|---|---|
| **Signal** | **Model** | **RSPWV** | **LPOCS** | **TFEK** | **TFOK** |
| Signal 1 | 1 | **0.73** | 0.59 | 0.60 | 0.60 |
| Signal 2 | 1 | **0.52** | 0.50 | 0.47 | 0.48 |
| Signal 3 | 1 | 0.47 | **0.55** | 0.42 | 0.42 |
| Signal 4 | 1 | 0.42 | 0.40 | 0.44 | **0.57** |
| Signal 5 | 1 | 0.34 | **0.44** | 0.32 | 0.27 |
| Signal 6 | N/A | N/A | N/A | N/A | N/A |

Table 5.4: Rényi entropy of TF distributions for tested signal examples. A lower value indicates better localization.

| | **Rényi entropy** | | | | |
|---|---|---|---|---|---|
| **Signal** | **Model** | **RSPWV** | **LPOCS** | **TFEK** | **TFOK** |
| Signal 1 | 8.13 | **8.47** | 10.14 | 10.13 | 10.19 |
| Signal 2 | 7.46 | **8.74** | 9.99 | 10.31 | 10.15 |
| Signal 3 | 8.24 | **8.82** | 10.35 | 10.37 | 10.40 |
| Signal 4 | 8.04 | **8.57** | 9.65 | 10.10 | 9.41 |
| Signal 5 | 7.99 | **9.45** | 10.23 | 11.12 | 10.69 |
| Signal 6 | N/A | **10.58** | 11.70 | 12.37 | 11.88 |

# Chapter 6

# Mixed TF and Parametric Component Estimation for Time-varying Signals

## 6.1 Introduction

In many practical signal applications involving amplitude and/or phase-modulated carrier signals, we encounter discrete-time signals which can be represented as

$$s[n] = a[n]e^{j\phi[n]}, \tag{6.1}$$

where $a[n]$ and $\phi[n]$ are the real amplitude and phase functions, respectively. Such signals are common in radar [3], sonar applications [8], and in many other natural problems [23]. A multi-component signal is a linear combination of these types of signals and is given by

$$s[n] = \sum_{i=1}^{L} a_i[n]e^{j\phi_i[n]}, \tag{6.2}$$

where $s_i[n] = a_i[n]e^{j\phi_i[n]}$ is the $i^{th}$ component and $L$ is the number of components. Clearly, the linear decomposition of the multi-component signal in terms

of such components is not unique. Some other restrictions should be put on the components to have a unique decomposition [23]. In general, a component is the part of the multi-component signal which is identifiable in time, in frequency, or in mixed time-frequency plane. Therefore, we will assume that the different components are well separated in time-frequency plane and have a small instantaneous bandwidth compared to separation between components. In this regard, the components, as discussed in Section 3.2 have sparse representation in TF plane.

The main problem is to separate the components from each other or to recover one of the components knowing that the components are disjoint and sparse on the TF plane. In general the approaches for the solution are those which use nonparametric time-frequency methods and those of parametric ones. In case where the desired signal component is separable or disjoint in one of time or frequency domain, then, with some sort of time or frequency masking, the component can be estimated. When the signals are disjoint neither in time nor in frequency domain, then time-frequency processing methods are needed for component separation. But, in some cases, even though the components are not separated in time or in frequency, the fractional Fourier Transform [38, 39, 92] can be used to separate the components at a fractional domain, where they are disjoint.

Time frequency distribution- (TFD) based waveform reconstruction techniques try to recover a time-domain signal whose distribution is close to a valid TFD. The method in [22], synthesizes a time-domain signal from its bilinear TFD.

The well-known time-frequency method is the Wigner-Distribution [21] based signal synthesis [8,22,24,93]. The main drawback related to time-frequency methods is the cross-terms and resolution of the time-frequency representations [19]. Therefore, there have been many efforts to obtain cross-term-free and high-resolution TFDs [89, 94, 95]. With the methods proposed in Section 3.2 and Section 5.2 high resolution cross-term free TFDs are obtained.

In parametric model a signal or component is represented as a linear combination of some known basis functions [34,96], and the component parameters are estimated. In many radar and sonar applications the polynomials are good basis functions.

If the phase and amplitude functions in (6.2) are polynomials and amplitude function is constant or slowly varying, the Polynomial Phase Transform (PPT) [96,97] is a practical tool for parameter estimation. While the method is practical, it has difficulties in time-varying amplitude and multi-component cases [98]. It is also suboptimal because the components are extracted in a sequential manner.

Another solution is the ML estimation of the parameters. The related method is explained in [34, 98]. The ML estimation of the parameters requires a multivariable nonlinear optimization problem to be solved. Therefore, the solution requires iterative techniques like nonlinear conjugate gradient (NL-CG) or quasi-Newton-type algorithms and is computationally intensive [34, 98]. Another requirement is a good initial estimate which avoids possible local minima. But it estimates all parameters as a whole and is optimal in this respect. Also it does not suffer from cross-terms related to time-frequency techniques.

In [96] an algorithm is explained which extracts the components using PPT in a sequential manner. In [99] a mixed time-frequency and PPT-based algorithm is proposed. The examples with the ML approach are given in [34, 98].

In this section a method is proposed which uses ML estimation. Similar to [99], the initial estimates are obtained from time-frequency representation of the multicomponent signal and then all parameters are estimated by ML estimation. Since ML estimation requires large amount of computation, a method is proposed to reduce the computations. The proposed method iterates amplitude and phase parameters separately by assuming that the other is known. The method is different from the ones given in [34, 98], where the amplitude parameters are eliminated analytically and the resultant equivalent cost function is minimized.

Eliminating amplitude parameters analytically results in a cost function which

has less number of parameters. But it is computationally more complex in terms of function and gradient evaluations, which are needed in nonlinear optimization iterations.

Since the cost functions for separate amplitude and phase parameters are less complex, the amount of computation is reduced with the proposed method compared to the case where amplitude parameters are eliminated analytically. Furthermore, by using the proposed method in an expectation maximization loop [100], a better reconstruction error level is obtained. The results are verified with simulation examples.

## 6.2 Problem Formulation and ML Estimation

Let $x[n]$ be a discrete-time process consisting of the sum of a deterministic multi-component signal and additive white Gaussian noise given by

$$x[n] = \sum_{i=1}^{L} a_i[n] e^{j\phi_i[n]} + w[n], \quad n = 0, 1, \cdots, N-1 \tag{6.3}$$

where $w[n]$ is the complex noise process. Denoting $g_k[n]$ and $p_k[n]$ as the real-valued basis functions for amplitude and phase terms, respectively, we have

$$a_i[n] = \sum_{k=0}^{P_i} a_{i,k} g_k[n], \tag{6.4}$$

and

$$\phi_i[n] = \sum_{k=0}^{Q_i} b_{i,k} p_k[n] \tag{6.5}$$

where $a_{i,k}$ and $b_{i,k}$ are the real valued amplitude and phase coefficients for the $i^{th}$ component. Similarly $P_i + 1$ and $Q_i + 1$ are the number of coefficients for amplitude and phase functions of the $i^{th}$ component. In general, basis functions can be any functions which are square integrable and spans the space of real and integrable functions in a given observation interval. Also they can be selected to be different for amplitude and phase and for each component. In this work they are assumed to be polynomial for both amplitude and phase and for all

components. Therefore, $P_i$ and $Q_i$ corresponds to polynomials orders for the amplitude and phase of the ith component, respectively.

Defining the amplitude and phase coefficients of the $i^{th}$ component by the vectors

$$\boldsymbol{a}_i = [a_{i,0} \quad a_{i,1} \quad a_{i,2} \quad \cdots \quad a_{i,P_i}]^T, \tag{6.6}$$

and

$$\boldsymbol{b}_i = [b_{i,0} \quad b_{i,1} \quad b_{i,2} \quad \cdots \quad b_{i,Q_i}]^T \tag{6.7}$$

respectively, we can define parameter vectors for all the components as follows

$$\boldsymbol{a} = [\boldsymbol{a}_1^T \quad \boldsymbol{a}_2^T \quad \cdots \quad \boldsymbol{a}_L^T]^T, \tag{6.8}$$

and

$$\boldsymbol{b} = [\boldsymbol{b}_1^T \quad \boldsymbol{b}_2^T \quad \cdots \quad \boldsymbol{b}_L^T]^T. \tag{6.9}$$

Let the time vector $n$ be defined as

$$\boldsymbol{n} = [0 \quad 1 \quad 2 \quad \cdots \quad N-1]^T. \tag{6.10}$$

The following notation will be used

$$\boldsymbol{x} = x[\boldsymbol{n}] = [\ x[0] \quad x[1] \quad x[2] \quad \cdots \quad x[N-1]\ ]^T, \tag{6.11}$$

$$\boldsymbol{w} = w[\boldsymbol{n}] = [\ w[0] \quad w[1] \quad w[2] \quad \cdots \quad w[N-1]\ ]^T, \tag{6.12}$$

and

$$e^{j\phi_i[\boldsymbol{n}]} = [\ e^{j\phi_i[0]} \quad e^{j\phi_i[1]} \quad e^{j\phi_i[2]} \quad \cdots \quad e^{j\phi_i[N-1]}\ ]^T \tag{6.13}$$

where the bold characters $\boldsymbol{x}$,$\boldsymbol{w}$, and $e^{j\phi_i[\boldsymbol{n}]}$ are all $N \times 1$ vectors. Based on the definitions given in (6.11)-(6.13) the following matrices are defined:

$$\boldsymbol{\Phi}_i = [\ g_0[\boldsymbol{n}] \bullet e^{j\phi_i[\boldsymbol{n}]} \quad g_1[\boldsymbol{n}] \bullet e^{j\phi_i[\boldsymbol{n}]} \quad \cdots \quad g_{P_i}[\boldsymbol{n}] \bullet e^{j\phi_i[\boldsymbol{n}]}\ ] \tag{6.14}$$

and the composite matrix:

$$\boldsymbol{\Phi} = [\ \boldsymbol{\Phi}_1 \quad \boldsymbol{\Phi}_2 \quad \cdots \quad \boldsymbol{\Phi}_L\ ] \tag{6.15}$$

where "$\bullet$" in (6.14) denotes component-by-component multiplication of vectors. $\boldsymbol{\Phi}_i$ , $i = 1, 2, \cdots L$ are $N \times (P_i + 1)$ matrices which contain the phase parameters

only and are defined for each component. The matrix $\mathbf{\Phi}$ is an $N \times \sum_{i=1}^{L}(P_i+1)$ matrix and again contains the phase parameters for all components. Based on the above definitions the expression in (6.3) can be written in matrix notation as

$$\boldsymbol{x} = \mathbf{\Phi}\boldsymbol{a} + \boldsymbol{w} \tag{6.16}$$

In this equation the amplitude parameter vector $\boldsymbol{a}$ enters the equation in a linear way, while the phase parameter vector $\boldsymbol{b}$ enters the equation in a nonlinear way through $\mathbf{\Phi}$. Now the problem is to estimate combined parameter vector $\boldsymbol{\theta} = [\boldsymbol{b}^T \quad \boldsymbol{a}^T]^T$ given the observed data vector $\boldsymbol{x} = [\ x[0] \quad x[1] \quad x[2] \quad \cdots \quad x[N-1]\ ]^T$. It is assumed that the observed data length $N$ is sufficiently greater than the total number of estimated parameters given by $M = \sum_{i=1}^{L}\{(P_i+1)+(Q_i+1)\}$. In this way the matrix notaion in (6.16) will be an over determined system of equations.

Since components are assumed to be well separated on TFD, the number of components, can be estimated from TFD. We assume that $L$ is known in this section. Similarly $P_i$ and $Q_i$ are assumed to be known. A method to estimate them can be found in [96,97]. It was also shown that [101] the polynomial orders can be estimated by segmentation.

With the additive white Gaussian noise assumption, the probability density function (pdf) of data vector $\boldsymbol{x}$, given the parameter vector $\boldsymbol{\theta}$ is given by

$$p(\boldsymbol{x}|\boldsymbol{\theta}) = \frac{1}{(\pi\sigma^2)^N} exp\left\{-\frac{1}{\sigma^2}\|\boldsymbol{x} - \mathbf{\Phi}\boldsymbol{a}\|_2^2\right\} \tag{6.17}$$

where $\sigma^2$ is the noise variance. The log likelihood function is defined as,

$$\Lambda = \log p(\boldsymbol{x}|\boldsymbol{\theta}) = -N(ln\pi + 2ln\sigma) - \frac{1}{\sigma^2}\|\boldsymbol{x} - \mathbf{\Phi}\boldsymbol{a}\|_2^2 \tag{6.18}$$

Since $\boldsymbol{x}$ and $\mathbf{\Phi}$ are complex, by defining $\bar{\boldsymbol{x}} = [Re\{\boldsymbol{x}\}^T \quad Im\{\boldsymbol{x}\}^T]^T$ and $\mathbf{\Psi} = [Re\{\mathbf{\Psi}\}^T \quad Im\{\mathbf{\Psi}\}^T]^T$ , the log-likelihood function can be rewritten in real quantities as

$$\Lambda = -N(ln\pi + 2ln\sigma) - \frac{1}{\sigma^2}\|\bar{\boldsymbol{x}} - \mathbf{\Psi}\boldsymbol{a}\|_2^2 \tag{6.19}$$

Maximizing the log likelihood in (6.19) corresponds to minimizing

$$f(\boldsymbol{a}, \boldsymbol{b}) = \|\bar{\boldsymbol{x}} - \mathbf{\Psi}\boldsymbol{a}\|_2^2. \tag{6.20}$$

For a given phase vector $\boldsymbol{b}$, this cost function is quadratic in amplitude vector $\boldsymbol{a}$. Therefore, amplitude vector $\boldsymbol{a}$ can be solved analytically as

$$\hat{\boldsymbol{a}} = (\boldsymbol{\Psi}^T\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^T\bar{\boldsymbol{x}} \tag{6.21}$$

Using this separability feature of the parameter set and substituting (6.21) in (6.19) the original log-likelihood function can be replaced by

$$\Lambda = -N(ln\pi + 2ln\sigma) - \frac{1}{\sigma^2}J(\boldsymbol{b}) \tag{6.22}$$

where

$$J(\boldsymbol{b}) = \bar{\boldsymbol{x}}^T\boldsymbol{P}_{\boldsymbol{\Psi}}^{\perp}\bar{\boldsymbol{x}} \tag{6.23}$$

and

$$\boldsymbol{P}_{\boldsymbol{\Psi}} = \boldsymbol{\Psi}(\boldsymbol{\Psi}^T\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^T \tag{6.24}$$

is the matrix whose columns spans the signal space and its orthogonal complement is given by

$$\boldsymbol{P}_{\boldsymbol{\Psi}}^{\perp} = \boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{\Psi}} \tag{6.25}$$

While the original cost function was a function of $\boldsymbol{a}$ and $\boldsymbol{b}$, this new augmented function is a function of $\boldsymbol{b}$ only. Like the original cost function this new cost function $J(\boldsymbol{b})$ is also nonlinear in $\boldsymbol{b}$. Therefore, minimization requires iterative methods like nonlinear conjugate gradient or quasi-Newton type methods. These iterative methods require also a good initial estimate to avoid possible local minima. In [34] initial estimates are obtained by PPT. After $\boldsymbol{b}$ is solved iteratively, $\boldsymbol{a}$ is obtained by (6.21).

## 6.3 Alternating Phase and Amplitude Method

The separability feature of the original cost function in (6.19) allows us to reduce the number of unknown parameters via analytical method. Since the resultant cost function is just a function of phase parameters, we will call this method Phase-Only (PO) method. Though PO deals with reduced set of parameters, the resultant cost function $J(\boldsymbol{b})$ is highly nonlinear and more complicated in terms of

function and gradient evaluations. This is a disadvantage when the minimization of the reduced cost function is to be obtained via nonlinear iterative methods. A special case of (6.3) is the problem of estimating $L$ components where the components have a constant amplitude and constant frequency as given by,

$$x[n] = \sum_{i=1}^{L} a_{i,0} e^{j(b_{i,1} n + b_{i,0})} + w[n], \quad n = 0, 1, \cdots, N-1, \tag{6.26}$$

where we have $P_i = 0$ and $Q_i = 1, \quad i = 1, 2, \ldots, L$. For this special case an efficient method which is named as iterative quadratic maximum likelihood (IQML) estimation method [102–105] is developed. IQML is another iterative method that, instead of the augmented cost function $\boldsymbol{J}(\boldsymbol{b})$, minimizes an equivalent cost function whose parameters corresponds to a digital filter coefficients. It is used to find an approximate solution to the ML estimatimation problem. But it is more efficient compared to minimization of $\boldsymbol{J}(\boldsymbol{b})$. However, in our case we have a more general problem where the components have time varying amplitudes and frequencies and IQML is not applicable. Therefore, in this work, an alternative method is proposed for the general problem in (6.3). The method carries out two minimization algorithms in an alternating manner. The method divides the original minimization problem given by (6.19) into two sub minimizations. The idea is to find one parameter set assuming that the other set is known. First assuming that the initial phase estimate $\boldsymbol{b}^0$ is known, the cost function

$$f_{\boldsymbol{a}}(\boldsymbol{a}) = f(\boldsymbol{a}, \hat{\boldsymbol{b}}^0) = \|\bar{\boldsymbol{x}} - \hat{\boldsymbol{\Psi}}^0 \boldsymbol{a}\|_2^2 \tag{6.27}$$

is formed and minimized, and a solution $\hat{\boldsymbol{a}}^1$ is obtained, where $\hat{\boldsymbol{\Psi}}^0$ is the matrix obtained by initial phase parameter estimate $\boldsymbol{b}^0$. Then using this amplitude estimate $\hat{\boldsymbol{a}}^1$ a second cost function

$$f_{\boldsymbol{b}}(\boldsymbol{b}) = f(\hat{\boldsymbol{a}}^1, \boldsymbol{b}) = \|\bar{\boldsymbol{x}} - \boldsymbol{\Psi} \hat{\boldsymbol{a}}^1\|_2^2 \tag{6.28}$$

is formed and minimized, and a solution $\hat{\boldsymbol{b}}^1$ is found. These two minimizations constitute one cycle of proposed algorithm. By repeating this cycle, taking $\hat{\boldsymbol{b}}^1$ as the new initial phase estimate, the estimates $\hat{\boldsymbol{a}}^2$ and $\hat{\boldsymbol{b}}^2$ are obtained. By repeating the cycles sufficiently many times, the final estimates $\hat{\boldsymbol{a}}^*$ and $\hat{\boldsymbol{b}}^*$ are obtained as shown in

$$\hat{\boldsymbol{b}}^0 \rightarrow \hat{\boldsymbol{a}}^1 \rightarrow \hat{\boldsymbol{b}}^1 \rightarrow \hat{\boldsymbol{a}}^2 \rightarrow \hat{\boldsymbol{b}}^2 \rightarrow \hat{\boldsymbol{a}}^3 \rightarrow \hat{\boldsymbol{b}}^3 \cdots \hat{\boldsymbol{a}}^* \rightarrow \hat{\boldsymbol{b}}^* \tag{6.29}$$

The proposed method, which we will call, from now on, Alternating Phase and Amplitude (APA) method, is a generalization of the so-called coordinate descent method [106, 107], where the minimization of a multi-variable function is done by sequentially minimizing with respect to a single variable or coordinate and keeping the others fixed. By cyclically or by a different pattern repeating the same process a minimum for the function is searched. A generalization of coordinate descent method is the Block Coordinate Descent (BCD) method, where the variables are separated into blocks containing more than one variable and the minimization is done over a block of variables and keeping the others fixed. In our case we have two blocks, and the minimization over one block, namely over the amplitude parameter block, is convex.

Past convergence analysis for block coordinate descent methods requires sub-problems to have unique solutions [107], but this property does not hold here: The sub-problem for amplitude (6.27) is convex but not strictly convex and the sub problem for the phase (6.28) is non-convex. Hence the sub-problem for the amplitude has a global solution but it may not be unique and the sub-problem for the phase does not have a unique solution. It may have many local solutions. Fortunately, for the case of two blocks, Grippo and Sciandrone [108] showed that this uniqueness condition is not needed. They proved that the result does not require uniqueness of the solution in each subproblem, which is that any limit point of the sequence generated based on the optimal solutions of each of the two sub-blocks is a stationary point. Directly from (Grippo and Sciandrone, 2000, Corollary 2), the proposed algorithm APA is convergent. But, the convergence may not be a global one only local convergence is assured.

The main trick with proposed algorithm is that during amplitude and phase minimizations we do not have to find the actual minimum. What we are looking for is a sufficient improvement from the current estimate that we have. Therefore, for the phase iterations rather than iterating down to the convergence point we can iterate a sufficient number of iterations to get some improvement. The same is valid for the minimization of $f_{\boldsymbol{a}}(\boldsymbol{a})$ if we decide to use conjugate gradient. But overall alternating phase and amplitude iterations will allow us to converge to a minimum. The first minimization can be chosen to be the minimization of $f_{\boldsymbol{b}}(\boldsymbol{b})$

instead of $f_{\boldsymbol{a}}(\boldsymbol{a})$. Then the sequence in (6.29) will start by $\hat{\boldsymbol{a}}^0$ The decision about which one to start with should be based on which initial parameter vector, $\hat{\boldsymbol{a}}^0$ or $\hat{\boldsymbol{b}}^0$ , is more close to its actual. This cannot be known in advance, but, based on success of the method by which the initial estimates $\hat{\boldsymbol{a}}^0$ or $\hat{\boldsymbol{b}}^0$ are obtained, a decision can be given.

Like $J(\boldsymbol{b})$, the function $f_{\boldsymbol{b}}(\boldsymbol{b})$ is also nonlinear, and we need iterative methods like nonlinear conjugate gradient or quasi-Newton. These methods converge to local minimum and do not guarantee global minimum unless initial estimates are sufficiently close to global optimum. Therefore, we need to find a method which gives us initial estimates. While in [34] initial estimates are obtained by PPT, in this paper we obtained the initial estimates from time-frequency methods. Any cross term-free TF distribution can be used for initial estimates. In this work Short-Time Fourier Transform (STFT) is used as the TFD for its simplicity.

At first cycle, the phase iterations will be started by $\hat{\boldsymbol{b}}^0 = \hat{\boldsymbol{b}}^{TF}$ where $\hat{\boldsymbol{b}}^{TF}$ is the estimate obtained from time-frequency method. In later cycles, the previous cycle estimates will be used. If minimization of $f_{\boldsymbol{a}}(\boldsymbol{a})$ is done analytically, then we will not need any initial value. If iterative methods are going to be used again, the initial estimate $\hat{\boldsymbol{a}}^0 = \hat{\boldsymbol{a}}^{TF}$ obtained from time-frequency method can be used.

As we stated before it is assumed that the different components are well separated in time-frequency plane and have a small instantaneous bandwidth; that is, the components are not crossing each other. Therefore, by using magnitude TFD, the ridges of each component are detected on TF plane. The algorithm detects the ridges on TF plane by detecting local frequency maximums for each time index. Also by using a threshold the effect of noise is reduced, and the IF is detected at points where component is stronger than noise. Therefore, even though when the weak end of some components is interfering on TF plane with some other stronger component, the IF of stronger component is detected at that point, and the week part of other components is not detected. But the estimates obtained with this method, though they are not the best ones, will be sufficient as initial parameters.

Table 6.1: The alternating phase and amplitude (APA) algorithm.

| | |
|---|---|
| 1 | Compute $\|STFT\|$ for $x[n]$, and detect the ridges and the number of components |
| 2 | Compute the instantaneous frequency (IF) $\hat{v}_i[n]$ and $\hat{v}_i(t)$ via polynomial fit for each component |
| 3 | Compute $\hat{\phi}_i(t) = 2\pi \int_0^\pi \hat{v}_i(\tau)d\tau + \hat{\phi}_i(0)$ and $\hat{\phi}_i[n]$, determine $\hat{\boldsymbol{b}}_i^{TF}$ where $\hat{\phi}_i(0)$ is the phase offset estimated from the data |
| 4 | Compute $x[n]e^{-j\hat{\phi}_i[n]}$ and low-pass filter to get $\hat{a}_i[n]$ |
| 5 | Using polynomial fit get $\hat{\boldsymbol{a}}_i^{TF}$ |
| 6 | Minimize $f_{\boldsymbol{b}}(\boldsymbol{b})$ and $f_{\boldsymbol{a}}(\boldsymbol{a})$ in an alternating manner using $\hat{\boldsymbol{a}}_i^0 = \hat{\boldsymbol{a}}_i^{TF}$ and $\hat{\boldsymbol{b}}_i^0 = \hat{\boldsymbol{b}}_i^{TF}$ |

Then from the ridges the instantaneous frequency (IF) samples ($\hat{v}_i[n]$) for each component are estimated and by polynomial fit corresponding polynomial is obtained. Then by integrating this polynomial the phase function $\hat{\phi}_i[n]$ and polynomial coefficients $\hat{\boldsymbol{b}}_i^{TF}$ for each component are obtained. By deciphering $x[n]$ by $e^{-j\hat{\phi}_i[n]}$ and low-pass filtering the result, the amplitude estimate $\hat{a}_i[n]$ is obtained for each component. Again by polynomial fit $\hat{\boldsymbol{a}}_i^{TF}$ is obtained for each component. The overall steps for the proposed APA algorithm are summarized in Table 6.1.

The initial estimates are obtained from signal TFD by steps 1-5 given in Table 6.1. Some other methods could also be used. But in this work the main focus was on the last step. Therefore, though the steps 1-5 were implemented, the efficiency and performance of this part have not been studied in detail. The only concern was to get initial estimates which are close enough to actual values to avoid local minima if possible. But it should be noted that for the comparison purposes the same initial conditions have been used for the proposed APA algorithm and the phase-only method given in [34].

An important issue that we need to question is the uniqueness of the solution to the optimization problem in (6.19). Since we express a component in terms of amplitude and phase functions and these functions are expressed in terms of basis functions, we need to question the uniqueness of the global optimum at three levels.

Starting form the last level, given a phase function $\phi_i[n]$, uniqueness of the parameter vector $\boldsymbol{b}_i$ for this function can be assured if the base functions $p_k[n]$, $k = 0, 1, ..., Q_i$, are independent of each other. The same is valid for amplitude function $a_i[n]$ and parameter vector $\boldsymbol{a}_i$.

Uniqueness at the amplitude and phase function level (model functions level) will not be assured due to phase ambiguity, because if $a_i[n]$ and $\phi_i[n]$ constitute a component then $a_i[n]$ and $\phi_i[n] + \pi$ will also constitute the same component. Therefore, even though $\boldsymbol{a}_i$ is unique for $a_i[n]$ and $\boldsymbol{b}_i$ is unique for $\phi_i[n]$, the pair $a_i[n]$ and $\phi_i[n]$ will not be unique for $s_i[n]$ and, as a result, $\boldsymbol{\theta}_i = [\boldsymbol{b}_i^T \quad \boldsymbol{a}_i^T]^T$ will not be unique for $s_i[n]$. This shows that the global optimum is not unique in terms of model functions, hence in terms of parameter vector $\boldsymbol{\theta} = [\boldsymbol{b}^T \quad \boldsymbol{a}^T]^T$.

On the other hand uniqueness at signal $s_i[n]$ or component level will be possible if the components are well separated on TFD [23] . In simple terms if no component is coinciding at the same time-frequency point with some other component then the components which constitute the sum in (6.2) can be found uniquely. In some cases, even the two components are overlapping in time frequency plane, their unique separation is possible. It is shown that [109, 110] if the TF support of the overlapping area between two components is smaller than the support of an Hermite-Gaussian (HG) function of order zero then they can be separated uniquely. In our case we have the assumption that the components are non-overlapping.

Therefore, even though uniqueness is not satisfied at model functions level hence at parameter level, it can be satisfied at component or signal level with the restrictions on time-frequency plane. In fact, the solution ambiguity in model

or parameter space will not affect the final performance of the component reconstruction as long as the combination of model functions or model parameters gives the same signal or component. In our case we extract the initial parameters for a component from related TF area which is disjoint. Therefore, assuming that the initial parameters are close enough to global optimum, we use these restrictions, which will make the component level uniqueness possible, at the beginning.

On the contrary to the assumptions made on time frequency support of components, in simulations, one example (Ex2) is selected such that the components are slightly crossing each other. But most of the parts are non overlapping, and these parts allow estimation of an initial IF which will help uniqueness, because, we have assumed in Section 2 that the phase orders $Q_i$s are also known. With this assumption, the set of ambiguous IF estimates hence phase estimates are eliminated for this example, because fitting other ambiguous IFs to the known polynomial order will result in higher fit error. Therefore, for similar examples, the time-frequency restriction can be slightly relaxed.

### 6.3.1 Analysis of Computational Cost

With the phase-only method the resultant cost function $J(\boldsymbol{b})$ is given by (6.23). For the sake of the ease of computation we reorganize this equation and have

$$J(\boldsymbol{b}) = \bar{\boldsymbol{x}}^T \boldsymbol{P}_{\boldsymbol{\Psi}}^{\perp} \bar{\boldsymbol{x}} = \bar{\boldsymbol{x}}^T \bar{\boldsymbol{x}} - (\boldsymbol{\Psi}^T \bar{\boldsymbol{x}})^T (\boldsymbol{\Psi}^T \boldsymbol{\Psi})^{-1} \boldsymbol{\Psi}^T \bar{\boldsymbol{x}} \tag{6.30}$$

where $\boldsymbol{\Psi} = [\boldsymbol{\Psi}_1 \ \boldsymbol{\Psi}_2 \ \boldsymbol{\Psi}_2 \ \cdots \ \boldsymbol{\Psi}_L]$ and $\boldsymbol{\Psi}_i$ is given by

$$\begin{aligned} \boldsymbol{\Psi}_i &= \begin{bmatrix} Re\{\boldsymbol{\Phi}_i\} \\ Im\{\boldsymbol{\Phi}_i\} \end{bmatrix} \\ &= \begin{bmatrix} g_o[\boldsymbol{n}] \bullet \cos(\phi_i[\boldsymbol{n}]) & g_1[\boldsymbol{n}] \bullet \cos(\phi_i[\boldsymbol{n}]) & \cdots & g_{P_i}[\boldsymbol{n}] \bullet \cos(\phi_i[\boldsymbol{n}]) \\ g_o[\boldsymbol{n}] \bullet \sin(\phi_i[\boldsymbol{n}]) & g_1[\boldsymbol{n}] \bullet \sin(\phi_i[\boldsymbol{n}]) & \cdots & g_{P_i}[\boldsymbol{n}] \bullet \sin(\phi_i[\boldsymbol{n}]) \end{bmatrix}, \end{aligned} \tag{6.31}$$

where "$\bullet$" again denotes component-by-component multiplication of vectors. The gradient of $J(\boldsymbol{b})$ is given by [34]

$$\nabla J(\boldsymbol{b}) = -2\bar{\boldsymbol{x}}^T \boldsymbol{P}_{\boldsymbol{\Psi}}^{\perp} \boldsymbol{B}, \tag{6.32}$$

where

$$\boldsymbol{B} = [\boldsymbol{B}_1 \ \boldsymbol{B}_2 \ \cdots \ \boldsymbol{B}_L], \tag{6.33}$$

is the matrix which is obtained from concatenation of the sub-matrices,

$$\boldsymbol{B}_i = [\tilde{\boldsymbol{b}}_{i,0} \ \tilde{\boldsymbol{b}}_{i,1} \ \cdots \ \tilde{\boldsymbol{b}}_{i,Q_i}], \tag{6.34}$$

and which are obtained from derivative with respect to phase parameters as

$$\tilde{\boldsymbol{b}}_{i,k} = \frac{\partial \boldsymbol{\Psi}_i}{\partial b_{i,k}} \boldsymbol{R}_i^T \bar{\boldsymbol{x}} \quad k = 0, 1, \cdots, Q_i. \tag{6.35}$$

where $\boldsymbol{R}$ is obtained from

$$\boldsymbol{R} = \boldsymbol{\Psi}(\boldsymbol{\Psi}^T \boldsymbol{\Psi})^{-1} = [\boldsymbol{R}_1 \ \boldsymbol{R}_2 \ \cdots \ \boldsymbol{R}_L]. \tag{6.36}$$

The derivative of $\boldsymbol{\Psi}_i$ with respect to $b_{i,k}$ is computed as follows:

$$\frac{\partial \boldsymbol{\Psi}_i}{\partial b_{i,k}} = \tilde{\boldsymbol{\Psi}}_i \bullet \boldsymbol{G}_k \tag{6.37}$$

where $\tilde{\boldsymbol{\Psi}}_i$ is the reordered version of $\boldsymbol{\Psi}_i$ given by

$$\tilde{\boldsymbol{\Psi}}_i = \begin{bmatrix} -Im\{\boldsymbol{\Phi}_i\} \\ Re\{\boldsymbol{\Phi}_i\} \end{bmatrix} \tag{6.38}$$

and $\boldsymbol{G}_k$ has the same dimensions as $\tilde{\boldsymbol{\Psi}}_i$ and at each column contains the same $2N \times 1$ vector $\begin{bmatrix} p_k[\boldsymbol{n}] \\ p_k[\boldsymbol{n}] \end{bmatrix}$. The multiplication between $\tilde{\boldsymbol{\Psi}}_i$ and $\boldsymbol{G}_k$ is component by component. With the proposed method, the minimization of $f_{\boldsymbol{a}}(\boldsymbol{a})$ either by CG or analytically is relatively easy. Similarly, the computation of $f_{\boldsymbol{b}}(\boldsymbol{b}) = \|\bar{\boldsymbol{x}} - \boldsymbol{\Psi}\hat{\boldsymbol{a}}^0\|^2$ is also easy. By defining

$$\boldsymbol{z} = \boldsymbol{\Psi}\hat{\boldsymbol{a}}^0 = \sum_{i=1}^{L} \boldsymbol{z}_i \tag{6.39}$$

and

$$\boldsymbol{z}_i = \boldsymbol{\Psi}_i \hat{\boldsymbol{a}}_i^0 = \begin{bmatrix} Re\{\boldsymbol{\Phi}_i\} \\ Im\{\boldsymbol{\Phi}_i\} \end{bmatrix} \hat{\boldsymbol{a}}_i^0 = \begin{bmatrix} \boldsymbol{z}_{iR} \\ \boldsymbol{z}_{iI} \end{bmatrix} = \begin{bmatrix} \sum_{k=0}^{P_i} \hat{a}_{i,k} g_k[\boldsymbol{n}] \bullet \cos(\phi_i[\boldsymbol{n}]) \\ \sum_{k=0}^{P_i} \hat{a}_{i,k} g_k[\boldsymbol{n}] \bullet \sin(\phi_i[\boldsymbol{n}]) \end{bmatrix} \tag{6.40}$$

we can write

$$f_{\boldsymbol{b}}(\boldsymbol{b}) = \|\bar{\boldsymbol{x}} - \boldsymbol{\Psi}\hat{\boldsymbol{a}}^0\|^2 = \|\bar{\boldsymbol{x}} - \boldsymbol{z}\|^2. \tag{6.41}$$

Using (6.39)-(6.41) the gradient of $f_{\boldsymbol{b}}(\boldsymbol{b})$, $\nabla f_{\boldsymbol{b}}(\boldsymbol{b})$ is obtained as

$$\nabla f_{\boldsymbol{b}}(\boldsymbol{b}) = -2(\bar{\boldsymbol{x}} - \boldsymbol{z})^T \left[ \frac{\partial \boldsymbol{z}}{\partial b_{1,0}} \; \frac{\partial \boldsymbol{z}}{\partial b_{1,1}} \; \cdots \; \frac{\partial \boldsymbol{z}}{\partial b_{1,Q_1}} \; \cdots \; \frac{\partial \boldsymbol{z}}{\partial b_{L,0}} \; \frac{\partial \boldsymbol{z}}{\partial b_{L,1}} \; \cdots \; \frac{\partial \boldsymbol{z}}{\partial b_{L,Q_L}} \right] \tag{6.42}$$

where

$$\frac{\partial \boldsymbol{z}}{\partial b_{i,l}} = \begin{bmatrix} \boldsymbol{z}_{iR} \\ \boldsymbol{z}_{iI} \end{bmatrix} \bullet \begin{bmatrix} p_l[\boldsymbol{n}] \\ p_l[\boldsymbol{n}] \end{bmatrix}. \tag{6.43}$$

Considering the equations (6.30)-(6.38) and the equations (6.39)-(6.43) it is apparent that function and gradient evaluations for $J(\boldsymbol{b})$ are much more complicated compared to $f_{\boldsymbol{b}}(\boldsymbol{b})$ and $f_{\boldsymbol{a}}(\boldsymbol{a})$. But in order to get a tangible comparison a computational cost analysis has been done and the results are summarized in Tables 24, where, the analysis is based on the assumption that both for the minimization of $J(\boldsymbol{b})$ and $f_{\boldsymbol{b}}(\boldsymbol{b})$ the quasi-Newton algorithm BFGS [106] is used.

The second columns in Tables 6.2 - 6.4 give the required computation for each step during one BFGS or CG iteration. The last columns give the number of multiplications per step. where $\bar{P}_i = P_i + 1$ and $\bar{Q}_i = Q_i + 1$ represent number of parameters for amplitude and phase functions of the $i^{th}$ component. Parameters $N_a = \sum_{i=1}^{L} P_i$ and $N_b = \sum_{i=1}^{L} Q_i$ represent total number of amplitude and phase parameters for all components, respectively. $F_k$ and $G_k$ represent $\boldsymbol{b}_{(k)}$ denotes phase parameter vector for all the components at $k^{th}$ iteration of BFGS. In order to differentiate it from the $\boldsymbol{b}_i$, which is the phase parameter vector for the $i^{th}$ component, the index is taken into parenthesis. Similarly $\boldsymbol{a}_{(i)}$ denotes amplitude parameter vector for all the components at $i^{th}$ iteration of conjugate gradient.

During computation cost analysis some assumptions were made. For example, the matrix inversion cost of an $N_a \times N_a$ matrix was taken as $N_a^3$ multiplications. These types of assumptions do not alter main results but allow us to get a final value.

Considering the phase iterations for $J(\boldsymbol{b})$ in Table 6.2 and phase iterations for $f_{\boldsymbol{b}}(\boldsymbol{b})$ in Table 6.3, we can see that the main step which contributes to the computations is the line search step. This step requires the function and gradient evaluations. Also, comparing the computation cost at this step in parenthesis we see

Table 6.2: Phase iterations for $J(\boldsymbol{b})$ using quasi-Newton (BFGS) algorithm

| **Step** | **Computation** | **Multiplication Cost** |
|---|---|---|
| Initial | $\boldsymbol{H}_0 = \boldsymbol{I}_{\boldsymbol{N}_b}$ | - |
| 1 | $\boldsymbol{d}_k = -\boldsymbol{H}_k \nabla J(\boldsymbol{b}_{(k)})$ | $N_b^2$ |
| 2 | $\alpha_k = -\min_\alpha J(\boldsymbol{b}_k + \alpha \boldsymbol{d}_k)$ | $F_k\{2N(0.5N_a^2 + 2.5N_a + N_b + 10L)$ |
| | Line search | $+N_a^3 + N_a^2 + N_a\}$ |
| | with Wolfe Conditions | $+G_k\{2N(1.5N_a^2 + 3.5N_a + 2N_b$ |
| | | $+2\sum_{i=1}^{L} P_i Q_i + 10L + 1) + N_a^3\}$ |
| 3 | $\boldsymbol{b}_{(k+1)} = \boldsymbol{b}_{(k)} + \alpha \boldsymbol{d}_k$ | $N_b$ |
| 4 | $\boldsymbol{s}_k = \boldsymbol{b}_{(k+1)} - \boldsymbol{b}_{(k)}$ | $N_b + 1$ |
| | $\boldsymbol{y}_k = \nabla J(\boldsymbol{b}_{(k+1)}) - \nabla J(\boldsymbol{b}_{(k)})$ | |
| | $\rho_k = 1/(\boldsymbol{y}_k^T \boldsymbol{s}_k)$ | |
| 5 | $\boldsymbol{C}_1 = (\boldsymbol{I} - \rho_k \boldsymbol{s}_k \boldsymbol{y}_k^T)$ | |
| | $\boldsymbol{C}_2 = (\boldsymbol{I} - \rho_k \boldsymbol{y}_k \boldsymbol{s}_k^T)$ | |
| | $\boldsymbol{H}_{k+1} = \boldsymbol{C}_1 \boldsymbol{H}_k \boldsymbol{C}_2 + \rho_k \boldsymbol{s}_k \boldsymbol{s}_k^T$ | $5N_b^2 + 3N_b$ |

Table 6.3: Phase iterations for $f_{\boldsymbol{b}}(\boldsymbol{b})$ using quasi-Newton (BFGS) algorithm

| **Step** | **Computation** | **Multiplication Cost** |
|---|---|---|
| Initial | $\boldsymbol{H}_0 = \boldsymbol{I}_{\boldsymbol{N}_b}$ | - |
| 1 | $\boldsymbol{d}_k = -\boldsymbol{H}_k \nabla J(\boldsymbol{b}_{(k)})$ | $N_b^2$ |
| 2 | $\alpha_k = -\min_\alpha J(\boldsymbol{b}_k + \alpha \boldsymbol{d}_k)$ | $2NF_k\{N_a + N_b + 11L + 1\}$ |
| | Line search (Wolfe Conditions) | $+2NG_k\{N_a + 3N_b + 11L + 1\}$ |
| 3 | $\boldsymbol{b}_{(k+1)} = \boldsymbol{b}_{(k)} + \alpha \boldsymbol{d}_k$ | $N_b$ |
| 4 | $\boldsymbol{s}_k = \boldsymbol{b}_{(k+1)} - \boldsymbol{b}_{(k)}$ | $N_b + 1$ |
| | $\boldsymbol{y}_k = \nabla J(\boldsymbol{b}_{(k+1)}) - \nabla J(\boldsymbol{b}_{(k)})$ | |
| | $\rho_k = 1/(\boldsymbol{y}_k^T \boldsymbol{s}_k)$ | |
| 5 | $\boldsymbol{C}_1 = (\boldsymbol{I} - \rho_k \boldsymbol{s}_k \boldsymbol{y}_k^T)$ | |
| | $\boldsymbol{C}_2 = (\boldsymbol{I} - \rho_k \boldsymbol{y}_k \boldsymbol{s}_k^T)$ | |
| | $\boldsymbol{H}_{k+1} = \boldsymbol{C}_1 \boldsymbol{H}_k \boldsymbol{C}_2 + \rho_k \boldsymbol{s}_k \boldsymbol{s}_k^T$ | $5N_b^2 + 3N_b$ |

Table 6.4: Amplitude iterations via Minimization of $f_{\boldsymbol{a}}(\boldsymbol{a})$ using conjugate gradient (CG) algorithm

| **Step** | **Computation** | **Multiplication Cost** |
|---|---|---|
| Initial | $\boldsymbol{A} = \boldsymbol{\Psi}^T\boldsymbol{\Psi}, \boldsymbol{y} = \boldsymbol{\Psi}^T\bar{\boldsymbol{x}}$ | - |
| | $\boldsymbol{r}_0 = \boldsymbol{y} - \boldsymbol{A}\boldsymbol{a}_{(0)} = \boldsymbol{\Psi}^T(\bar{\boldsymbol{x}} - \boldsymbol{\Psi}\boldsymbol{a}_{(0)})$ | $2N(3N_a + N_b + 10L)$ |
| 1 | $\alpha_i = (\boldsymbol{r}_i^T\boldsymbol{r}_i)/(\boldsymbol{d}_i^T\boldsymbol{A}\boldsymbol{d}_i)$ | $N_a^2 + 2N_a + 1$ |
| 2 | $\boldsymbol{a}_{(i+1)} = \boldsymbol{a}_{(i)} + \alpha_i\boldsymbol{d}_i$ | $N_a$ |
| 3 | $\boldsymbol{r}_{i+1} = \boldsymbol{r}_i - \alpha_i\boldsymbol{A}\boldsymbol{d}_i$ | $N_a$ |
| 4 | $\beta_{i+1} = (\boldsymbol{r}_(i+1)^T\boldsymbol{r}_(i+1))/(\boldsymbol{r}_i^T\boldsymbol{r}_i)$ | $N_a + 1$ |
| 5 | $\boldsymbol{d}_{i+1} = \boldsymbol{r}_{i+1} + \beta_{i+1}\boldsymbol{d}_i$ | $N_a$ |

that while for $J(\boldsymbol{b})$ the computation cost is $O(NN_a^2)+O(NN_b)+O(N\sum_{i=1}^{L}\bar{P}_i\bar{Q}_i)$, it is $O(NN_a) + O(NN_b)$ for$f_{\boldsymbol{b}}(\boldsymbol{b})$.

If minimization of $f_{\boldsymbol{a}}(\boldsymbol{a})$ is done via conjugate gradient (CG) algorithm then the computation cost is given in Table 6.4. If the minimum is to be found analytically, then the cost of (6.21) need to be taken into account. Using similar analysis, it will be found that the cost of finding minimum of $f_{\boldsymbol{a}}(\boldsymbol{a})$ is approximately $2N(2N_a + N_b + 10L) + 2N_a^3 + N_a^2$.

For a better comparison of APA and PO methods, we need to consider the overall complexity of the two methods. For the minimization of $J(\boldsymbol{b})$, we need to compute the cost of each BFGS iteration, which consists of five steps, and multiply with the number of iterations. On the other hand, for the proposed APA method we need to compute the cost of minimizing $f_{\boldsymbol{b}}(\boldsymbol{b})$ and plus the cost of minimizing $f_{\boldsymbol{a}}(\boldsymbol{a})$ and multiply the result with the number of cycles of alternating phase and amplitude minimizations.

The cost of line search step in minimization of $J(\boldsymbol{b})$ and $f_{\boldsymbol{b}}(\boldsymbol{b})$ with BFGS requires the number of function and gradient evaluations $F_k$ and $G_k$ to be known. But, the actual numbers of the evaluations are not known beforehand; hence, we

need to find them via simulations.

### 6.3.2 Expectation Maximization with Alternating Phase and Amplitude Method

In ML estimation the aim is to maximize the conditional pdf $p(\boldsymbol{x}|\boldsymbol{\theta})$ or its logarithm, that is, $L(\boldsymbol{\theta}) = \log p(\boldsymbol{x}|\boldsymbol{\theta})$, where, $\boldsymbol{x}$ is the observation data vector, $\boldsymbol{\theta}$ is the parameter vector to be estimated, and $L(\boldsymbol{\theta})$ is the logarithmic likelihood function. In most of cases, if the pdf is not Gaussian, analytic maximization is difficult. Therefore, the Expectation Maximization (EM) [100, 111] procedure is used to simplify the maximization iteratively.

The key idea underlying EM is to introduce a latent or hidden variable $\boldsymbol{z}$ whose pdf depends on $\boldsymbol{\theta}$ with the property $p(\boldsymbol{z}|\boldsymbol{\theta})$ whose maximization is easy or, at least, easier than maximizing $p(\boldsymbol{x}|\boldsymbol{\theta})$. The observed data $\boldsymbol{x}$ without hidden or missing data is called incomplete data.

EM is an efficient iterative procedure to compute the Maximum Likelihood (ML) estimate in the presence of missing or hidden data. In other words, the incomplete data $\boldsymbol{x}$ is enhanced by guessing some useful additional information. The hidden vector $\boldsymbol{z}$ is called as complete data in the sense that, if it were fully observed, then estimating $\boldsymbol{\theta}$ would be an easy task.

Technically $\boldsymbol{z}$ can be any variable such that $\boldsymbol{\theta} \rightarrow \boldsymbol{z} \rightarrow \boldsymbol{x}$ is a Markov chain, that is, $\boldsymbol{z}$ is such that, $p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\theta})$ is independent of $\boldsymbol{\theta}$. Therefore, we have

$$p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\theta}) = p(\boldsymbol{x}|\boldsymbol{z}) \tag{6.44}$$

While in some problems there are natural hidden variables, in most of the cases they are artificially defined.

In ML parameter estimation problem given in Section 6.2 the EM method is applied as follows. Assume that we would like to estimate the amplitude and phase parameters $a_k$ and $b_k$ for the $k^{th}$ component given the data $x[n]$ expressed by

(6.3). The data is incomplete in the sense that it includes the linear combination of all the other components together with the noise. But if we knew, somehow, the other components given by

$$d_k[n] = \sum_{i \neq k} a_i[n] e^{j\phi_i[n]} \tag{6.45}$$

then we would be able to define the following new data vector:

$$x_k[n] = x[n] - d_k[n], \quad n = 0, 1, \cdots, N-1 \tag{6.46}$$

In that case, the problem would be, given the data sequence

$$x_k[n] = a_k[n] e^{j\phi_k[n]} + w[n], \quad n = 0, 1, \cdots, N-1, \tag{6.47}$$

estimate the parameters $\boldsymbol{a}_k$ and $\boldsymbol{b}_k$. As we are going to estimate the phase and amplitude parameters of the $k^{th}$ component, $x_k[n]$ can be considered as the complete data in the EM context. Similar to multi-component case given in Section 6.2 the matrix notation and related logarithmic likelihood function for this single component case is

$$\boldsymbol{x}_k = \boldsymbol{\Phi}_k \boldsymbol{a}_k + \boldsymbol{w} \tag{6.48}$$

$$\Lambda_k = -N(ln\pi + 2ln\sigma) - \frac{1}{\sigma^2} \|\bar{\boldsymbol{x}}_k - \boldsymbol{\Psi}_k \boldsymbol{a}_k\|_2^2 \tag{6.49}$$

The minimization of (6.49) can be done either by PO method or by the APA method explained in Section 6.3. But, since we do not know the other components, we would not be able to compute the summation $d_k[n]$ given in (6.45). The only thing that we can do is to get an estimate for the other components. This is what the EM method suggests us. Therefore, for all components, the following EM iteration steps are carried out.

The EM iterations given in Table 6.5 will be carried out for sufficiently many times and when there is no significant change in the value of estimates compared to previous iteration, the iterations will be stopped.

Table 6.5: Expectation Maximization (EM) iteration steps for multi-component signal parameter estimation

| **Step** | **Computation** |
|---|---|
| Initial | Get initial estimates $[\hat{\boldsymbol{a}}_k^T \quad \hat{\boldsymbol{b}}_k^T]^T,\ k = 1, 2, \cdots, L$ via any method |
| 1 | Construct $\hat{\boldsymbol{x}}_k = \boldsymbol{x} - \sum_{i \neq k} \hat{\boldsymbol{\Psi}}_i \hat{\boldsymbol{a}}_i,\ k = 1, 2, \cdots, L$ |
| 2 | Minimize $\Lambda_k = -N(ln\pi + 2ln\sigma) - \frac{1}{\sigma^2}\|\hat{\boldsymbol{x}}_k - \boldsymbol{\Psi}_k \boldsymbol{a}_k\|_2^2,\ k = 1, 2, \cdots, L$ |
| 3 | Update the initial estimates with results in Step 2, and go to Step 1 |

The important thing in the EM method is that the initial estimates should be close enough to the actual values so that the estimate for complete data $\hat{\boldsymbol{x}}_k$ given at Step 1 is not too deteriorated compared to its actual.

Actually, the alternating phase and amplitude minimization proposed in Section 6.3 can also be considered as an application of EM method. While for the minimization of $f_{\boldsymbol{b}}(\boldsymbol{b})$ the amplitude parameters $\boldsymbol{a}$ are the missing or hidden variables, for the minimization of $f_{\boldsymbol{a}}(\boldsymbol{a})$ the phase parameters are missing or hidden variables.

During each EM iteration a mono-component system of equation given by (6.48) is constructed. The related objective function is minimized by proposed APA method. Then this is done for all components and overall steps are repeated for a number of EM iterations. Since the computation cost for APA is $O(N(N_a+N_b))$ and does not involve squares of $N_a$ and $N_b$, minimizing one by one is expected to have a comparable computational cost to that of multi-component case. But since we repeat overall steps for a number of EM iterations, the cost will increase at a ratio of number of EM iterations. Also since during each EM step, we need to compute $d_k[n]$ and $x_k[n]$ given by (6.45) and (6.46), this requires going from parameter space to component or signal space and will also increase computations. Therefore, using EM with the proposed APA method will increase the computational cost compared to the APA method. But, it will be still less than the cost of phase only method because the phase-only method

has $O(NN_a^2) + O(NN_b) + O(N\sum_{i=1}^{L} \bar{P}_i\bar{Q}_i)$ order computation, while EM will approximately have $O(R_{EM}NN_a) + O(R_{EM}NN_b)$ order computations, where $R_{EM}$ is the number of the EM iterations.

### 6.3.3 Cramer-Rao Bounds for Mean Square Reconstruction Error

Before comparing the proposed APA method with any other method in terms of computational cost, we first need to compare them in terms of attainable mean square reconstruction error performance. For that purpose we need to have the Cramer-Rao bounds on selected performance criteria.

Given the likelihood function $\Lambda$ in (6.18) the Fisher Information Matrix (FIM) for the parameter set $\boldsymbol{\theta} = [\boldsymbol{b}^T \quad \boldsymbol{a}^T]^T$ is obtained by

$$F_{ij} = -E\left\{\frac{\partial^2 \Lambda}{\partial\theta_i \partial\theta_j}\right\} \tag{6.50}$$

The matrix is obtained [34] as

$$\boldsymbol{F} = \frac{2}{\sigma^2} Re\left\{\left[[\boldsymbol{A\Phi}]^H \quad [\boldsymbol{A\Phi}]\right]\right\} \tag{6.51}$$

where

$$\boldsymbol{A} = [\boldsymbol{A}_1 \ \boldsymbol{A}_2 \ \cdots \ \boldsymbol{A}_L], \tag{6.52}$$

and

$$\boldsymbol{A}_i = j[\ p_0[\boldsymbol{n}] \bullet s_i[\boldsymbol{n}] \quad p_1[\boldsymbol{n}] \bullet s_i[\boldsymbol{n}] \quad \cdots \quad p_{Q_i}[\boldsymbol{n}] \bullet s_i[\boldsymbol{n}]\ ] \tag{6.53}$$

where $s_i[\boldsymbol{n}]$ is the signal vector obtained by taking values at each time instant and "$\bullet$" denotes Hadamard or component-by-component multiplication of the vectors. An important property of the FIM for $\Lambda$ is that it does not depend on $\boldsymbol{a}$ and $\boldsymbol{b}$ directly but, rather, through phase functions $\phi_i[n]$ and signal components, $s_i[n]$. It also depends on basis functions.

Cramer-Rao bound on variances (auto and cross) of the ML estimates of the parameter set $\boldsymbol{\theta} = [\boldsymbol{b}^T \quad \boldsymbol{a}^T]^T$ is simply the inverse of FIM [112], that is,

$$CRB(\boldsymbol{\theta}) = \boldsymbol{F}^{-1} \tag{6.54}$$

In an actual application rather than the parameters $\boldsymbol{a}$ and $\boldsymbol{b}$ , we will be interested in signal components $s_i[n]$. Therefore, we will drive the bounds on the variance of the estimate for the signal components at time instant $n$. The component $s_i[n]$ is a function of the parameter set $\boldsymbol{\theta}_i = [\boldsymbol{b}_i^T \quad \boldsymbol{a}_i^T]^T$. Having $CRB(\boldsymbol{\theta}_i)$, which is a sub-matrix of $CRB(\boldsymbol{\theta})$, the $CRB(s_i[n])$ can be obtained as [113]

$$CRB(s_i[n]) = (\boldsymbol{s}'_{i,n})^H CRB(\boldsymbol{\theta}_i)\boldsymbol{s}'_{i,n}, \tag{6.55}$$

where

$$\boldsymbol{s}'_{i,n} = \frac{\partial s_i[n]}{\boldsymbol{\theta}_i}. \tag{6.56}$$

using (6.4), (6.5) and (6.53) $\boldsymbol{s}'_{i,n}$ will be obtained as

$$\boldsymbol{s}'_{i,n} = [\ \boldsymbol{A}_i[n] \quad \boldsymbol{\Phi}_i[n]\ ]^T. \tag{6.57}$$

$\boldsymbol{s}'_{i,n}$ is simply the transpose of the row of $[\ \boldsymbol{A}\ \boldsymbol{\Phi}\ ]$ corresponding to time instant $n$. Since in our application we have $N$ time instants we need to compute (6.55) for all of them. But, in order to get an overall performance indication, we will sum them up and obtain the following bound as a reference for the component reconstruction error performance:

$$CRB(s_i) = \sum_{n=0}^{N-1} CRB(s_i[n]), \tag{6.58}$$

where $s_i$ denotes the $i^{th}$ component. This is the total variance bound for the estimate of the signal values at all time instants between 0 and $N-1$.

### 6.3.4 Simulation Results

Though in terms of computation cost some comparison between proposed APA method and phase only method is given in Section 6.3.1, in this section some simulation results are given. For the simulation, three non-stationary multi-component signals were selected. The first two examples have two components and the last example has three components. The real part of components and the magnitude STFT plot of the multi-component signals are given in Figures 6.1 and 6.2.

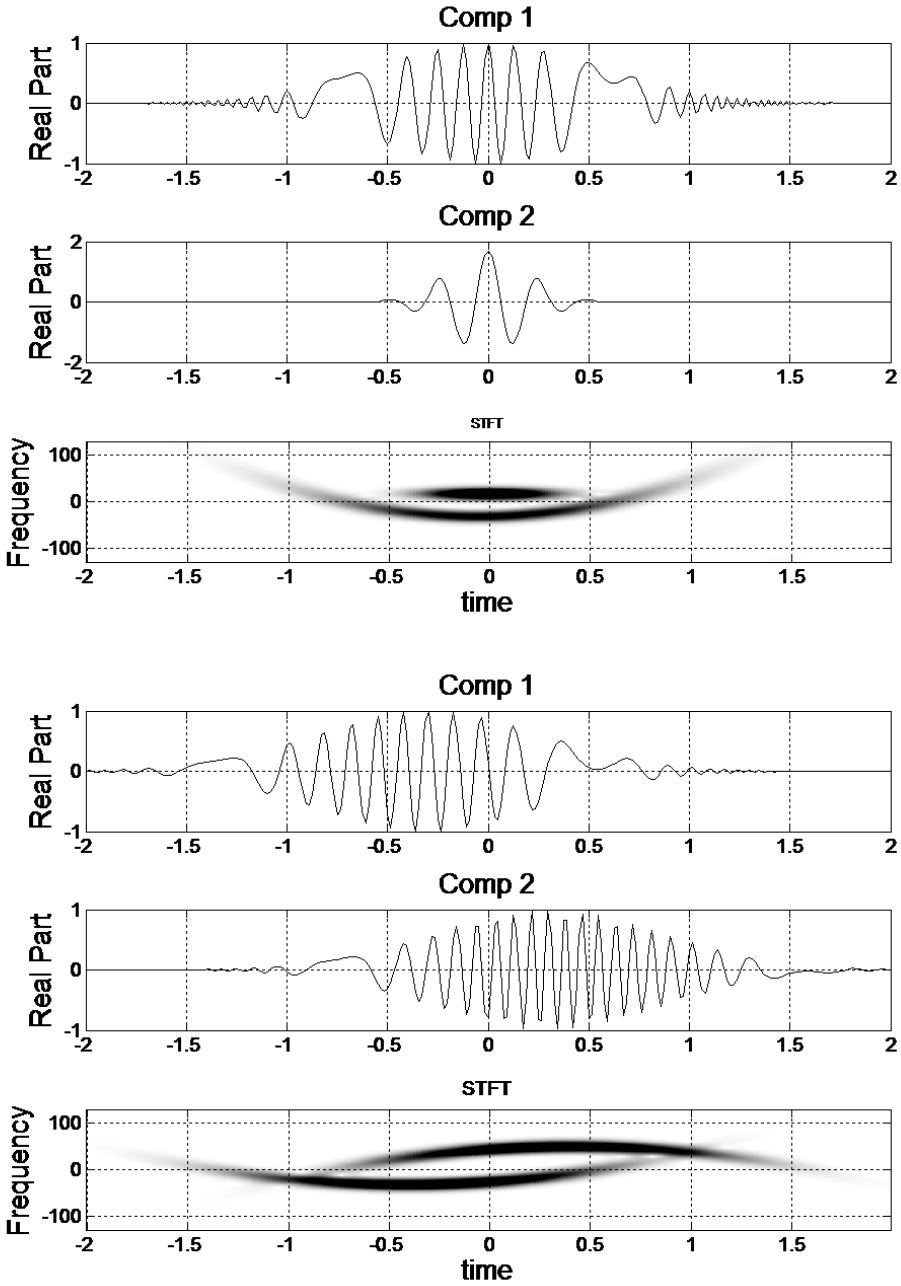


Figure 6.1: The multi-component signals for Ex1 (top) and Ex2 (bottom) with two components.

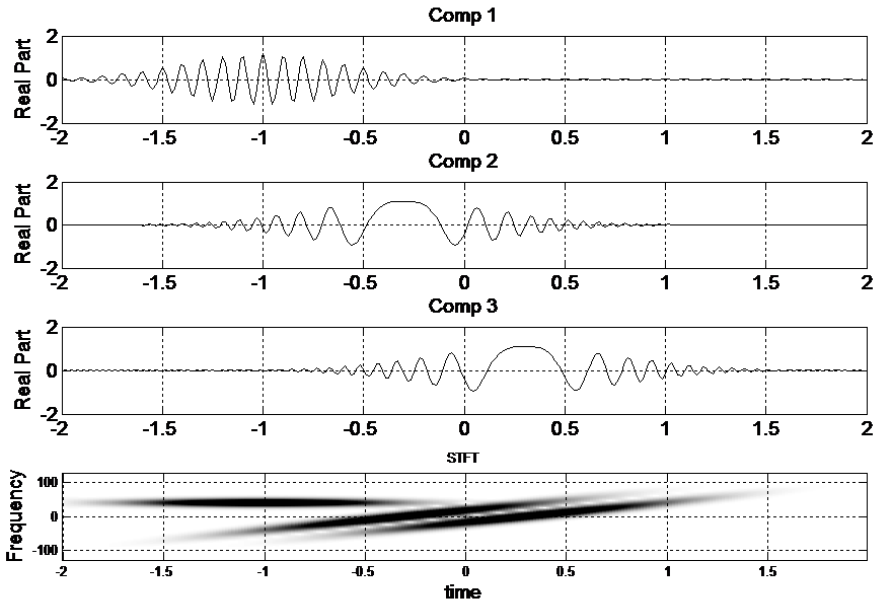


Figure 6.2: The multi-component signal Ex3 with 3 components.

All the examples were selected to be non-stationary signals with 256 samples. The components for the examples were obtained by sampling the following amplitude and phase functions selected with proper parameters and time shifting.

$$a(t,\alpha) = \sqrt[n]{2\alpha}e^{-\pi\alpha t^2} \tag{6.59}$$

$$\phi(t, f_c, \beta, \gamma) = \pi(2f_c t + \beta t^2 + \gamma t^3) \tag{6.60}$$

While Ex1 and Ex2 include components with quadratic phase terms, Ex3 includes two chirps and a Gaussian pulse. Since the phase terms are already polynomials their orders were taken directly for the simulation. But since the amplitude parts are obtained by a Gaussian pulse, their polynomial fit orders were used. The polynomial orders for the examples are given in Table 6.6:

The simulations were carried out as follows: For a given noise realization, the initial estimates $\hat{\boldsymbol{a}}^0 = \hat{\boldsymbol{a}}^{TF}$ and $\hat{\boldsymbol{b}}^0 = \hat{\boldsymbol{b}}^{TF}$ were obtained from TFD. Then using this initial phase parameters, the cost function minimization was carried out by iterating the BFGS algorithm up to a maximum number of steps. The maximum number of steps was set to values 4, 6, 8, 10, 14, 20, and 26, respectively, and the

Table 6.6: Amplitude and phase orders for the components

| | **Polynomial Orders** | | | | | |
|---|---|---|---|---|---|---|
| | **Component1** | | **Component2** | | **Component3** | |
| | Amplitude | Phase | Amplitude | Phase | Amplitude | Phase |
| **Ex1** | 10 | 3 | 20 | 1 | | |
| **Ex2** | 10 | 3 | 10 | 3 | | |
| **Ex3** | 10 | 1 | 10 | 2 | 10 | 2 |

reconstruction error defined by,

$$e_i = \sum_{n=0}^{N-1} |\hat{s}_i[n] - s_i[n]|^2, \tag{6.61}$$

was computed for each component. This error, averaged for many simulation runs, will give us the total of experimental mean square reconstruction error of all time instants for a component and will be compared to the corresponding Cramer Rao Bound given by (6.58).

Then the proposed APA method was iterated with the same initial conditions used for minimization of $J(\boldsymbol{b})$ and with three different scenarios, which define the number of phase iterations and the alternating cycles. Then the minimization with PO and APA were repeated for another noise realization.

In the first scenario of the APA method, denoted by APA1, the number of phase iterations for the minimization of $f_{\boldsymbol{b}}(\boldsymbol{b})$ was taken as the half of that used for minimization of $J(\boldsymbol{b})$. The number of alternating cycles for APA1 was selected as five. For the second scenario, denoted by APA2, the phase iterations for the minimization of $f_{\boldsymbol{b}}(\boldsymbol{b})$ was taken the same as used for $J(\boldsymbol{b})$ and the number of alternating cycles was selected as 8. The third scenario was the EM algorithm with the same conditions as APA1. The EM algorithm given in Table 6.5 was repeated for four iterations.

In all scenarios with the proposed method, the amplitude parameters were computed analytically. Looking at Table 6.4, it is seen that, compared to the minimization of $f_{\boldsymbol{b}}(\boldsymbol{b})$, the cost of minimization of $f_{\boldsymbol{a}}(\boldsymbol{a})$ is lower substantially because the main contribution to computation cost of minimizing $f_{\boldsymbol{a}}(\boldsymbol{a})$ comes from the initialization step and this step is computed once per alternating cycle. Similarly, if minimum $f_{\boldsymbol{a}}(\boldsymbol{a})$ is found analytically, the cost is again small compared to the phase cost.

The Quasi-Newton (BFGS) was implemented with line search algorithm suggested by Nocedal [114], which saves the gradient computations as much as possible. Therefore the minimization of $J(\boldsymbol{b})$ is even favored.

Using above scenarios for each SNR value between 8dB and 20dB, the simulation was carried out for 400 runs. During each run, together with the component reconstruction error, the total number of function and gradient evaluations was also measured for each method and scenario. By averaging 400 runs, the average of the reconstruction error given by (6.61) and the average of the function and gradient evaluations were computed. Based on the average function and gradient evaluations the computation cost for each method and scenario was obtained.

Using simulation results two group of figures were obtained. In Figures 6.3 - 6.6 the attained average reconstruction error (MSE) vs. SNR is plotted for the proposed APA method and for the phase only (PO) method given in [34]. On Figures 6.3 - 6.6 the corresponding Cramer-Rao Bound (CRB) computed using (6.58) is also plotted. PO stands for phase only method. APA1, APA2 and EM stand for the proposed method with scenario 1, 2 and expectation maximization, respectively.

On the other hand, in Figures 6.7 - 6.12 the attained average reconstruction error vs. required computation cost, in terms of millions of multiplications, is plotted for three SNR values. These are 8dB, 14 dB and 20 dB respectively. In Figures 6.7 - 6.12 the Cramer Rao bound (CRB) is also shown as a base line.

In Figures 6.3 - 6.6 the aim is to show that, the proposed method converges to a

comparable, or even better reconstruction error levels than the phase only method [34] in some cases, for a given SNR value and the same initial conditions. But in Figures 6.7 - 6.12 the aim is to show that, whatever the attained reconstruction error level, the proposed method converges with substantially less number of multiplications for a given SNR value and the same initial conditions.

From Figures 6.3 - 6.6 we observe that the proposed method with scenarios APA1, APA2 and EM has a comparable error performance to the phase only method. While for Ex1 the performance of EM is better than the others, for other examples the performance is comparable. Therefore, with the proposed APA method and EM method that uses APA, we are able to solve the optimization problem in (6.19) iteratively and reach a comparable MSE performance compared to the PO method. On the other hand the computational cost performance of the proposed APA and EM method is significantly better than that of PO method. Figures 6.7 - 6.12 show this situation clearly. Figure 6.7 shows that the average reconstruction error for component 1, with the proposed method using the first scenario (APA1), the final reconstruction error level is reached by around three million multiplications A similar level is reached with more than 20 million multiplications by PO method. The multiplication required for the same level for second scenario (APA2) is around 6 millions. On the other hand using EM a better error level is obtained. Similar results can be observed for component 2 as given in 6.8. From Figures 6.11 and 6.12 we see that again for Ex2 and Ex3 at SNR 8dB the proposed method reaches final reconstruction error faster than the PO method.

As can be seen from Figures 6.9 and 6.10 Increasing SNR to 14 or 20 dB for Ex1 makes the benefit of using APA1 or APA2 apparent. The same advantage was observed for Ex2 and Ex3 also. While at low SNR EM is usually better than the others as the SNR increases the advantage of EM is vanishing.

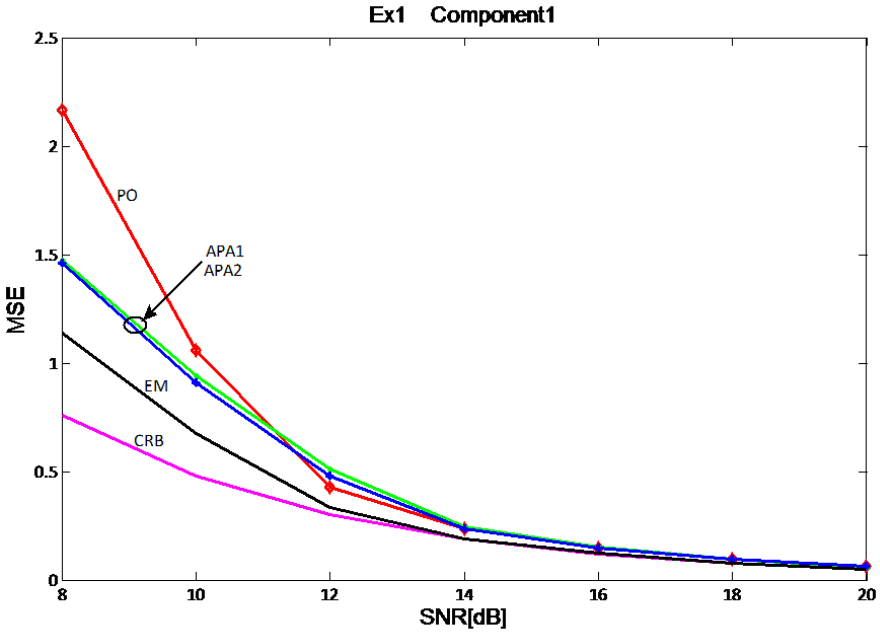


Figure 6.3: Experimental MSE vs. SNR for Ex1 Component1

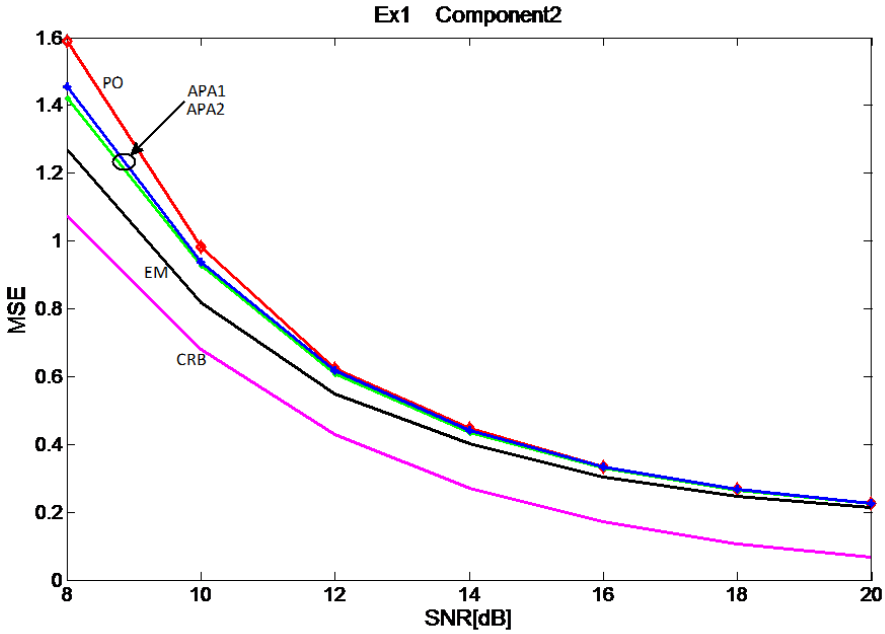


Figure 6.4: Experimental MSE vs. SNR for Ex1 Component2

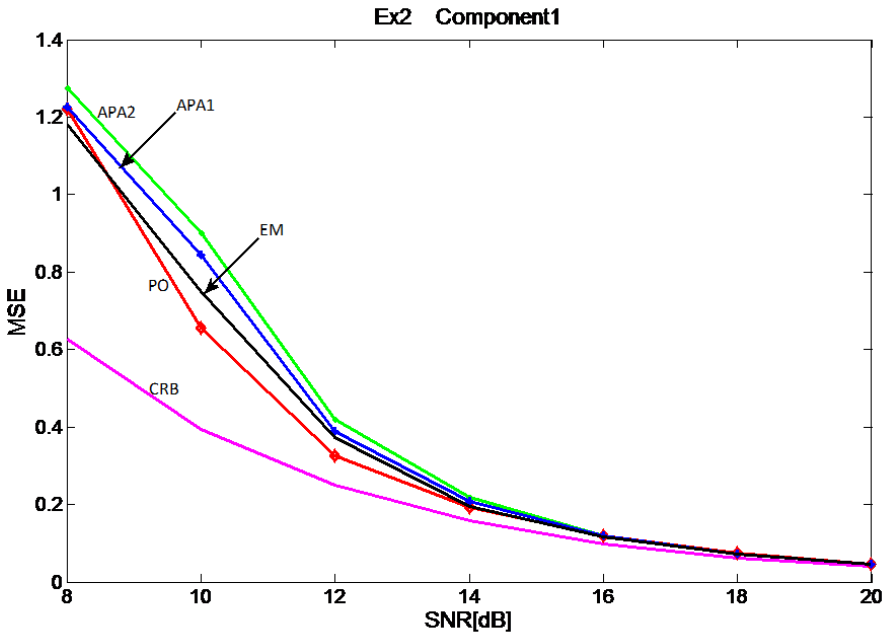


Figure 6.5: Experimental MSE vs. SNR for Ex2 Component1

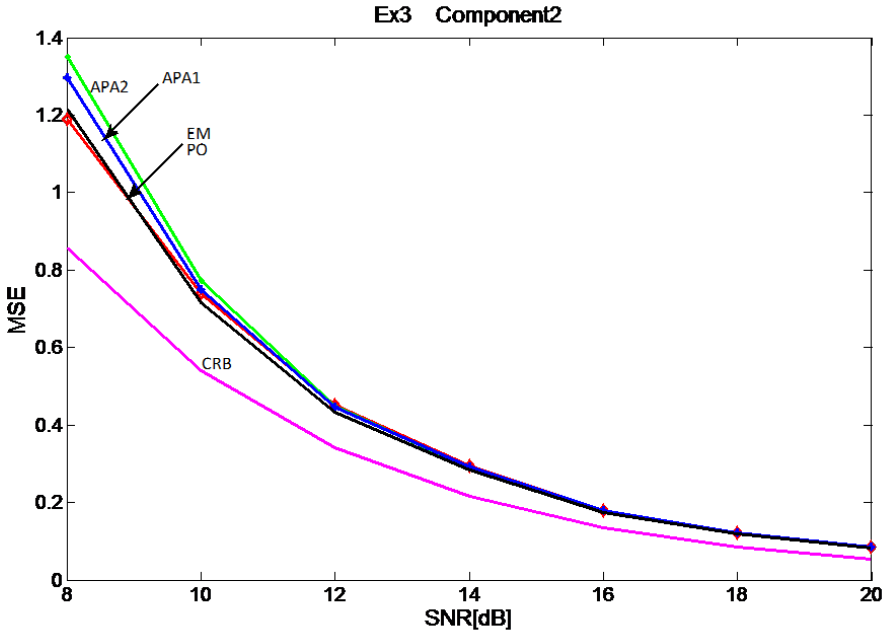


Figure 6.6: Experimental MSE vs. SNR for Ex3 Component2

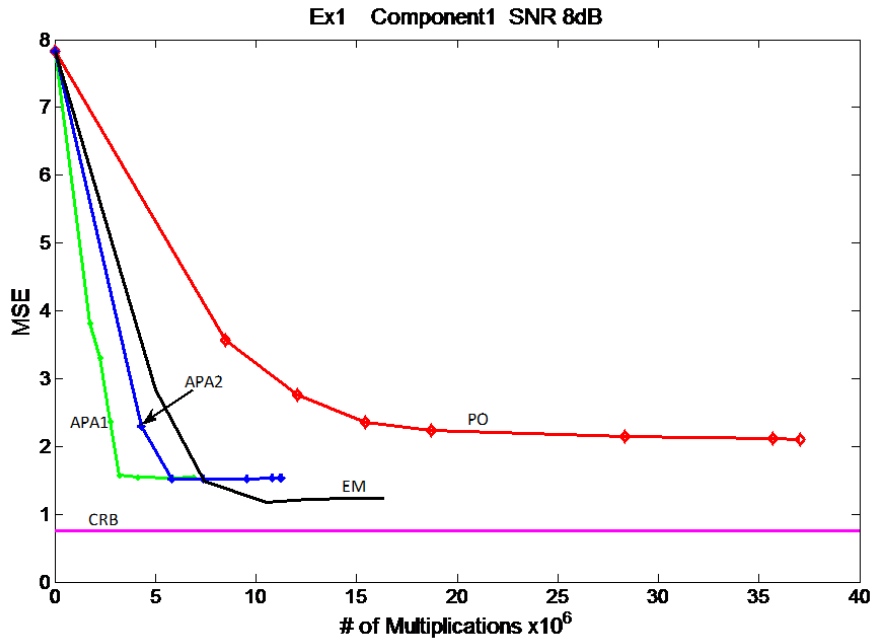


Figure 6.7: Experimental MSE vs. computational cost for Ex1 at 8dB (Component 1)

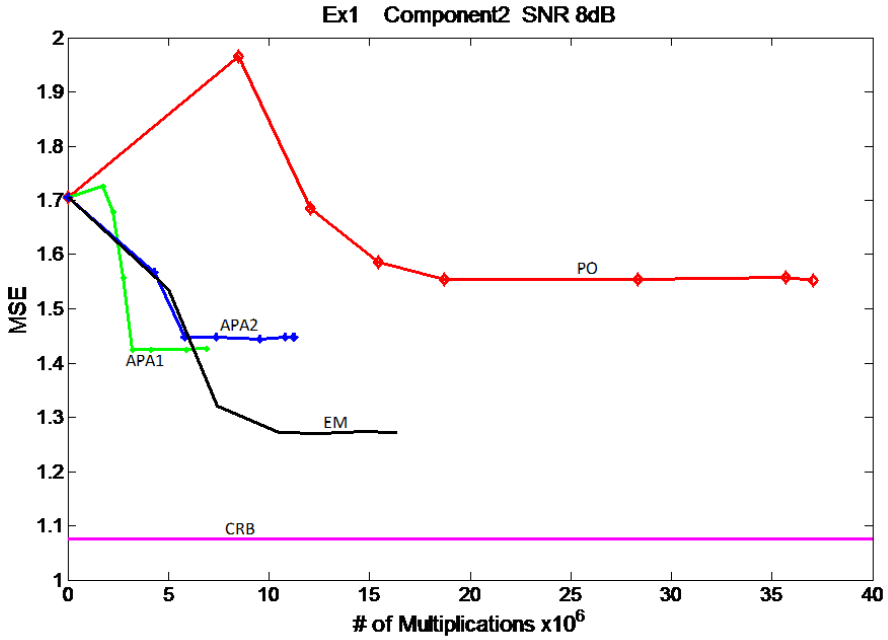


Figure 6.8: Experimental MSE vs. computational cost for Ex1 at 8dB (Component 2)

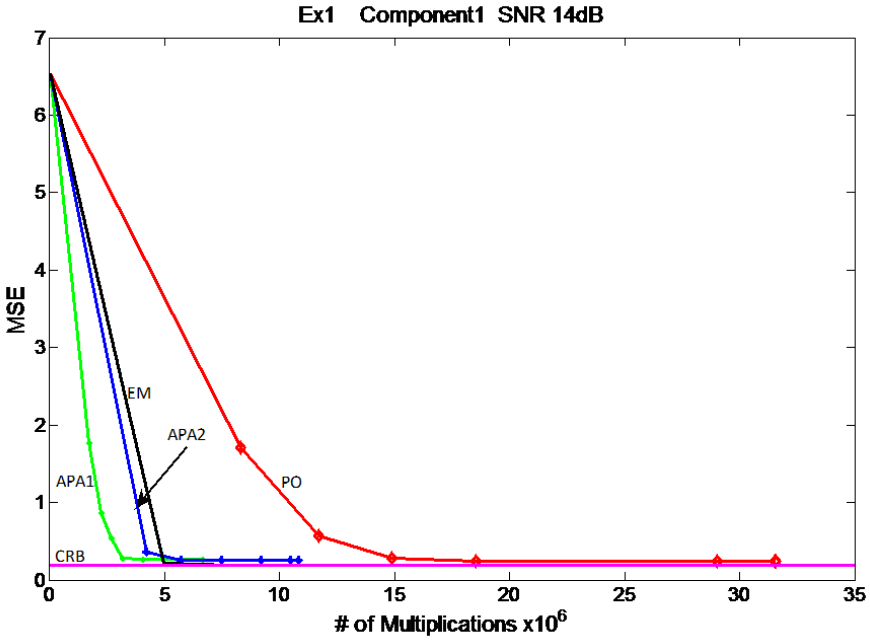


Figure 6.9: Experimental MSE vs. computation cost for Ex1 at 14dB (Component 1

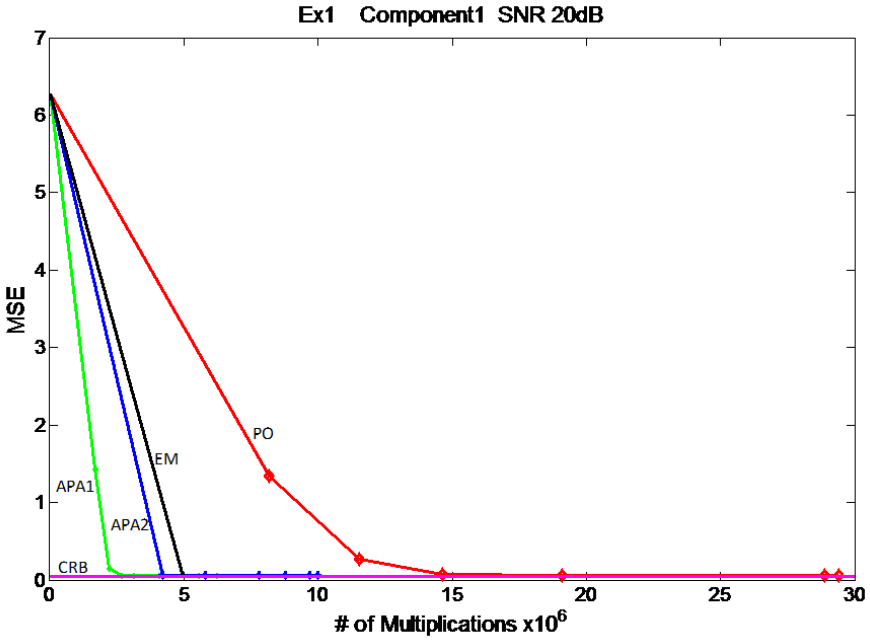


Figure 6.10: Experimental MSE vs. computational cost for Ex1 at 20dB (Component 1

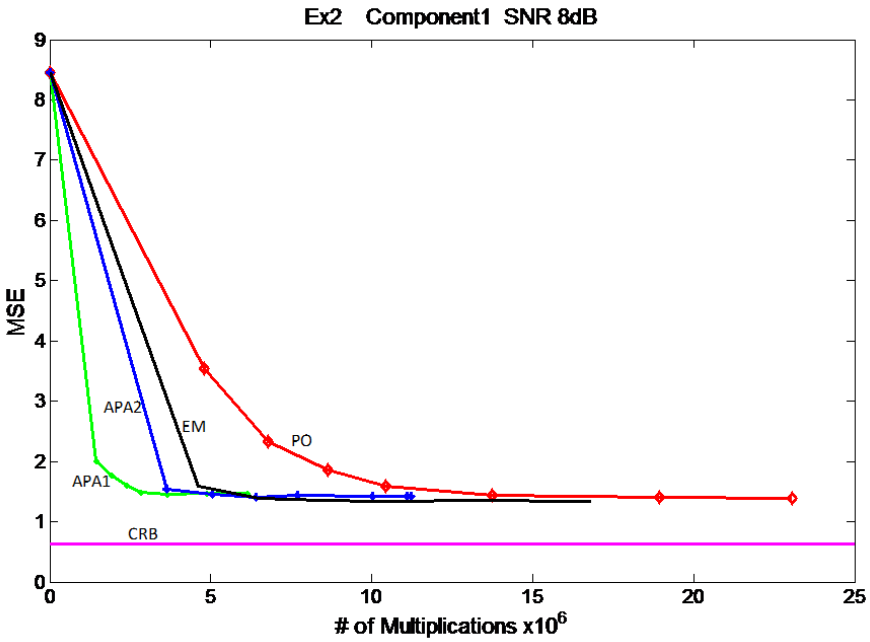


Figure 6.11: Experimental MSE vs. computational cost for Ex2 at 8dB (Component 1

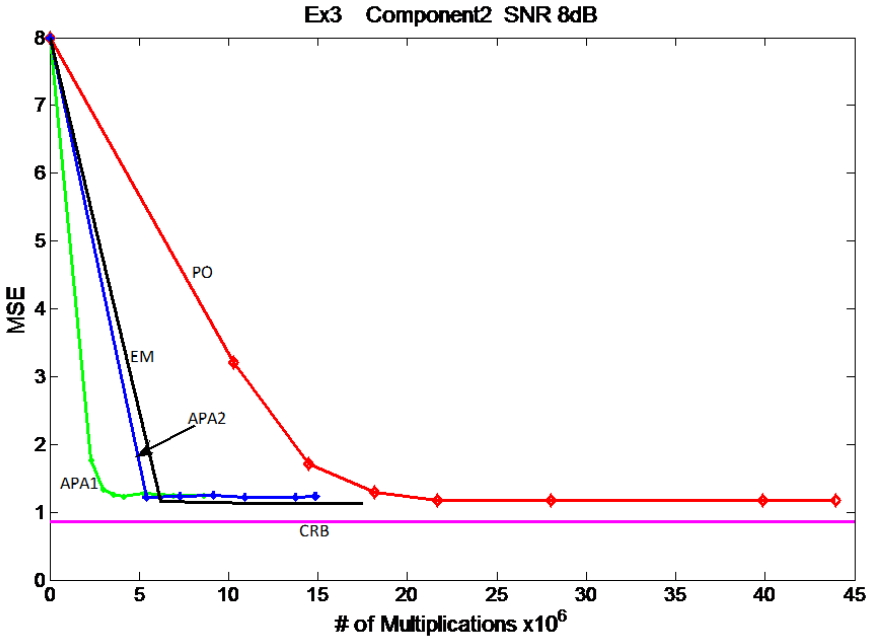


Figure 6.12: Experimental MSE vs. computational cost for Ex3 at 8dB (Component 2

## 6.4 Parameter Estimation with Sparsity Constraint

In the ML parameter estimation method, APA, developed in 6.3, we have minimized two cost functions, given by

$$f_{\boldsymbol{a}}(\boldsymbol{a}) = f(\boldsymbol{a}, \hat{\boldsymbol{b}}^0) = \|\bar{\boldsymbol{x}} - \hat{\boldsymbol{\Psi}}^0 \boldsymbol{a}\|_2^2 \tag{6.62}$$

and

$$f_{\boldsymbol{b}}(\boldsymbol{b}) = f(\hat{\boldsymbol{a}}^1, \boldsymbol{b}) = \|\bar{\boldsymbol{x}} - \boldsymbol{\Psi} \hat{\boldsymbol{a}}^1\|_2^2 \tag{6.63}$$

The amplitude cost function uses $l_2$ as the penalty and is convex. The phase cost function also uses $l_2$ as the penalty, but is non convex. The minimization with these cost functions will not find any possible sparse solution due to the error distributing feature of $l_2$ norm. Sparsity is both important in the case of sparse signal reconstruction and for the better noise immunity. Therefore, in order to look for a possible sparse solution, the following amplitude and phase cost functions can be alternatives to the ML parameter estimation method.

$$f_{\boldsymbol{a}}(\boldsymbol{a}) = \|\bar{\boldsymbol{x}} - \hat{\boldsymbol{\Psi}}^0 \boldsymbol{a}\|_2^2 + \lambda \|\boldsymbol{a}\|_1 \tag{6.64}$$

and

$$f_{\boldsymbol{b}}(\boldsymbol{b}) = \|\bar{\boldsymbol{x}} - \boldsymbol{\Psi} \hat{\boldsymbol{a}}^1\|_2^2 + \lambda \|\boldsymbol{b}\|_1 \tag{6.65}$$

where $\lambda$ is the regularization parameter. Using $l_1$ norm in cost function will force the minimization towards a sparse solution. But the minimization problems, both for amplitude and for the phase functions, have an equation structure where the number of equations is larger than the number of parameters or unknowns. In this respect, the problem for the amplitude is an overdetermined system of linear equations. Similarly the problem for the phase is also an overdetermined non-linear system of equations. In other words, the sparsity is incorporated into the problem at the beginning by assuming that the amplitude and phase functions of the $i^{th}$ component are represented by the orders $P_i$ and $Q_i$ respectively. Even knowing this fact, the cost function 6.64, instead of 6.62, was used in APA method. The aim was to examine whether a better solution will be obtained or not with the $l_1$ regularization. The reason for choosing the amplitude cost function, rather than

phase is that, the number of parameters for the amplitude is much higher than that of phase. The simulation with this cost function has shown no improvement.

The amplitude and phase functions are assumed to be slowly varying signals. As it was stated before, the basis for representing the amplitude and phase functions was selected as polynomials. The orders for representing each function were selected based on their variation. These orders were fixed at the beginning. Since they have a slow variation, they can be represented in Fourier or Discrete Cosine Transform (DCT) domain with a low number of coefficients. Therefore, this side information can also be incorporated into minimization. With this motivation the following cost function,

$$f_{\boldsymbol{a}}(\boldsymbol{a}) = \|\bar{\boldsymbol{x}} - \hat{\boldsymbol{\Psi}}^0 \boldsymbol{a}\|_2^2 + \lambda \|\boldsymbol{M}\boldsymbol{a}\|_1 \tag{6.66}$$

where $\boldsymbol{M} \in R^{N \times N}$ is the measurement matrix given by,

$$\boldsymbol{M} = \boldsymbol{D}\boldsymbol{G} \tag{6.67}$$

where $\boldsymbol{D} \in R^{N \times N}$ is the DCT matrix and $\boldsymbol{G} \in R^{N \times \sum_{i=1}^{L}(P_i+1)}$ is given by

$$\boldsymbol{G} = [\boldsymbol{G}_1 \ \boldsymbol{G}_2 \ \cdots \ \boldsymbol{G}_L] \tag{6.68}$$

and $\boldsymbol{G}_i \in R^{N \times (P_i+1)}$ is the matrix containing polynomial basis functions expressed as

$$\boldsymbol{G}_i = [g_o[\boldsymbol{n}] \quad g_1[\boldsymbol{n}] \ \cdots \ g_{P_i}[\boldsymbol{n}]] \tag{6.69}$$

Once computed, the matrix $\boldsymbol{M}$ is fixed at the beginning of the problem. Through simulation, it was observed that the cost function 6.66, which has sparsity constraint for amplitude function, produces better results for low $SNR$ ($SNR < 8dB$) values. In figures 6.13 the result of simulation are shown. The further research on this topic is considered as a future work.

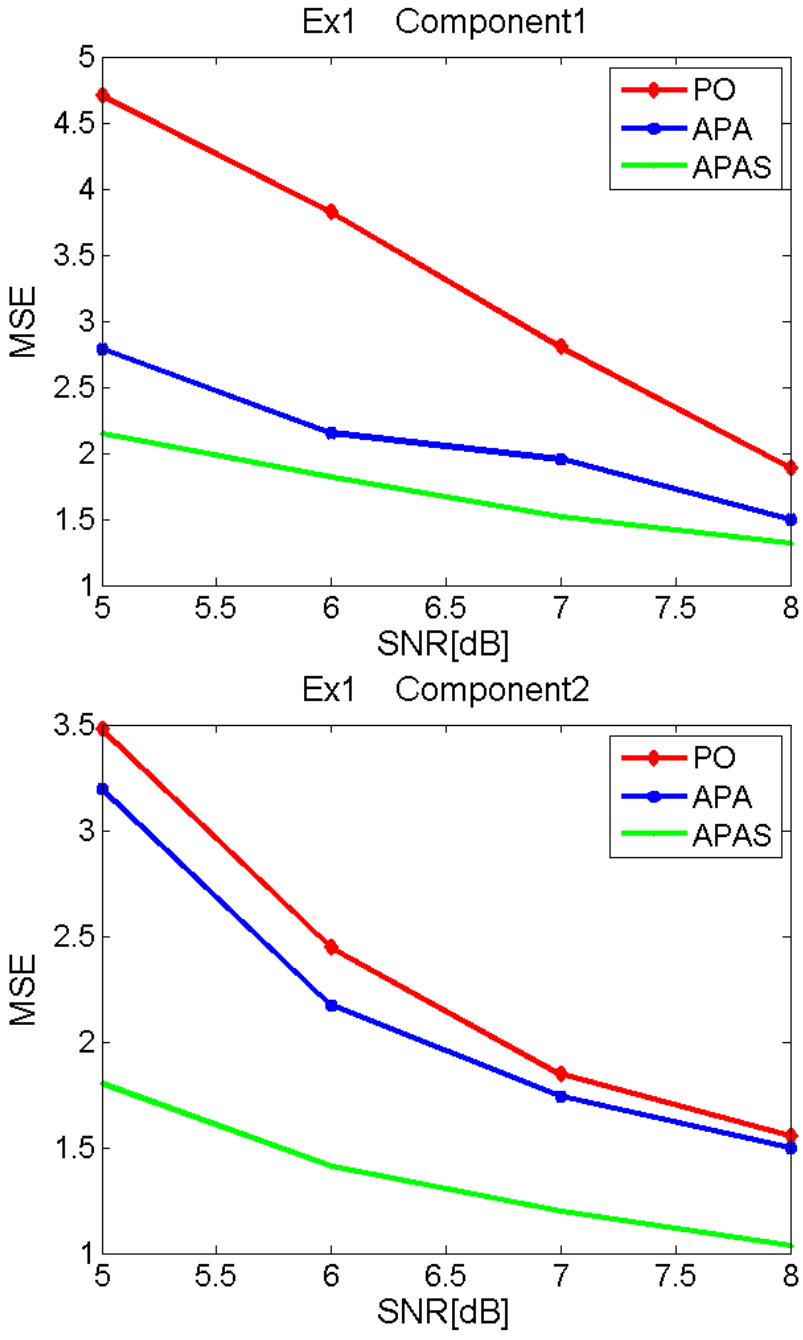


Figure 6.13: Experimental MSE vs. SNR for Ex1 Component 1 (top) and Component 2 (bottom) . PO: phase only method, APA: alternating phase and amplitude method, APAS: APA with sparsity constraint using equitation (6.66) with $\lambda = 0.25$.

# Chapter 7

# Conclusion

Sparsity is an important side information that allows the reconstruction of a signal with a small number of linear and random measurements. For many time-varying and multi-component signals, the TF representation is sparse. This is the point where the CS and TF analysis meet. But there is a difference between the two. In the original CS problem, the aim is to reconstruct the signal from a small number of linear and random measurements at hand. But in the sparsity-based TF reconstruction problem, actually we have all the information, namely the AF domain coefficients. But we intentionally select a set of those coefficients which allow cross-term free reconstruction. Otherwise, the TF will be reconstructed as whole including the cross terms.

The solution to original sparsity-based TF reconstruction with $l_1$ minimization and AF domain constraints was observed to produce over-localized and unacceptable results. Therefore, some relaxation or regularization is needed for an acceptable result. One relaxation is to put a restriction on $l_1$ norm of the solution. The other relaxation can be put on AF domain coefficients. That is, instead of an exact AF coefficient constraint, an approximate AF coefficient constraint can be used. But both of these relaxations mean specifying regularization parameters. In this thesis an epigraph projection-based POCS (Lifted POCS) method was introduced as a solution for this problem. With the developed Lifted POCS

method, cross term-free and high resolution TF representations were obtained without specifying regularization parameters. Through quantitative comparisons it was seen that the method is better than all of the existing smoothing based methods. Compared to reassignment based methods, which have higher localization, it was observed that the Lifted POCS methods are comparable, or in many cases, better in terms of overall localization and similarity based assessment. Considering the spiky nature of the reassignment, Lifted POCS method is preferable in terms of physical meaning related to the signal.

The selection of the type and the size of the AF domain set is an issue which needs further research. Towards this end, two methods were developed. For the same AF domain set, it was observed that selecting only the real part of AF coefficients produces a better TF reconstruction both in terms of localization, and similarity to the desired model TF. It was also observed that, by selecting only the phase of all AF domain coefficients, it is possible to reconstruct a high resolution cross term free TF distribution for most of the signals. However, the AF domain set and size selection problem still needs further investigation. Also, incorporation of additional constraints into Lifted POCS method may produce better results. This will also accelerate the convergence.

The main drawback with the Lifted POCS method was the computation. Although the POCS iterations are using the Fast Fourier Transform (FFT) algorithm, the computation load is still higher than the existing methods. Therefore, an improvement is needed. To this end, a kernel-based TF reconstruction method was developed. The kernel was obtained with a single projection onto the epigraph cost function of $l_1$ norm. The results were obtained faster and, comparable to the optimized kernel. It was observed that the Lifted POCS methods produce better results than both estimated kernel-based method and optimum kernel-based method. Therefore, a faster implementation of CS or Lifted POCS-based methods requires further research.

Sparsity is also used when estimating the components from a time-varying multi-component signal, where there are several components which are disjointed

on TF plane. A parametric representation and related cost function was constructed. An iterative method has been proposed to estimate the components of a multi-component signal via parametric maximum likelihood (ML) estimation. The components on the TF plane are assumed to be well separated. Though, they can be estimated, it was also assumed that the number of components and polynomial orders for amplitude and phase functions are known. The resultant minimization problem was divided into separate amplitude and phase minimizations. With the proposed alternating phase and amplitude minimizations, the computation cost of the original minimization problem reduced significantly. Also, via simulations it was shown that, at low SNR, a better reconstruction error is achieved when the proposed method is used in an EM algorithm.

The initial estimates were obtained from time-frequency distribution. They can also be obtained via Polynomial Phase Transform (PPT). Depending on the performance of the method by which initial estimates are obtained, good initial conditions can be obtained, and the computations can be saved even further.

For the aim of having better estimates, the sparsity constraint was also incorporated into the ML parameter estimation problem. A cost function, which is a mixture of $l_1$ and $l_2$, was used for this purpose. However, no improvement was observed. This is because the sparsity side information is incorporated into the parametric ML estimation problem from the beginning. In other words, it is assumed that the amplitude or phase functions can be represented by the polynomials and the related orders are fixed at the beginning. However, it was observed that by selecting proper basis functions, other than polynomials, and using the sparsity constraint it is possible to have better mean square error at low SNR.

In conclusion,the convex programming methods that use sparsity were developed for time-frequency problems. With these methods cross term-free and high resolution time-frequency distributions were obtained. Also, a parametric ML estimation method was developed for component estimation from a time-varying and multi-component signal.

# Bibliography


[1] F. Auger, P. Flandrin, P. Gonçalves, and O. Lemoine, "Time-Frequency Toolbox For use with MATLAB," 2005.

[2] S. Barbarossa and A. Scaglione, "Adaptive time-varying cancellation of wideband interferences in spread-spectrum communications based on time-frequency distributions," *IEEE Transactions on Signal Processing*, vol. 47, no. 4, pp. 879–898, 2005.

[3] D. R. Wehner, *High-Resolution Radar (2nd ed.).* Boston: Artech House, 1994.

[4] V. C. Chen and S. Qian, "Joint time-frequency transform for radar range-doppler imaging," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 2, pp. 486–499, 1998.

[5] N. Whitelonis and H. Ling, "Radar signature analysis, using a joint time-frequency distribution based on compressed sensing," *IEEE Transactions on Antemnas and Propagation*, vol. 62, no. 2, pp. 755–763, 2014.

[6] V. C. Chen, F. Li, S. HO, and H. Wechsler, "Micro-doppler effect in radar: Phenomenon , model,and simulation study," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 1, pp. 2–21, 2006.

[7] H. Gao, L. Xie, S. Wen, and Y. Kuang, "Micro-doppler signature extraction from ballistic target with micro-motions," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 4, pp. 1969–1982, 2010.

[8] G. C. Gaunaum and H. C. Strifors, "Signal analysis by means of time-frequency (Wigner-type) distributions - applications to sonar and radar echoes," *Proceedings of the IEEE*, vol. 84, no. 9, pp. 1231–1248, 1996.

[9] M. Sun, S. Qian, X. Yan, S. B. Baumann, X.-G. Xia, R. E. Dahl, N. D. Ryan, and R. J. Sclabassi, "Localizing functional activity in the brain through time-frequency analysis and synthesis of the eeg," *Proceedings of the IEEE*, vol. 84, no. 9, pp. 1302–1311, 1996.

[10] A. T. Tzallas, M. G. Tsipouras, and D. I. Fotiadis, "Epileptic seizure detection in EEGs using timefrequency analysis," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 5, pp. 703–710, 2009.

[11] W. G. Anderson and R. Balasubramanian, "Time-frequency detection of gravitational waves," *Phys. Rev. D*, vol. 60, p. 102001, Oct 1999.

[12] J. Sylvestre, "Time-frequency detection algorithm for gravitational wave bursts," *Phys. Rev. D*, vol. 66, p. 102004, Nov 2002.

[13] M. Blodt, M. Chabert, J. Regnier, and J. Faucher, "Mechanical load fault detection in induction motors by stator current time-frequency analysis,," *IEEE Transactions on Industry Applications*, vol. 42, no. 6, pp. 1454–1463, 2006.

[14] G. B. Kliman and J. Stein, "Methods of motor current signature analysis," *Electric Machines and Power Systems*, vol. 20, no. 5, pp. 463–474, 1992.

[15] B. Yazici and G. B. Kliman, "An adaptive statistical time-frequency method for detection of broken bars and bearing faults in motors using stator current," *IEEE Transactions on Industry Applications*, vol. 35, no. 2, pp. 442–452, 1999.

[16] S. Rajagoplan, T. G. Habetler, J. Restrepo, and J. Aller, "Non-stationary motor fault detection using recent quadratic time-frequency representations," *IEEE Transactions on Industry Applications*, vol. 44, no. 3, pp. 735–744, 2008.

[17] G. Yu, S. Mallat, and E. Bacry, "Audio denoising by time-frequency block thresholding," *IEEE Transactions on Signal Processing*, vol. 56, pp. 1830–1839, May 2008.

[18] O. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Transactions on Signal Processing*, vol. 52, pp. 1830–1847, July 2004.

[19] L. Cohen, "Time-frequency distributions - a review," *Proceedings of the IEEE*, vol. 77, no. 7, pp. 941–981, 1989.

[20] L. Cohen, *Time Frequency Analysis*. Englewood Clifs, NJ: Prentice Hall, 1995.

[21] T. A. C. Claasen and W. F. G. Mecklenbraiuker, "The Wigner distribution - A tool for time-frequency signal analysis; part III: relations with other time-frequency signal transformations," *Philips Journal of Research*, vol. 35, no. 6, pp. 372 – 389, 1980.

[22] G. F. Boudreaux-Bartels and T. W. Parks, "Time-varying filtering and signal estimation using Wigner distribution synthesis techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 3, pp. 442–451, 1986.

[23] L. Cohen, "What is a multicomponent signal," *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 113–116, 1992.

[24] W. Krattenthaler and F. Hlawatsch, "Time-frequency design and processing of signals via smoothed Wigner distributions," *IEEE Transactions on Signal Processing*, vol. 41, no. 1, pp. 278–287, 1993.

[25] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transaction on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006. [Online]. Available: http://dx.doi.org/10.1109/TIT.2005.862083.

[26] D. L. Donoho, "Compressed sensing," *IEEE Transaction on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[27] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies," *IEEE Transaction on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4016283.

[28] P. Flandrin and P. Borgnat, "Time-frequency energy distributions meet compressed sensing," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 2974–2982, 2010.

[29] E. Candés and J. Romberg, "l1 -MAGIC: Recovery of sparse signals via convex programming," *User's Guide of the $l_1$ -MAGIC Toolbox for MATLAB*, 2005.

[30] L. Bregman, "Finding the common point of convex sets by the method of successive projection.(Russian)," *USSR Dokl. Akad. Nauk SSSR*, vol. 7, no. 3, pp. 200–217, 1965.

[31] E. J. Candes and J. Romberg, "Practical signal recovery from random projections," *In Proceedings of SPIE Computational Imaging*, vol. 5674, pp. 76–86, 2005.

[32] Z. Deprem and A. E. Çetin, "Crossterm-free Time-Frequency Distribution Reconstruction via Lifted Projections," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 1, pp. 1–13, 2015.

[33] Z. Deprem, K. Leblebicioğlu, O. Arıkan, and A. E. Çetin, "A Complexity-Reduced ML Parametric Signal Reconstruction Method," *EURASIP Journal on Advances in Signal Processing*, pp. 1–14, 2011.

[34] B. Friedlander and J. M. Francos, "Estimation of amplitude and phase parameters of multicomponent signals," *IEEE Transactions on Signal Processing*, vol. 43, no. 4, pp. 917–926, 1995.

[35] F. Hlawatsch and G. Boudreaux-Bartels, "Linear and quadratic time-frequency signal representations," *IEEE Signal Processing Magazine*, vol. 9, no. 2, pp. 21–67, 1992.

[36] L. Durak and O. Arikan, "Short-Time Fourier Transform: Two fundamental properties and an optimal implementation," *IEEE Transactions on Signal Processing*, vol. 51, no. 5, pp. 1231–1242, 2003.

[37] L. Durak, *Novel Time-Frequency Analysis Techniques For Deterministic Signals*. PhD thesis, Bilkent University, Ankara,Turkey, 2003.

[38] H. M. Ozaktaş, Z. Zalevsky, and M. A. Kutay, *The Fractional Fourier Transform with Applications in Optics and Signal Processing*. John Wiley, 2000.

[39] L. B. Almeida, "The fractional fourier transform and time-frequency representations," *IEEE Transactions on Signal Processing*, vol. 42, pp. 3084 – 3091, 1994.

[40] A. K. Özdemir and O. Arıkan, "Fast computation of the ambiguity function and the wigner distribution on arbitrary line segments," *IEEE Transactions on Signal Processing*, vol. 49, no. 2, pp. 381–393, 2001.

[41] S. L., "L-class of time-frequency distributions," *IEEE Signal Processing Letters.*, vol. 3, no. 1, pp. 22–25, 1996.

[42] Y. Wang and Y. Jiang, "New time-frequency distribution based on the polynomial Wigner-Ville distribution and L-Class of Wigner-Ville distribution," *IET Signal Processing*, vol. 4, no. 2, pp. 130–136, 2010.

[43] P. O. S. B.Boashash, "Polynomial Wigner Ville distributions and their relationship to time-varying higher order spectra," *IEEE Transactions on Signal Process.*, vol. 42, no. 1, pp. 216–220, 1994.

[44] S. Ram and H. Ling, "Application of the reassigned joint time-frequency transform to wideband scattering from waveguide cavities," *IEEE Antennas Wireless Propagation Letters*, vol. 6, pp. 580–583, 2007.

[45] P. Flandrin, F. Auger, and E. Chassande-Mottin, "Time-frequency reassignment: From principles to algorithms," in *A. Papandreou-Suppappola (Ed.), Applications in Time-Frequency Signal Processing*, pp. 179 –203, CRC Press, 2003. Ch. 5.

[46] K. Kodera, *Analyse Numerique de Signaux Geophysiques Non-Stationnaires*. PhD thesis, Universite de Paris VI, France, 1976.

[47] K.Kodera, R. Gendrin, and C. D. Villedary, "Analysis of time-varying signals with small bt values," *IEEE Transactions on Acoustics Speech and Signal Processing, ASSP*, vol. 26, no. 1, pp. 6476–1978, 1978.

[48] F. Auger and P. Flandrin, "Improving the readability of timefrequency and timescale representations by the reassignment method," *IEEE Transaction on Signal Process.*, vol. 43, pp. 1068–1089, 1995.

[49] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the Institute of Radio Engineers*, vol. 37, no. 1, pp. 10–21, 1949 [Online] Available: http://ieeexplore.ieee.org/xpls/abs all.jsp?arnumber=1697831.

[50] G. Baraniuk, "Compressed sensing [lecture notes]," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–124, 2007.

[51] E. Candes, "Compressive sampling," *in Proceedings of International Congress of Mathematics*, vol. 3, pp. 1433–1452, 2006.

[52] S. S. Chen, "Basis pursuit," *Ph.D. dissertation, Stanford University*, 1995.

[53] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, 1998.

[54] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorihm for linear inverse problems," *SIAM Journal of Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[55] M. A. T. Figueiredo, D. R. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other 127

inverse problems,," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, December 2007.

[56] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33(1), no. 4, pp. 1–22, 2010.

[57] E. T. Hale, W. Yin, and Y. Zhang, *A fixed-point continuation method for l1 -regularized minimization with applications to compressed sensing.* Rice University, technical report tr07-07 ed., 2007.

[58] H. Guven, H. Ozaktas, A. Cetin, and B. Barshan, "Signal recovery from partial fractional fourier domain information and its applications," *IET Signal Processing*, vol. 2, pp. 15–25, March 2008.

[59] A. E. Çetin and H. M. Özaktaş, "Resolution enhancement of low resolution wavefields with," *Electronics Letters*, vol. 39, pp. 1808–1810, Dec 2003.

[60] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, vol. 20, no. 1-2, pp. 89–97, 2004.

[61] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms,," *Physica D*, vol. 60, no. 4, pp. 259–268, November 1992 [Online]. Available: http://dx.doi.org/10.1016/0167-2789(92)90242-F.

[62] L. Rudin, *Images, Numerical Analysis of Singularities and Shock Filters.* PhD thesis, California Institute of Technology, Pasadena, California, 1987.

[63] S. Osher and L. I. Rudin, "Feature oriented image enhancement using shock filters," *SIAM Journal of Numerical Analysis*, vol. 27, p. 919, 1990.

[64] K. Kose, V. Cevher, and A. Cetin, "Filtered variation method for denoising and sparse signal processing," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3329–3332, 2012.

[65] K. Köse, *Signal and Image Processing Algorithms Using Interval Convex Programming and Sparsity.* PhD thesis, Bilkent University, Ankara,Turkey, 2013.

[66] W. J. Williams, M. L. Brown, and A. . H. III, "Uncertainty, information and time-frequency distributions," *Proceedings of the SPIE*, vol. 1556, p. 144, Dec 1991.

[67] D. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Transactions on Information Theory*, vol. 47, pp. 2845–2862, Nov 2001.

[68] A. E. Cetin, "Reconstruction of signals from fourier transform samples," *Signal Processing*, pp. 129–148, 1989.

[69] A. E. Çetin, A. Bozkurt, O. Gunay, Y. H. Habiboğlu, K. Kose, R. A. Sevimli, and M. Tofighi, " Projections onto Convex Sets (POCS) Based Optimization by Lifting ," in *1st IEEE Global Conference on Signal and Information Processing*, (Austin, Texas, U.S.A.), December 3-5 2013.

[70] G. Chierchia, N. Pustelnik, J.-C. Pesquet, and B. Pesquet-Popescu, "An epigraphical convex optimization approach for multicomponent image restoration using non-local structure tensor," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1359–1363, 2013.

[71] D. Youla and H. Webb, "Image Restoration by the Method of Convex Projections: Part I Num2014;theory," *IEEE Transactions on Medical Imaging*, vol. 1, no. 2, pp. 81–94, 1982.

[72] P. Combettes, "The foundations of set theoretic estimation,," *Proceedings of the IEEE*, vol. 81, no. 2, pp. 182–208, 1993.

[73] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. New York: Springer, 2011.

[74] I. Sezan and H. Stark, "Image restoration by the method of convex projections: Part 2-applications and numerical results," *IEEE Transactions on Medical Imaging*, vol. 1, no. 2, pp. 95–101, 1982.

[75] H. Stark, *Image recovery: theory and application*. Florida: Academic Press, 1987.

[76] L. Gubin, B. Polyak, and E. Raik, "The method of projections for finding the common point of convex sets," *USSR Computational Mathematics and Mathematical Physics*, vol. 7, no. 6, pp. 1–24, 1967.

[77] A. E. Çetin, O. N. Gerek, , and Y. Yardimci, "Equiripple fir filter design by the fft algorithm," *IEEE Signal Processing Magazine*, vol. 14, no. 2, pp. 60–64, 1997.

[78] T.-C. Chen, "Joint signal parameter estimation of frequency hopping communications," *IET Communications*, vol. 6, no. 4, pp. 381–389, 2012.

[79] R. G. Baraniuk, P. Flandrin, A. J. E. M. Janssen, and O. Michel, "Measuring time-frequency information content using the Rényi entropies," *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1391–1409, 2001.

[80] S. Kullback, *Information Theory and Statistics*. John Wiley and Sons, 1959.

[81] K. Pearson, "Notes on regression and inheritance in the case of two parents," in *Proceedings of the Royal Society of London*, vol. 58, pp. 240–242, 1895.

[82] E. Levina and P. Bickel, "The earth mover's distance is the mallows distance: some insights from statistics," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 2, pp. 251–256 vol.2, 2001.

[83] M. Hayes, J. Lim, and A. Oppenheim, "Signal reconstruction from phase or magnitude," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, pp. 672–680, Dec 1980.

[84] F. Hlawatsch and P. Flandrin, "The interference structure of the Wigner Distribution and related time-frequency signal representations," *in The Wigner Distribution Theory and Applications in Signal Processing, W. Mecklenbrauker and F. Hlawatsch, Eds.*, pp. 59–133, 1997.

[85] P. Flandrin, "Some features of time-frequency representations of multicomponent signals," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, vol. 9, pp. 266–269, Mar 1984.

[86] H. I. Choi and W. J. Williams, "Improved timefrequency representation of multicomponent signals using exponential kernels," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP37, p. 862871, June 1989.

[87] A. Papandreou and G. F. Boudreaux-Bartels, "Distributions for time frequency analysis: A generalization of choiwilliams and the butterworth distributions," *IEEE Transactions on Signal Processing*, vol. 5, p. 181184, 1992.

[88] R. G. Baraniuk and D. L. Jones, "Signaldependent timefrequency analysis using a radially gaussian kernel," *Signal Processing.*, vol. 32, no. 3, pp. 263 – 284, june 1993.

[89] D. L. Jones and R. G. Baraniuk, "An adaptive optimal-kernel time-frequency representation," *IEEE Transactions on Signal Processing*, vol. 43, no. 10, pp. 2361 – 2371, 1995.

[90] A. Ozdemir and O. Arikan, "A high resolution time frequency representation with significantly reduced cross-terms," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, vol. 2, pp. II693–II696 vol.2, 2000.

[91] B. Ristic and B. Boashash, "Kernel design for time-frequency signal analysis using the radon transform," May 1993.

[92] H. M. Ozaktas, B. Barshan, D. Mendlovic, and L. Onural, "Convolution, filtering, and multiplexing in fractional fourier domains and their relation to chirp and wavelet transforms," *Journal of the Optical Society of America A*, vol. 11, pp. 547–559, 1994.

[93] K.-B. Yu and S. Cheng, "Signal synthesis from Pseudo-Wigner distribution and applications," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-35, no. 9, 1987.

[94] L. Cohen and T. Posch, "Positive time-frequency functions,"

[95] P. L. Loughlin, J. W. Pitton, and L. E. Atlas, "Construction of positive time-frequency distribution," *IEEE Transactions on Signal Processing,*, vol. 42, no. 10, pp. 2697–2705, 1994.

[96] B. Friedlander, *Parametric signal analysis using the polynomial phase transform*. 1993.

[97] S. Peleg, *Estimation and detection with the discrete polynomial transform*. Davis CA: Dept. Elec. Comput. Eng. Univ. of Califomia, 1993.

[98] A. M. Z. Duc Son Pham, "Analysis of multicomponent phase signals," *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 56–65, 2007.

[99] A. Francos and M. Porat, "Analysis and synthesis of multi component signals using positive time-frequency distributions," *IEEE Transactions on Signal Processing*, vol. 47, no. 2, pp. 493–504, 1999.

[100] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. New York: John Wiley, 1996.

[101] Z. Deprem, A. E. Çetin, and O. Arıkan, "AM/FM Signal Estimation With Micro Segmentation and Polynomial Fit," *Signal, Image and Video Processing Journal*, vol. 8, no. 3, pp. 399–413, 2014.

[102] V. Nagesha and S. Kay, "On frequency estimation with the iqml algorithm," *IEEE Transactions on Signal Processing*, vol. 42, pp. 2509–2513, Sep 1994.

[103] J. Li, P. Stoica, and Z.-S. Liu, "Comparative study of IQML and MODE direction-of-arrival estimators," *IEEE Transactions on Signal Processing*, vol. 46, pp. 149–160, Jan 1998.

[104] R. Kumaresan, L. Scharf, and A. Shaw, "An algorithm for pole-zero modeling and spectral analysis," Jun 1986.

[105] Y. Bresler and A. Macovski, "Exact maximum likelihood parameter estimation of superimposed exponential signals in noise," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, pp. 1081–1089, Oct 1986.

[106] D. B. Luenberger and Y. Ye, *Linear and Nonlinear Optimization*. No. (ISBN 978-0-387-74502-2), Springer, 3 ed., 2008.

[107] D. P. BERTSEKAS, *onlinear Programming*. Belmont, MA 02178-9998: Athena Scientific, second edition ed., 1999.

[108] L. GRIPPO and M. SCIANDRONE, "On the convergence of the block nonlinear gauss-seidel method under convex constraints," *Operations Research Letters*, vol. 26, pp. 127–136, 2000.

[109] Y. K. Alp and O. Arıkan, "Time-frequency analysis of signals using support adaptive Hermite-Gaussian expansions," *Digital Signal Processing*, vol. 22, no. 6, pp. 1010 – 1023, 2012.

[110] Y. K. Alp, *Novel Solutions to Classical Signal Processing Problems in Optimization Framework*. PhD thesis, Bilkent University, Ankara, Turkey, 2014.

[111] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from in-complete data via the em algorithm," *Journal of the Royal Statistical Society: Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[112] S. Zacks, *The Theory of Statistical Inference*. New York: Wiley, 1971.

[113] C. R. Rao, *Linear Statistical Inference and Its Applications*. New York: Wiley, 1965.

[114] N. J. and W. S. J., *Numerical Optimization*. No. (ISBN 0387987932)(651s), New York: Springer, 1999.

# Appendix A

# The Projection onto Epigraph Set of a Convex Function

## A.1 The Projection onto Epigraph Set by Succesive Projections

In lifted POCS method orthogonal projections onto sets $C_f$ and $C_{AF}$ has to be performed. In this Appendix the projection operation onto the epigraph set $C_f$ of a convex function $f$ is described. While the projection onto measurement set $C_{AF}$ is obtained using Fourier Transform relations in (3.17)-(3.19), the projection onto $C_f$ given in (4.2) cannot be obtained in a closed form. The projection onto $C_f$ is implemented using successive projections onto supporting hyperplanes.

Given $\boldsymbol{x} \in \boldsymbol{R}^n$, $f : \boldsymbol{R}^n \to R$ a convex function, the epigraph set of the function $f$ is defined as

$$C_f = \{\boldsymbol{w} = [\boldsymbol{x}^T \quad v]^T \in \boldsymbol{R}^{n+1} \quad | \quad f(\boldsymbol{x}) \leq v\} \tag{A.1}$$

where $\boldsymbol{w} \in \boldsymbol{R}^{n+1}$ is a vector defined in lifted domain, $v \in \boldsymbol{R}$ is the last element

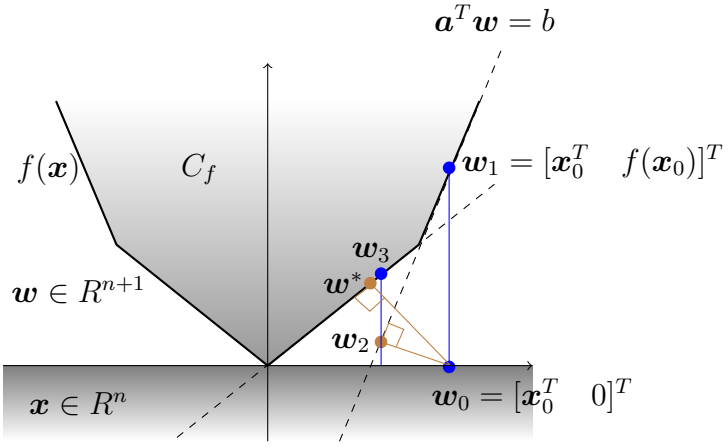


Figure A.1: Projection onto epigraph set $C_f$ by successive projections onto supporting hyperplanes

of $\boldsymbol{w}$.

Given the initial point $\boldsymbol{w}_0 = [\boldsymbol{x}_0^T \quad 0]^T$ , a supporting hyperplane for $C_f$ is defined at $\boldsymbol{x}_0$. The supporting hyperplane is the set of points in $\boldsymbol{R}^{n+1}$ satisfying $\boldsymbol{a}^T\boldsymbol{w} = b$ where $\boldsymbol{a}$ and $b$ are given by

$$\boldsymbol{a} = \begin{bmatrix} \nabla f(\boldsymbol{x}_0) \\ -1 \end{bmatrix} \in \boldsymbol{R}^{n+1} \tag{A.2}$$

$$b = \boldsymbol{a}^T \begin{bmatrix} \boldsymbol{x}_0 \\ f(\boldsymbol{x}_0) \end{bmatrix} \in \boldsymbol{R} \tag{A.3}$$

where $\nabla f(\boldsymbol{x}_0)$ is the gradient of the cost function $f$ at $\boldsymbol{x}_0$. The supporting hyperplane, as shown in Figure A.1, is tangent at $\boldsymbol{w}_1 = [\boldsymbol{x}_0^T \quad f(\boldsymbol{x}_0)]^T$ to $C_f$. The vector $\boldsymbol{w}_0$ is projected onto this hyperplane and $\boldsymbol{w}_2 = [\boldsymbol{x}_2^T \quad v_2]^T$ is obtained.

Then a second supporting hyperplane is defined at $\boldsymbol{x}_2$. This second hyperplane is again tangent at $\boldsymbol{w}_3 = [\boldsymbol{x}_2^T \quad f(\boldsymbol{x}_2)]^T$ to $C_f$. The vector $\boldsymbol{w}_0$ is re-projected onto second hyperplane and $\boldsymbol{w}_4 = [\boldsymbol{x}_4^T \quad v_4]^T$ is obtained. This iteration continues until the projected point $\boldsymbol{w}_k$ satisfies $\boldsymbol{w}^* = [\boldsymbol{x}^{*T} \quad v^*]^T \in C_f$. Since this is an iterative process the iterations are stopped after a fixed number of steps or there is no

improvement between consecutive steps. Once an increase in distance is detected, a refinement should be done for the point at which hyperplane is defined.

The distance $\|\boldsymbol{w}_i - \boldsymbol{w}_0\|_2$ between point to be projected and current projection will not always decrease for high values of iteration $i$ therefore the distance need to be monitored. In this case the hyperplane for $f$ should be defined at the point $(\boldsymbol{x}_i + \boldsymbol{x}_{i-1})/2$.

If the gradient $\nabla f(\boldsymbol{x}_0)$ is not computable then the concept of subgradient can be used to determine a supporting hyperplane at $\boldsymbol{x}_0$. The pseudo-code for projection onto epigraph set of a convex cost function is listed in Algorithm 3 in Appendix A.2.

## A.2 The pseudo-code for projection onto epigraph set of a convex cost function

**Algorithm 3** The pseudo-code for projection onto epigraph set of a convex cost function

**function** $\boldsymbol{x}_p$=EPIGP($\boldsymbol{x0}$)
$\boldsymbol{w}_0 = [\boldsymbol{x}_0^T \quad 0]^T$
$i = 0 \quad ; \epsilon = 10^{-3}$
  **while** $err \geq \epsilon$ **do**
    $\boldsymbol{w}_i = [\boldsymbol{x}_i^T \quad 0]^T \quad ; \quad d_i = \|\boldsymbol{w}_i - \boldsymbol{w}_0\|_2^2$ %distance before projection
    $\boldsymbol{a} = [\nabla f(\boldsymbol{x}_i)^T \quad -1]^T \quad b = \boldsymbol{a}^T[\boldsymbol{x}_i^T \quad f(\boldsymbol{x}_i)]$ %supporting hyperplane
    $\boldsymbol{w}_p = \boldsymbol{w}_i + \frac{b - \boldsymbol{a}^T \boldsymbol{w}_i}{\|\boldsymbol{a}\|^2}\boldsymbol{a} \quad = \quad [\boldsymbol{x}_p^T \quad v]^T$ %orthogonal projection
    $d_p = \|\boldsymbol{w}_p - \boldsymbol{w}_0\|_2^2$ %distance after projection
    $\boldsymbol{w}_R = [\boldsymbol{x}_i^T \quad f(\boldsymbol{x}_i)]^T \quad ; \quad \boldsymbol{w}_L = [\boldsymbol{x}_p^T \quad f(\boldsymbol{x}_p)]^T$
    **if** $d_p > d_i$ **then** $\boldsymbol{w}_{i+1} = (\boldsymbol{w}_L + i\boldsymbol{w}_R)/(i+1)$ % check for distance
    **else** $\boldsymbol{w}_{i+1} = (i\boldsymbol{w}_L + \boldsymbol{w}_R)/(i+1)$
    **end if**
    $\boldsymbol{x}_{i+1} = (\boldsymbol{w}_{i+1})_{1:n}$
    $err = \|\boldsymbol{x}_{i+1} - \boldsymbol{x}_i\|_2 / \|\boldsymbol{x}_i\|$
    $i = i + 1$
  **end while**
**end function**

# Appendix B

# Signal Examples Used in Simulations

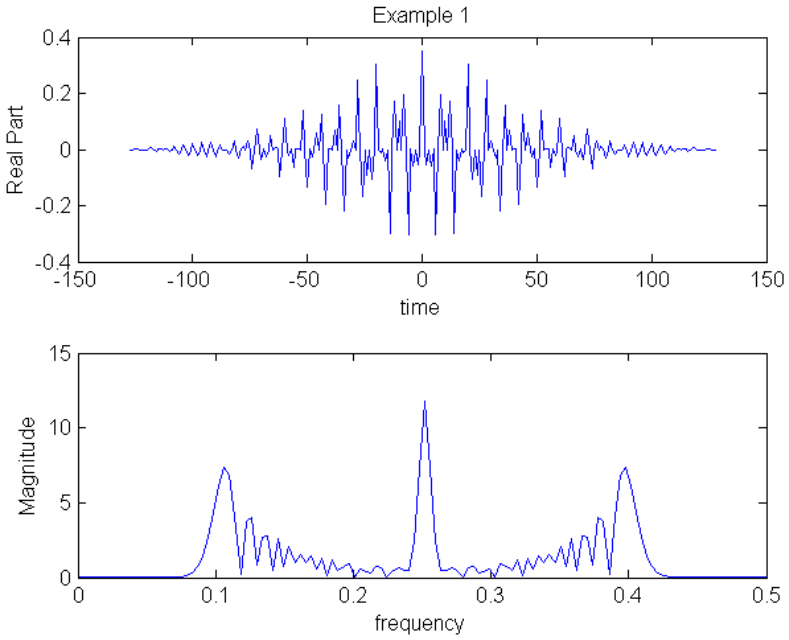


Figure B.1: Time domain representation (top) and the Fourier transform (FFT) of the Example signal 1 where the frequency is normalized to sampling frequency.

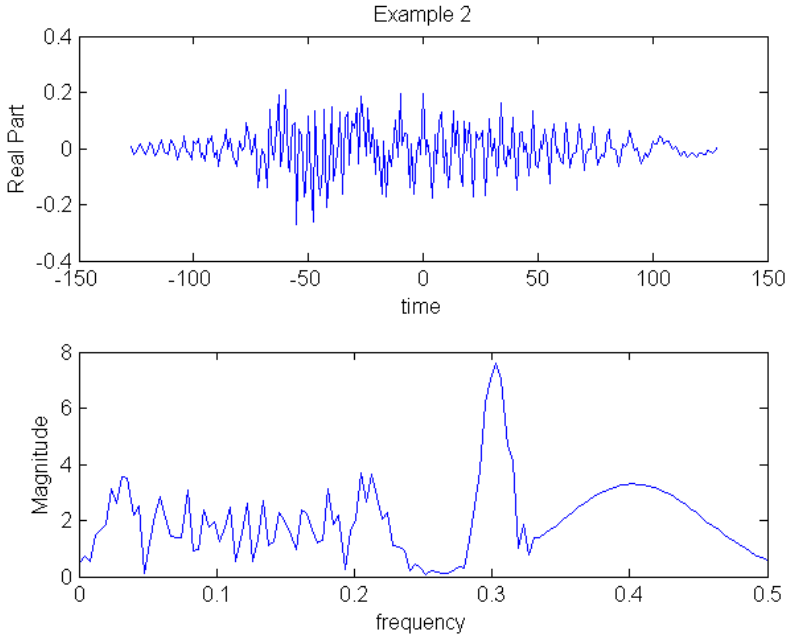


Figure B.2: Time domain representation (top) and the Fourier transform (FFT) of the Example signal 2 where the frequency is normalized to sampling frequency.

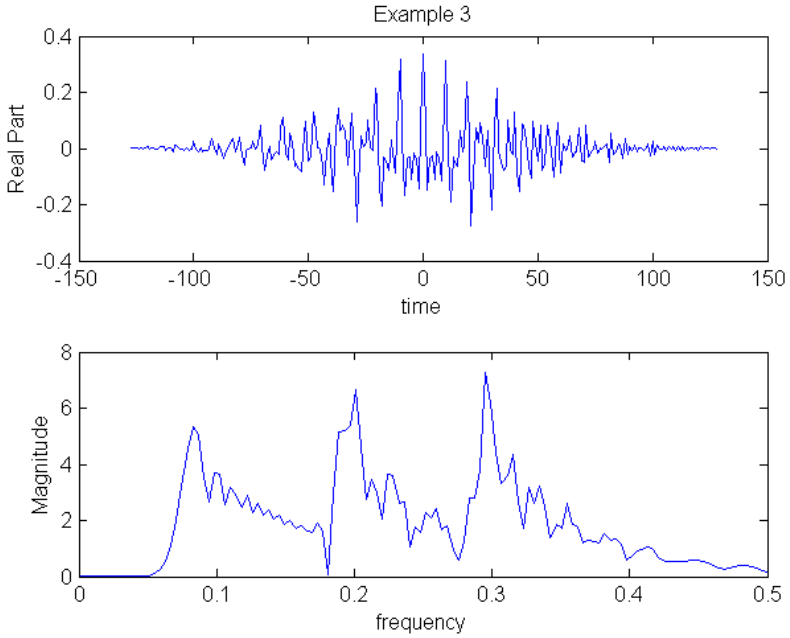


Figure B.3: Time domain representation (top) and the Fourier transform (FFT) of the Example signal 3 where the frequency is normalized to sampling frequency.

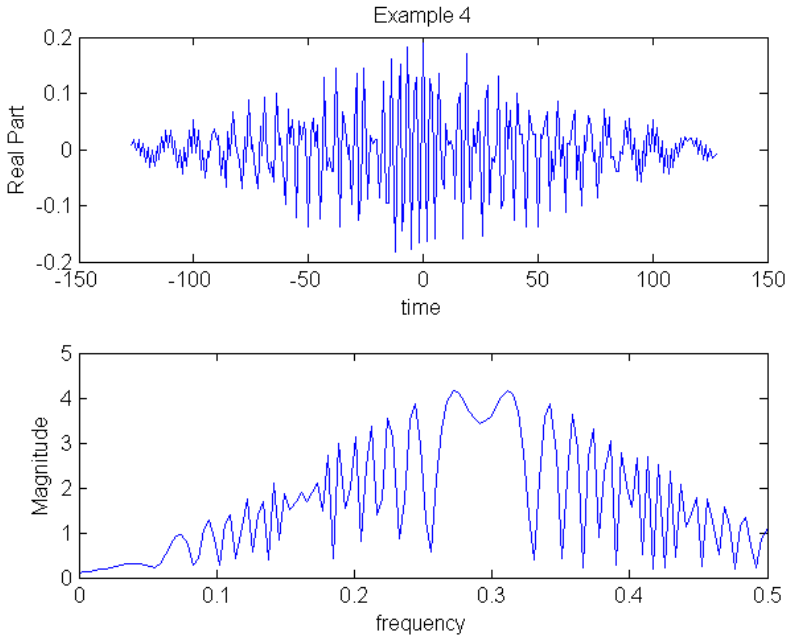


Figure B.4: Time domain representation (top) and the Fourier transform (FFT) of the Example signal 4 where the frequency is normalized to sampling frequency.

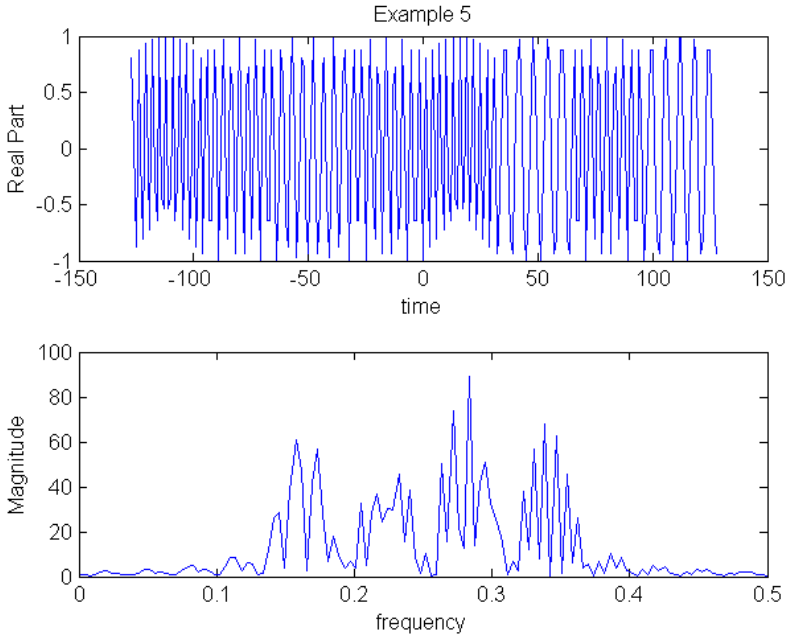


Figure B.5: Time domain representation (top) and the Fourier transform (FFT) of the Example signal 5 where the frequency is normalized to sampling frequency. The signal is a segment from a Frequency Hopping MFSK signal.

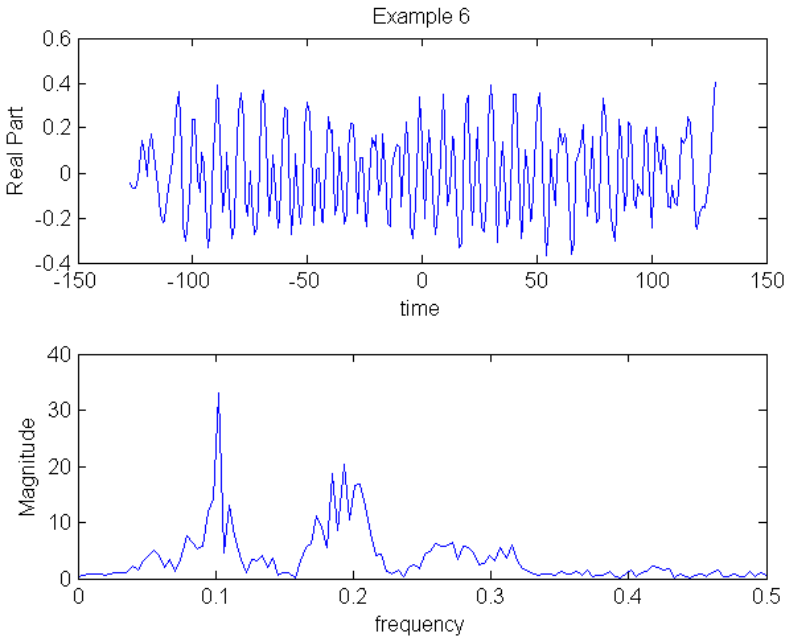


Figure B.6: Time domain representation (top) and the Fourier transform (FFT) of the Example signal 6 where the frequency is normalized to sampling frequency. The signal was taken from a dolphin's click-signal segment

# Appendix C

# Hadamard Product

Given any two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, with the same dimensions, their Hadamard or element-vise product, $\boldsymbol{A} \bullet \boldsymbol{B}$ is a matrix of the same dimension given by

$$(\boldsymbol{A} \bullet \boldsymbol{B})_{ij} = \boldsymbol{A}_{ij}\boldsymbol{B}_{ij} \tag{C.1}$$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \bullet \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} & \cdots & a_{1n}b_{1n} \\ a_{21}b_{21} & a_{22}b_{22} & \cdots & a_{2n}b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}b_{m1} & a_{m2}b_{m2} & \cdots & a_{mn}b_{mn} \end{bmatrix} \tag{C.2}$$