New algorithms for the analysis of signals with time varying instantaneous frequency

Wei, Yongmei

2006

Wei, Y. (2006). New algorithms for the analysis of signals with time varying instantaneous frequency. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/3448>

<https://doi.org/10.32657/10356/3448>

Nanyang Technological University

Downloaded on 26 Apr 2023 18:45:34 SGT

New Algorithms for the Analysis of Signals with Time-varying Instantaneous Frequency

WEI Yongmei

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2006

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

.....
Date

.....
Wei Yongmei

Acknowledgment

I would like to express my deepest gratitude to my supervisor, Dr. Guoan Bi, for his direction, kindness and many invaluable comments and suggestions throughout the whole research project in both the academic and non-academic aspects.

I also take the opportunity to acknowledge my family members and my boy friend who have shown so much concern and care for me in the course of my studies. Their encouragement and unyielding support have contributed to the completion of this thesis.

Thanks are also given to the Nanyang Technological University for the award of a research scholarship which enabled me to undertake this research.

Contents

Acknowledgements	i
Contents	ii
Summary	vi
List of Figures	ix
List of Tables	xii
List of Abbreviations	xiii
1 Introduction	1
1.1 Motivation	1
1.1.1 Parameter estimation of polynomial phase signals	3
1.1.2 The excision of the polynomial phase interference	5
1.1.3 Time frequency analysis of time-varying signals	8
1.2 Objectives	9
1.3 Major contribution of the Thesis	10

1.3.1	Fast polynomial time frequency transform	10
1.3.2	The interference excision by MDPFT in DS-SS system	11
1.3.3	The efficient realization of the modified LPTFT	13
1.4	Organization of the Thesis	15
2	Background	17
2.1	The PPS and its CRLB	17
2.2	Polynomial time frequency transform	18
2.2.1	Definition of PTFT	19
2.2.2	The detection of PPSs with PTFT	22
2.3	High order ambiguity function	23
2.3.1	High order ambiguity function	23
2.3.2	Non-identifiability of HAF and the Product HAF	25
2.4	Discrete polynomial Fourier transform	27
2.4.1	Definition of DPFT	27
2.4.2	Comparison	28
2.5	Time frequency transforms	29
2.5.1	The traditional linear TFT - short time Fourier transform . .	31
2.5.2	Nonlinear TFTs	32
2.5.3	Local polynomial time frequency transform	34
3	Fast maximum likelihood estimation of polynomial phase signals	35
3.1	Review on the parameter estimation of PPSs	36

3.1.1	Sub-optimal maximum likelihood methods	37
3.1.2	Maximum likelihood methods	39
3.2	Advantages of PTFT	41
3.3	DIT radix-2 FFT and the fast algorithm for the 2nd order PTFT . .	45
3.3.1	Decimation-in-time radix-2 FFT	45
3.3.2	The fast algorithm for the 2nd order PTFT	47
3.4	The fast algorithm for the 3rd order PTFT	48
3.4.1	The fast algorithm	49
3.4.2	Computational Complexity	53
3.5	The generalized fast algorithm for PTFT	55
3.5.1	Quasi-periodic property	55
3.5.2	Key property	56
3.5.3	The fast algorithm	58
3.5.4	Computational complexity	59
3.6	The experiment	62
3.7	Conclusion	62

4 Modified DPFT and its application for the interference excision in

	DS-SS system	64
4.1	Introduction	64
4.1.1	DS-SS communication system	64
4.1.2	Problem description and previous work	65
4.2	The modified DPFT-based excision method	72

4.2.1	DPFT-based method and the side lobe distortion	73
4.2.2	Modified DPFT-based excision method	76
4.2.3	Performance analysis	79
4.3	Narrowband interference excision	85
4.3.1	Performance analysis	87
4.3.2	Simulation results	93
4.4	Broadband interference excision	96
4.4.1	Performance Analysis	99
4.4.2	Multi-Component PPI	100
4.4.3	Simulation Results	102
4.5	Conclusion	109
5	The efficient analysis of time-varying signals with LPTFT	110
5.1	Introduction	110
5.2	Modified LPTFT for multi-component signals	116
5.3	Robust modified LPTFT	118
5.3.1	Robust FT	120
5.3.2	Robust modified LPTFT _p	122
5.4	Segmentation	123
5.5	The efficient realization	124
5.5.1	Estimation of $\mathbf{L}(n)$	125
5.5.2	Window Length Estimation	130
5.6	Experimental results	134

5.7 Conclusion	140
6 Conclusion and Future work	141
6.1 Conclusion	141
6.2 Recommendations for further research	143
6.2.1 Estimation of PPSs	143
6.2.2 PPI excision in spread spectrum communication systems . . .	144
6.2.3 Time frequency analysis of time-varying signals	145
Author's Publication List	146
Bibliography	148
Appendices	166
A Derivation of the output SNR of PTFT	166
B Proof of the quasi-periodic property	168
C Proof of the key property	170

Summary

This thesis aims to develop new digital signal processing algorithms for the analysis of signals, which contain time varying frequencies, with applications in wireless communications.

The first scheme contributes to a fast algorithm for the computation of polynomial phase time frequency transform (PTFT), which is the key transform for the maximum likelihood estimation of polynomial phase signals (PPSs). The new fast algorithm is derived by using two important properties. The quasi-periodic property explores the periodicity of the PTFT to reduce the total number of the outputs to be computed, and the key property decimates along each dimension of the PTFT to remove the inherent redundancy when the PTFT is computed by using the fast Fourier transform (FFT). A computational architecture similar to that of the radix-2 FFT is also proposed for an easy implementation of the proposed fast algorithm. With the fast algorithm, it is expected that the use of the maximum likelihood estimation can be widely employed for various applications to provide better statistical performances.

The second scheme is to excise PPSs based on modified discrete polynomial Fourier transform (MDPFT). This scheme is particularly designed for the application of the interference excision in direct sequence spread spectrum (DS-SS) communication systems since the interferences modeled as PPSs cover a large range of interference for practical applications. The key idea of MDPFT is to introduce

an extra fractional parameter compared with the traditional discrete polynomial Fourier transform (DPFT). With MDPFT, the undesirable influence brought by the side lobes is minimized for a better performance in terms of bit error rates. It is shown from both theoretical and simulation results that this method can be used to effectively excise both mono- and multi-component interferences which are modeled as PPSs. The computational complexity of MDPFT-based excision method is low and particularly appealing for real-time applications.

The third proposed scheme deals with the time-varying signals including those excluded by the category of PPSs. An efficient algorithm for the analysis of multi-component time-varying signals is proposed based on modified local polynomial time frequency transform (LPTFT) and robust modified LPTFT. The signals to be analyzed are divided into a number of segments and the desired parameters for computing modified LPTFT in each segment are estimated from PTFT in the frequency domain. Compared with other reported algorithms, the length of overlap between the consecutive segments is reduced to minimize the overall computational complexity. The concept of adaptive window lengths is also employed to achieve a better time-frequency resolution for each component. Numerical simulations with synthesized multi-component signals show that the proposed ones achieve a better performance on instantaneous frequency estimation with greatly reduced computational complexity.

List of Figures

2.1	PTFT of a 2nd order PPS.	21
2.2	HAF of a two-component 2nd order PPS.	25
2.3	Comparison between PHAF and HAF.	26
2.4	Comparison between DPFT and DFRFT of a linear chirp signal. . . .	28
3.1	Comparison of variances between PTFT and HAF of a PPS.	43
3.2	Output SNR comparisons.	44
3.3	Signal flow graph of the fast algorithm for the 3rd order PTFT. . . .	53
3.4	Comparison of computation time.	63
4.1	Block diagram of DS-SS system.	65
4.2	Block diagram of IF estimation approach.	67
4.3	Block diagram of interference synthesis approach.	69
4.4	Block diagram of transform based approach.	69
4.5	The windowed interference in the DPFT domain.	76
4.6	Comparison between BERs of the MDFT-based excision methods.	89
4.7	Block Diagram for simulations.	93

4.8	Comparison of BERs for interferences with different grid biases.	94
4.9	Comparisons of BERs achieved by the MDFT-based method.	95
4.10	Comparison of BERs achieved by different excision methods.	97
4.11	MDCFT of the signal containing multiple chirp-like components. . . .	103
4.12	Comparisons of BERs for the interference with different grid biases .	103
4.13	Comparisons of BERs achieved by using different excision methods. .	104
4.14	BERs for signal containing non-overlapping chirp interference.	106
4.15	The WVDs of the multi-component interference.	106
4.16	BER performance comparison.	108
5.1	The performance comparisons between different TFTs. (a) LPTFT with one set of parameters; (b) LPTFT with one set of parameters; (c) Modified LPTFT _p with two sets of parameters; (d) WVD.	119
5.2	Segmentation for $\alpha = 4, 2$ and 0 , and $Q = 5$	123
5.3	The PTFT of a sinusoidal FM signal with different overlap lengths. .	127
5.4	The 2nd order PTFT of a segmented sinusoidal FM signal.	128
5.5	The 2nd order PTFT of a segmented signal.	129
5.6	Comparisons between PTFT and robust PTFT of a segmented signal.	131
5.7	The comparisons of MSEs between modified LPTFT _p s.	135
5.8	The comparisons of MSEs between modified LPTFT _p s.	135
5.9	The comparisons of MSEs between modified LPTFT _p s.	136
5.10	The comparison between modified LPTFT _p s.	137
5.11	The comparison for the signal with α stable impulse noises.	138

5.12 The comparison for the signal with impulse noises. 139

5.13 The comparison for signals with α stable impulse noises. 140

List of Tables

3.1	The ratio of asymptotic variances between HAF-based methods and PTFT-based methods	42
4.1	Comparison of the computational complexity	99

List of Abbreviations

ABBREVIATIONS	FULL EXPRESSIONS
AM	Amplitude modulated
BER	Bit error rate
CDMA	Code division multiple access
CRLB	Cramer-Rao low bound
DCFT	Discrete chirp Fourier transform
DFRFT	Discrete fractional Fourier transform
DFT	Discrete Fourier transform
DPFT	Discrete polynomial Fourier transform
DS-SS	Direct sequence spread spectrum
FFT	Fast Fourier transform
FIR	Finite impulse response
FM	Frequency modulated
FRFT	Fractional Fourier transform

FT	Fourier transform
GTBP	Generalized bandwidth product
HAF	High order ambiguity function
HIM	High order instantaneous moment
IDCFT	Inverse discrete chirp Fourier transform
IF	Instantaneous frequency
IGAF	Integrated generalized ambiguity function
IMDFT	Inverse modified discrete Fourier transform
IMDPFT	Inverse modified polynomial Fourier transform
ISNR	Interference to noise and signal energy ratio
ISR	Interference to signal energy ratio
JEM	Jet engine modulation
LMS	Least mean square
LPTFT	Local polynomial time frequency transform
MDFT	Modified discrete Fourier transform
MDPFT	Modified discrete polynomial Fourier transform
ML	Maximum likelihood
MLE	Maximum likelihood estimation
MLPTFT	Modified local polynomial time frequency transform
MSE	Mean square error
OFDM	Orthogonal frequency division multiplexing
PHAF	Product high-order ambiguity function

PLWD	Polynomial L Wigner-Ville distribution
PPI	Polynomial phase interference
PPS	Polynomial phase signal
PN	Pseudorandom
PTFT	Polynomial time frequency transform
PWVD	Polynomial Wigner-ville distribution
RFT	Robust Fourier transform
RID	Reduced interference distribution
RMLPTFT	Robust modified local polynomial time frequency transform
RPTFT	Robust polynomial time frequency transform
SAR	Synthetic aperture radar
SNR	Signal to noise energy ratio
SPWVD	Smoothed pseudo Wigner-Ville distribution
SS	Spread spectrum
STFT	Short-time Fourier transform
SWVD	Smoothed Wigner-Ville distribution
TFD	Time frequency distribution
TFT	Time frequency transform
WHT	Wigner-Hough transform
WVD	Wigner-Ville distribution

Chapter 1

Introduction

1.1 Motivation

Experiments have shown that many natural signals, such as human voices, sounds of birds, bats and whale whistles, exhibit amplitude modulated (AM) and/or frequency modulated (FM) properties. In general, signals employed in various areas, including communications, image analysis, sonar and radar signal processing, also exhibit the properties of AM and/or FM, which have the form of

$$x(t) = a(t)e^{j2\pi\phi(t)}, \quad (1.1)$$

where $a(t)$ is the amplitude and $\phi(t)$ is the phase of the signal. Another important parameter of AM-FM signals is the instantaneous frequency (IF) $f(t) = \frac{1}{2\pi} \frac{d\phi(t)}{dt}$ defined as the derivative of the phase. Usually, $f(t)$ is a continuous function varying with time t and $x(t)$ is called as a time-varying signal. In this thesis, we mainly focus on the AM-FM signals with concentrated IF on the time frequency plane.

According to the Weierstrass approximation theorem [27], the phase $\phi(t)$ of $x(t)$ can be approximated by a polynomial function over a closed interval if $\phi(t)$ is a continuous function of t . Thus, time-varying signals with polynomial phases, also known as polynomial phase signals (PPSs), are of great importance. It has been reported that the PPSs have been used widely in various applications including audio analysis [28], biomedical engineering [120], filtering applications [35,62], image processing [2, 23, 70, 87, 106, 114] and communications [12, 16, 39, 41, 43, 67, 74, 79]. Other applications are also found in sonar and radar applications, which include modeling sonar echoes and the sounds of bats [8], identifying the type of radars in passive electronic intelligence systems [85], and estimating the range between targets and radar in pulse compression radar systems [100].

Although PPSs are good models used in many applications, the time-varying signals that cannot be directly modeled as PPSs also arise in a variety of applications. One example is the hyperbolic FM signal which is a good model for dolphin and whale whistles [43]. Hyperbolic FM signals also arise in the time varying underwater channel where the frequency may increase or decrease hyperbolically with time due to the dispersive property of ocean [78]. Another example is the sinusoidal FM signal. This kind of signals arises from the vibrating or rotating parts of the targets [38, 45] and also appears as a result of the so-called jet engine modulation (JEM) phenomenon [15, 115].

Thus, it is of great significance to develop better signal processing techniques including the analysis, synthesis, estimation and excision for time-varying signals in

various application fields. In this thesis, we focus on the following three areas.

1.1.1 Parameter estimation of polynomial phase signals

In many applications, the parameter estimation of PPSs is of great importance because these parameters carry desirable information.

In wireless communications, PPSs are used to model the channel, for example, in a non-geosynchronous satellite system, a typical burst mode communication or a highly mobile peer-to-peer context [39]. The channel parameters, such as the doppler rate and the doppler shift, are decided by the parameters of received signals which are modeled as PPSs if narrow-band linearly modulated signals are transmitted through a time-varying Rician channel [11, 39]. Generally, the accuracy of the estimation for these parameters directly influences the performance of the optimization of equalizers and transmitters. In radar applications, the parameter estimation of PPSs is also required for identifying radar type in passive electronic intelligence systems [96], estimating the kinematic parameters of moving targets [101] and compensating the motion effect in synthetic aperture radar which is aimed at imaging the earth from a satellite or an aircraft [11]. Recently, there is an increasing demand for high resolution estimation to recognize moving targets under different scenarios. e.g. hidden by foliage [11, 101]. In active sonar system, by critical estimating the parameters of the echoes which are modeled as PPSs, the navigation, detection, and classification of small insects can be realized in a highly cluttered time-varying environment [8]. In image processing, the parameters of PPSs are related with the

motion parameters, such as the velocity and the acceleration [2, 106].

As a result, estimation of the parameters of PPSs has been extensively studied and a variety of different techniques have been proposed [1, 11, 17, 51, 75, 85] in the past twenty years. It has been acknowledged by many authors [1, 51, 85, 118] that parametric methods can provide unlimited resolution compared with non-parametric methods [14, 24] which employ phase-locked loop or LMS and RLS adaptive filters tracking the phase of the signals [22], or the time frequency transform to estimate the IF of the signal. In the parametric category, there mainly exist maximum likelihood (ML) methods based on polynomial phase time frequency transform (PTFT) [1, 17, 48, 75, 84, 118] and sub-optimal ML estimation methods such as those based on high order ambiguity functions (HAF) [85] and their variations [11, 51, 84].

In Chapter 3 of the thesis, we focus on the ML estimation methods because they achieve better statistical performances in terms of lower estimation variances and higher output signal to noise energy ratio (SNR) for high order PPSs even under low input SNR compared with those sub-optimal methods [1, 17, 75, 84]. These advantages are important for applications where the input SNR is limited by the power of the transmitter or higher order of PPSs modeling is required. For example, longer synthetic apertures (or equivalently longer observation interval) in synthetic aperture radar (SAR) are required to obtain high resolution at low frequencies to recognize targets moving on the ground and hidden by foliage [11, 101]. In this case, the 2nd order PPSs are not appropriate for the signals observed in a longer period and higher order PPSs are required [11, 101].

It has been shown that the key transform used in ML methods is polynomial time frequency transform (PTFT) [1, 17, 48, 75, 84, 118]. Although the algorithms of fast Fourier transform (FFT) are available for use, the PTFT requires a huge computational complexity to deal with multidimensional computation. Fast quadratic phase transform [48] was proposed as an efficient algorithm for the 2nd-order PTFT to reduce the multiplicative complexity by a factor of $\log_2 N$ compared with that needed by directly using the FFTs. Unfortunately, there has not been a fast algorithm to support higher order PTFT in the literature. Therefore, finding a fast algorithm supporting the $(M + 1)$ th order PTFT becomes a critical issue for many real time applications. It is observed that one dimension can be exploited to achieve reduction of the computational complexity by using FFT algorithms and a limited saving has been achieved by using FFT to compute PTFT. Inspired by the idea of decimation-in-time FFT algorithm, we exploit the computational redundancy along each dimension of the PTFT and propose a fast algorithm for the computation of PTFT of any order.

1.1.2 The excision of the polynomial phase interference

Direct sequence spread spectrum (DS-SS) technique has been widely used in various communication systems. One well known commercial example is the code-division multiple-access (CDMA) cellular system. One of the most useful advantages of DS-SS systems is its capability of interference mitigation. However, it becomes difficult when the narrow-band interference is with high energy or the interference

becomes broadband [72, 76]. Polynomial phase interference (PPI) is a kind of the most important interference and has attracted much attention in recent years [4, 6, 10, 50, 64, 69, 111, 121, 122]. PPI includes both narrow band tone interference [69, 72, 121, 122] and a large class of broadband interference such as linear chirp interference [4, 6, 10, 50, 64, 111]. This class of interference is encountered often in many applications. For example, this kind of interference may appear if the DS-SS signal is transmitted in a band containing PPS transmission to enhance the security [98], which is common since many signals used in communications, radar, sonar systems and generated by natural beings can be modeled as PPSs [24, 85]. Another possible scenario arises for commercial CDMA or satellite overlay applications. The wide band CDMA or mobile user transmission must be demodulated in the presence of transmitted signals from co-channel users [39, 98], which may be modeled as PPSs [12, 16, 41, 67]. From the jammer point of view, the PPI is desired [10] since it has a constant modulus to achieve the maximization of the average transmitted energy for a given peak transmission power.

Among the numerous reported excision methods, the transform domain based excision technique [3, 26, 29, 65, 72, 76, 94, 97, 98, 109, 110, 121, 122] is the most popular framework for the interference excision because it has a simple concept and can be easily implemented. It usually transforms the received signal into another domain where the interference can be distinguished from the desired signal. The interference, thus, can be easily suppressed in that domain. A variety of transforms such as FT [72, 121, 122], short time FT [76], Wigner-Ville distribution [64] and lapped

transforms [69], have been used for the PPI excision.

However, window functions are generally used in the transform domain based excision methods to localize the signal in time. In general, the windowing operation produces side lobes (also called as frequency dispersion). The existence of the side lobes results in significant deterioration of the system performance because it is difficult to distinguish PPI components in the side lobes from the desired signal. Traditional solution is to select non-rectangular windows with low side lobes [97,121,122]. It is found that the optimal window is not fixed and varies according to the spectrum of the PPI in the narrowband case [121,122]. There are several disadvantages or unsolved problems for using non-rectangular windows in the transform-domain based excision methods. Firstly, it is well known that windows with lower side lobes have broader main-lobes. As a result, more components in the transform domain have to be excised, which unavoidably decreases the energy of the desired signal at the same time. Secondly, adaptive demodulation is required because non-rectangular window brings correlation to the pseudo-random sequences [97]. The use of adaptive demodulation complicates the receiver hardware and increases the total computational cost [98].

The most significant disadvantage is that the performance of the transform-based methods, including time-weighting transform-based methods, is sensitive to the parameters of the PPI. This is because, for a particular kind of window, the interference becomes most concentrated when the product of the initial frequency of PPI and the processing length is an integer, and the interference becomes less

concentrated with the increase of the fractional difference between the product and its nearest integer. When the product is an integer, it is observed that the use of rectangular window turns the interference into a single line, which is more concentrated than using other windows. In this way, the undesirable influence brought by side lobes is minimized. This observation motivates an investigation into a new technique to employ the difference in the excision process to reduce the sensitivity of the traditional transform-based method and enhance the overall performance in terms of bit error rates.

1.1.3 Time frequency analysis of time-varying signals

Time frequency transforms (TFTs) are the most effective tools to deal with time-varying signals. TFTs describe how the frequency content of a signal changes in time. There are mainly two types of TFTs which are linear TFTs and nonlinear TFTs, respectively. Each type of TFTs has its own advantages and disadvantages. For example, nonlinear TFTs have undesirable cross-terms which deteriorate their performances significantly compared with linear TFTs. On the other hand, they provides much higher resolution than linear TFTs.

Recently, a TFT, referred as local polynomial time frequency transform (LPTFT), has been proposed [53] with a set of extra parameters to approximate the phases of the segmented signals as polynomial functions. It is based on the idea that the analyzed signals can be modeled as PPSs in a local approximation due to the Weierstrass approximation theorem mentioned before, although PPSs cannot be good models

for these signals directly. LPTFT belongs to linear time frequency transform (TFT). Compared with the traditional linear TFTs, the LPTFT can provide high resolution for any kind of signal only if the extra parameters are estimated and updated correctly [55].

Unfortunately, the estimation and update of the extra parameters need huge computational costs and prevent the LPTFT from many practical applications. The computational cost is mainly due to the fact that the consecutive signal segments, on which the estimation process is implemented, are heavily overlapped because a signal segment is produced whenever the window is slid by one data point [30]. In order to reduce the computational complexity, attempts are made to reduce the length of overlap between the consecutive segments. Based on the idea of reducing the overlap length, effective methods of parameter and window estimation from signal segments are required to obtain a good compromise between the smoothness and resolution of LPTFT in the time frequency domain.

1.2 Objectives

This thesis aims to develop new algorithms concerning various aspects of time-varying signals with applications in the above mentioned three areas. Firstly, fast polynomial time frequency transform (PTFT) is to be developed to achieve a significantly reduced computational complexity for the parameter estimation of PPSs. Secondly, it is expected to develop new transforms utilizing an extra parameter representing the difference mentioned above for the narrowband and broadband in-

interference excision in DS-SS system to minimize the undesirable influence of the side lobes existed in the traditional transform domain based methods. Thirdly, an efficient realization of the local polynomial time frequency transform (LPTFT) is to be developed to reduce the computational complexity for the estimation of the parameters needed by computing LPTFT.

1.3 Major contribution of the Thesis

1.3.1 Fast polynomial time frequency transform

A general fast algorithm for PTFT of any order is proposed in Chapter 3. The numbers of both complex multiplications and additions are reduced by a factor of $2^M \log_2 N$ for N -point $(M + 1)$ th-order PTFT.

The fast algorithm exploits the intrinsic symmetric properties along each dimension of PTFT. Two properties of PTFT are discovered for the implementation of the fast algorithm. The quasi-periodic property indicates that the first half of the PTFT outputs along each dimension is either the same or the folded version as the second half along the same dimension of the PTFT. This property helps us to avoid computing about half the number of the PTFT points for each dimension. The key property is inspired by the concept of the decimation-in-time used in the fast algorithm of FT (FFT). A similar decimation process to that used in FFT is implemented along each dimension of PTFT. It is found that the larger index value along each dimension of PTFT at higher stages in the decimation process of FFT

can be directly computed by the smaller index value of PTFT at lower stages in the decimation process. In this way, the computation with larger index value along each dimension of PTFT is not required. Furthermore, the butterfly implementation structure of the fast algorithm, which is similar to that of FFT, is also proposed for an easy implementation.

1.3.2 The interference excision by MDPFT in DS-SS system

Since significant performance deterioration occurs when the product of the initial frequency of PPI and the processing length is not an integer, modified discrete polynomial Fourier transform (MDPFT) is proposed for the excision of PPSs achieving the minimum frequency dispersion of the interference. With an additional fractional parameter, the linear transform-MDPFT is developed to provide optimal concentration capability regardless of the parameters of PPI. Thus, the excision performance in terms of BER by using MDPFT is improved considerably. Furthermore, the computational cost of the proposed method is in the order of $N \log_2 N$ with N being the length of the processing blocks. This computational cost is comparable with the traditional FFT-based excision method for the tone interference excision. For the higher order PPI excision, the computational cost is even much less than that of the other existing time-frequency transform or filter-based methods [7, 21, 64], which are generally in the order of $N^2 \log_2 N$. This makes the proposed method more appealing for real time applications. The details of the performance analysis for the proposed method is given to verify the performance enhancement by deriving the

closed form of the bit error rate (BER) with respect to the estimation performance of the extra fractional parameter and the parameters of the PPI. Two special cases of the proposed MDPFT-based methods are analyzed in detail. They are the modified discrete Fourier transform (MDFT) for the excision of the narrowband PPI and the modified discrete chirp Fourier transform (MDCFT) for the excision of a kind of broadband PPI, called as linear chirp interference.

For the narrowband case, traditional FFT-based excision methods lack unified solutions for the decision of the optimal number of the excised components in the transform domain since the optimal number is sensitive to the parameters of PPI. In our proposed method, it is shown that removing the maximum component in the transform domain generally yields the best performance for the excision of mono-component interference. This is because most energy of the interference is confined in one MDFT bin when the interference to noise energy ratio (ISR) is in a particular range. When ISR is out of this range, the optimal number of the excised components in the transform domain is found to be increased with the energy of the interference. The closed form of this range is derived and the optimal number of the excised components is decided by comparing the amplitude of the interference with a pre-computed threshold. Since the performance of the proposed method deteriorates significantly when ISR is out of the range, it is expected to increase the processing block to enhance the estimation precision of the parameters of the PPI instead of increasing the number of components excised. For the broadband case, the performance of the proposed MDCFT-based excision method is derived with respect to

the estimation performance of the chirp rate and the initial frequency of PPI. The proposed MDCFT-based excision method is also extended to the multi-component PPI excision, which is quite difficult for the other existing methods. It is found that there are two cases for multi-component PPI, which are overlap and non-overlap in the DCFT-domain. Two techniques are developed to deal with these two cases to achieve a good compromise between the computational cost and the performance, respectively. It is shown that the excision performance for the non-overlap multi-component interference is generally better than that of the overlap case. This is because excising one component brings less influence to the other component for the non-overlap case. Experiments show that the proposed method can produce a good performance for mitigation of mono- and/or multi-component interferences.

1.3.3 The efficient realization of the modified LPTFT

Firstly, the traditional definition of LPTFT is extended to modified LPTFT and robust modified LPTFT to deal with multi-component signals and signals embedded in impulse noises, which can not be handled by the traditional LPTFT. The modified LPTFT is formed as the numeric mean of several LPTFTs with parameters matching for each component of the analyzed signals to achieve high resolution. The robust modified LPTFT uses robust FT instead of FT to combat with the severe influence brought by impulse noises.

Secondly, the concept of segmentation is introduced to reduce the huge computational cost of the traditional method by N times where N is length of processing

block. It is found that the computational load of the traditional method for the computation of LPTFT comes from the largest number of segments to be processed due to the use of the overlapping consecutive segments. We propose an efficient realization of modified LPTFT and robust modified LPTFT by reducing the overlap length between consecutive segments to minimize the number of segments to be processed. Effective methods of estimating the extra parameters are presented by using PTFT and robust PTFT for the computation of modified LPTFT and robust modified LPTFT, respectively.

Thirdly, a new window selection method is proposed to improve the resolution of the traditional method. The length of the windows used in modified LPTFT and robust modified LPTFT is adaptively matched to the characteristics of the signal components to avoid the deterioration of resolution for each signal component. This is realized by selecting initial window length to be small enough to provide acceptable accuracy of the approximation followed by increasing the actual length of the windows according to the properties of consecutive signal segments. The reasons for this kind of window selection are two-fold. Firstly, the approximation errors increase with the window length if the order of the modified LPTFT and robust modified LPTFT is lower than that of the phase of the signal segment. Secondly, for polynomial phase component whose order is not higher than that of modified LPTFT and robust modified LPTFT, the modified LPTFT and robust modified LPTFT yield a better resolution if longer window (or segment) is used.

1.4 Organization of the Thesis

The rest of the thesis is organized as follows:

Chapter 2 provides detailed background of PPSs and the Cramer-Rao lower bound of the parameters of PPSs. Then, the properties of several PPS-related transforms including PTFT, HAF and DPFT are introduced, which are utilized for the estimation and excision of PPSs. A brief introduction and review of time frequency transforms, such as short time Fourier transform (STFT) and Wigner-ville distribution (WVD), are given in the last part of this Chapter.

In Chapter 3, the fast algorithm for the computation of PTFT is proposed. Firstly, the existing estimation methods for PPSs are reviewed in detail, followed by the advantages of the estimation method by using PTFT. Then, the concept of decimation-in-time FFT algorithm is reviewed for the computation of the 2nd order PTFT. The fast algorithm for the 3rd order PTFT is proposed in detail, together with the analysis of the required computational complexity. Finally, the fast algorithm for any other order PTFT is generalized and experiments are conducted to show that significant reduction of computing time is achieved by the proposed fast algorithm.

Chapter 4 is dedicated to the excision of the PPI in DS-SS systems. The MDPFT-based excision method is proposed with an extra parameter. The analytical performance in terms of bit error rate (BER) is derived to show the necessity and validity of the proposed excision method. The narrowband and broadband examples with theoretical performance analysis and experiments are given to show a

significant enhancement compared with the other existing methods. Furthermore, MDPFT-based methods for the excision of multi-component PPI are also provided.

In Chapter 5, an efficient realization of modified LPTFT is proposed. The modified LPTFT and robust modified LPTFT are developed from LPTFT to deal with multi-component signals embedded in Gaussian or impulse noises. The two major steps required by the computation of the modified LPTFT and robust modified LPTFT, which are the estimation of parameters and window length, are illustrated in detail. Experiments and comparisons with the traditional LPTFT are provided.

The conclusion and recommendation for future work are given in Chapter 6.

Chapter 2

Background

2.1 The PPS and its CRLB

The mono-component $(M + 1)$ th order PPS is expressed as

$$s(n) = A(n)e^{j2\pi \sum_{m=0}^{M+1} a_m n^m} + w(n), \quad n = 0, 1, \dots, N - 1, \quad (2.1)$$

where $A(n)$ represents the random time-varying amplitude and the parameters a_m , $m = 0, 1, \dots, M, M + 1$ are associated with the polynomial phase. $A(n)$ can be a constant or described as a real-valued stationary mixing process (not necessarily Gaussian), which is decided by the application, and $w(n)$ is assumed to be a white complex circular Gaussian process. It can be easily seen that $s(n)$ becomes a harmonic signal when $M = 0$. When $M = 1$ and $M = 2$, $s(n)$ becomes a linear FM and Quadratic FM signal, respectively.

Multi-component PPSs are also very common for applications, such as radar and communications, since multiple targets, multiple paths or multi-component interfer-

ence often occur in real applications and the number of the components is equal to the numbers of targets, interference or multiple paths [11, 59, 84]. An L -component $(M + 1)$ th order PPS is in the form of

$$s(n) = \sum_{i=1}^L A_i(n) e^{j2\pi \sum_{m=0}^{M+1} a_{i,m} n^m} + w(n), \quad n = 0, 1, \dots, N-1. \quad (2.2)$$

It was reported in [86] that the Cramer-Rao lower bound (CRLB) for a_m defined in (2.1) is

$$\begin{aligned} \text{Var}\{a_m\} \geq & \frac{1}{2N^{2m+1}\text{SNR}} \left(\frac{1}{2m+1} + \frac{(M+2)^2}{2N(m+1)^2} - \frac{1}{2N} + O(N^{-2}) \right) \\ & ((M+m+2)C_{M+m+1}^m C_{M+1}^m)^2, \end{aligned} \quad (2.3)$$

when $A(n)$ is constant. In (2.3), $C_M^m = \frac{M!}{(M-m)!}$ represents the binomial coefficients and SNR refers to the signal to noise energy ratio. It can be seen from (2.3) that the CRLB of a_m is inversely proportional to the SNR and the $(2m+1)$ th power of the signal points. Some special cases are given by

$$\text{Var}\{a_0\} \geq \left(1 + \frac{M^2 + 4M + 3}{2N} + O(N^{-2}) \right) \frac{(M+2)^2}{2N \text{SNR}}, \quad (2.4)$$

$$\text{Var}\{a_1\} \geq \left(\frac{1}{3} + \frac{M^2 + 4M}{8N} + O(N^{-2}) \right) \frac{[(M+1)(M+2)(M+3)]^2}{2N^3 \text{SNR}}, \quad (2.5)$$

$$\text{Var}\{a_{M+1}\} \geq (1 + O(N^{-2})) \frac{(2M+3)(C_{2(M+1)}^{M+1})^2}{2N^{2M+3} \text{SNR}}. \quad (2.6)$$

2.2 Polynomial time frequency transform

Polynomial time frequency transform (PTFT) is an important tool for the analysis of PPSs defined in (2.1). It has been shown that the maximum likelihood estimation

(MLE) of $\mathbf{a} = \{a_1, \dots, a_{M+1}\}$ for the PPSs defined in (2.1) is expressed as

$$\begin{aligned} \{\hat{a}_1, \dots, \hat{a}_{M+1}\} &= \max_{\theta} \left| \sum_{n=0}^{N-1} s(n) e^{-j2\pi \sum_{m=1}^{M+1} \theta_m n^m} \right|^2, \\ \hat{a}_0 &= \angle \sum_{n=0}^{N-1} s(n) e^{-j2\pi \sum_{m=1}^M \theta_m n^m}, \\ \hat{A} &= \left| \sum_{n=0}^{N-1} s(n) e^{-j2\pi \sum_{m=1}^M \theta_m n^m} \right| / N, \end{aligned} \quad (2.7)$$

if $A(n)$ is constant [1, 75, 84], and

$$\begin{aligned} \{\hat{a}_1, \dots, \hat{a}_{M+1}\} &= \frac{1}{2} \max_{\theta} \left| \sum_{n=0}^{N-1} s^2(n) e^{-j2\pi \sum_{m=1}^{M+1} \theta_m n^m} \right|^2 \\ \hat{a}_0 &= \frac{1}{2} \angle \sum_{n=0}^{N-1} s^2(n) e^{j2\pi \sum_{m=1}^M \theta_m n^m} \\ \widehat{A(n)} &= \left| \sum_{n=0}^{N-1} s(n) e^{j2\pi \sum_{m=1}^M \theta_m n^m} \right| / N, \end{aligned} \quad (2.8)$$

if $A(n)$ is assumed to be a real-valued stationary mixing Gaussian process with any structure [17]. In the above equations, $\theta = \{\theta_1, \dots, \theta_{M+1}\}$ and $(M+1)$ is the number of parameters. It is worth mentioning that (2.8) is a nonlinear least square estimation of $\mathbf{a}$ when $A(n)$ is not Gaussian. More information about the performance of the estimation of PPSs with time-varying $A(n)$ is available in [17].

2.2.1 Definition of PTFT

To obtain the MLE, $\theta = \{\theta_1, \dots, \theta_{M+1}\}$ has to be digitized. Thus, $(M+1)$ th order PTFT is defined as

$$\text{PTFT}_x^{M+1}(k, \mathbf{l}) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j2\pi \phi_1(k, \mathbf{l}, n)}, \quad (2.9)$$

where $x(n) = s^2(n)$ for time varying $A(n)$, $x(n) = s(n)$ for constant $A(n)$, and

$$\phi_1(k, \mathbf{l}, n) = (k/N)n + (l_1/N_1)n^2 + (l_2/N_2)n^3 + \cdots + (l_M/N_M)n^{M+1}, \quad (2.10)$$

$$\mathbf{l} = (l_1, l_2, \dots, l_M); \quad l_i = 0, \dots, N_i - 1; \quad k = 0, \dots, N - 1.$$

where l_i/N_i is the discrete form of θ_i , where N_i controls the quantization error introduced by digitization. For each l_i in (2.10), where $1 \leq i \leq M$, N_i is the total number of values of l_i and N is assumed to be 2^q with a positive integer q . Without loss of generality, it is assumed that $N_j \geq N_i \geq N$ for $j > i$ to achieve a satisfactory accuracy of the parameter estimation [48]. Therefore, it is assumed that N_i/N is an integer for any i . For simplicity of presentation, let us define a demodulated sequence

$$y_q^{(1)}(n) = x(n) e^{-j2\pi\phi_2(\mathbf{l}, n)}, \quad (2.11)$$

where

$$\phi_2(\mathbf{l}, n) = (l_1/N_1)n^2 + (l_2/N_2)n^3 + \cdots + (l_M/N_M)n^{M+1}.$$

Thus, the PTFT of $x(n)$ is equivalent to the DFT of $y_q^{(1)}(n)$, which is expressed as

$$\text{PTFT}_x^{M+1}(k, \mathbf{l}) = \text{DFT}_n[y_q^{(1)}(n)], \quad (2.12)$$

and transforms a 1D sequence $x(n)$ into an $(M + 1)$ D $\text{PTFT}_x^{M+1}(k, \mathbf{l})$. The subscript n in (2.12) indicates that the DFT is performed in terms of index n and $\text{DFT}_n[x(n)] = \sum_{n=0}^{N-1} x(n)e^{-j2\pi\frac{nk}{N}}$. The PTFT can be used to estimate the parameters of both mono- and multi-component PPSs. For a mono-component PPS, the PTFT has a peak when the parameters $\{k/N, l_1/N_1, l_2/N_2, \dots, l_M/N_M\}$ match the

Figure 2.1 PTFT of a 2nd order PPS.

parameters $\{a_1, a_2, \dots, a_{M+1}\}$ in (2.1), respectively. For a multi-component PPS, the PTFT yields the same number of peaks as that of components in the PPS. Figure 2.1 indicates that the PTFT of a two-component PPS clearly shows the locations of two peaks even when the SNR is zero dB. Since $\mathbf{a}$ is estimated according to a defined grid, the performance of PTFT is influenced by the effect of quantization errors. It can be easily understood that sharp peaks appear in the PTFT if there exist sinusoidal signals in $y_q^{(1)}(n)$ defined in (2.11). Thus, the remaining frequency deviation caused by the quantization errors δ_i along each dimension is assumed to be much smaller than the frequency of the sinusoidal signals expressed as

$$\left| \sum_{i=1}^M n^i / (2N_i) \right| < \left| \sum_{i=1}^M n^i \delta_i \right| \ll |a_1| \leq 0.5, \quad n = 0, \dots, N-1, \quad (2.13)$$

since the maximum quantization error along dimension l_i of $\mathbf{l}$ is $1/(2N_i)$. Otherwise, the peaks may not be easily identified due to the quantization errors.

2.2.2 The detection of PPSs with PTFT

The detection problem for a constant-amplitude PPS in white Gaussian noise is described as:

$$H_0 : s(n) = w(n) \quad (2.14)$$

$$H_1 : s(n) = Ae^{j2\pi \sum_{m=0}^{M+1} a_m n^m} + w(n).$$

The generalized likelihood ratio detector [57] is derived by using the MLE of the parameters of PPSs, and the derivation of the test statistics is similar to that for harmonic signals [57]. The final test statistics $I(\hat{k}_0, \hat{\mathbf{l}}_0)$ is the squared peak modulus of the PTFT, where $\hat{\mathbf{l}}_0 = (\hat{a}_2 N_1, \dots, \hat{a}_{M+1} N_M)$ and $\hat{k}_0 = \hat{a}_1 N$, and H_1 is decided if

$$I(\hat{k}_0, \hat{\mathbf{l}}_0) = |\sqrt{N} \cdot \text{PTFT}_s^{M+1}(\hat{k}_0, \hat{\mathbf{l}}_0)|^2 > \gamma$$

where γ is the threshold decided by the number of samples N , the variances of the noise σ^2 and the probability of false alarm. It can be easily seen that the probability density function (PDF) of $I(\hat{k}_0, \hat{\mathbf{l}}_0)/(\sigma^2)$ is distributed as χ_2^2 under H_0 and as $\chi_2^2(\lambda)$ under H_1 where $\lambda = NA^2/(\sigma^2)$ [57]. As a result, the probability of false alarm P_{FA} is

$$\begin{aligned} P_{FA} &= 1 - (1 - P\{I(\hat{k}_0, \hat{\mathbf{l}}_0) > \gamma; H_0\})^{N \prod_{i=1}^M N_i} \\ &\approx 1 - (1 - N \prod_{i=1}^M N_i Q_{\chi_2^2}(\frac{\gamma}{\sigma^2})) \\ &= (N \prod_{i=1}^M N_i) e^{-\frac{\gamma}{2\sigma^2}}, \end{aligned} \quad (2.15)$$

and

$$\begin{aligned} P_D &= P\{I(\hat{k}_0, \hat{\mathbf{l}}_0) > \gamma; H_1\} \\ &= Q_{\chi_2'^2(\lambda)}\left(-\frac{\gamma}{\sigma^2}\right), \end{aligned} \quad (2.16)$$

where $Q_{\chi_2'^2(\lambda)}(x)$ represents the probability for satisfying the condition that a $\chi_2'^2$ distributed random variable with degree of freedom λ is larger than x . In summary, the detection performance is

$$P_D = Q_{\chi_2'^2(NA^2/\sigma^2)}\left(2\ln \frac{N \prod_{i=1}^M N_i}{P_{FA}}\right).$$

2.3 High order ambiguity function

2.3.1 High order ambiguity function

High order ambiguity function (HAF) is an effective transform to estimate the parameters of PPSs with constant amplitude and is defined as the FT of the high order instantaneous moment (HIM) [85]. Given a sequence $x(n)$, the HIM is defined recursively as

$$\begin{aligned} x_1(n) &= x(n), \\ x_2(n, \tau_1) &= x_1(n + \tau_1)x_1^*(n - \tau_1), \\ &\dots \\ x_M(n, \tau_{M-1}) &= x_{M-1}(n + \tau_{M-1})x_{M-1}^*(n - \tau_{M-1}), \\ x_{M+1}(n, \tau_M) &= x_M(n + \tau_M)x_M^*(n - \tau_M). \end{aligned} \quad (2.17)$$

Then, the HAF is defined as

$$P_{M+1}[x, f, \tau] = \frac{1}{N} \sum_{n=0}^{N-1} x_{M+1}(n, \tau_M) e^{-j2\pi f n}, \quad (2.18)$$

where $\tau = \{\tau_1, \dots, \tau_M\}$. If $x(n) = s(n)$, where $s(n)$ is a $(M+1)$ th order PPS defined in (2.1) with constant amplitude A , x_{M+1} is reduced to a harmonic with amplitude $A^{2^{M-1}}$, frequency $\widehat{f_{M+1}}$ and phase $\widehat{\phi}$, which are

$$\widehat{f_{M+1}} = (M+1)! \prod_i^M (\tau_i) a_{M+1}. \quad (2.19)$$

In this way, the parameter a_{M+1} can be estimated from the peak position of (2.18)

$$\widehat{a_{M+1}} = \frac{1}{(M+1)! \prod_i^M \tau_i} \operatorname{argmax}_f (P_{M+1}[x, f, \tau]). \quad (2.20)$$

By multiplying $e^{-j2\pi\phi(n)}$ ($\phi(n) = \widehat{a_{M+1}} n^{M+1}$) with $x(n)$, a PPS of order M is obtained. The above procedures are repeated to obtain the estimate of each a_m . The dynamic range of a_{M+1} is within $[-\prod_i^M \tau_i / (2(M+1)!), \prod_i^M \tau_i / (2(M+1)!)]$ [119,124] if it is estimated by HAF. Figure 2.2 shows the HAF of a two-component 2nd order PPS embedded with Gaussian noise. It can be easily seen from Figure 2.2 (a) that there are two peaks existing in the HAF representing the two components. Although the HAF is computationally efficient compared with the MLE realized by PTFT introduced in Section 2.2, HAF suffers from its own nonlinearity. For example, HAF cannot work well when the SNR is low and/or the order M is large due to the highly disturbing cross-terms introduced by the nonlinearity [86]. This problem can be easily seen by comparing Figure 2.1 with Figure 2.2 (b) where the SNR is the same in these two figures. Two peaks in Figure 2.2 (b) are not as distinct as in Figure 2.1 because HAF shows a much higher noise level due to its nonlinearity.

Figure 2.2 HAF of a two-component 2nd order PPS.

2.3.2 Non-identifiability of HAF and the Product HAF

It has been shown in [11] that if the highest order parameters $a_{i,M+1}$ of a multi-component PPS defined in (2.2) share the same value between any two or more components, spurious sinusoids which do not represent any component of the PPS arise in the HIM of $s(n)$ defined in (2.17) in addition to those indicating the components of the analyzed PPS. Thus, the spurious peaks lead to wrong estimation. This phenomenon is known as non-identifiability which is further shown in Figure 2.3 (a). The PPS used in this figure contains two components with parameters $a_{1,1} = 0.25$, $a_{1,2} = 0.001$, $a_{2,1} = 0.5$ and $a_{2,2} = a_{1,2}$. It can be seen from Figure 2.3 (a) that two spurious peaks arise when using the HAF.

To overcome this problem, the product of HAF using different sets of lags defined as PHAF [11] is proposed to deal with multi-component PPSs. With the assumption of R sets of lags $\tau^{(r)} = \{\tau_1^{(r)}, \dots, \tau_M^{(r)}\}$, where $1 \leq r \leq R$, the PHAF of $x(n)$ is defined

Figure 2.3 Comparison between PHAF and HAF.

as

$$P_{M+1}^{(R)}(x, f) = \prod_{r=1}^R P_{M+1}(x, \gamma_r f; \tau^{(r)}), \quad (2.21)$$

where

$$\gamma_r = \frac{\prod_{k=1}^M \tau_k^{(r)}}{\prod_{k=1}^M \tau_k^{(1)}}. \quad (2.22)$$

Because of the property that the locations of only the useful sinusoids in HIM are dependent with the lags, the scaling operation which is realized by using γ_r in the frequency domain of PHAF aligns the useful sinusoids. In contrast, the peaks of the spurious sinusoids fall in different locations after scaling. In this way, the useful peaks are enhanced and the spurious ones are suppressed. Figure 2.3 (b) shows the PHAF of the same signal used in Figure 2.3 (a). It can be easily seen from Figure 2.3 (b) that the spurious peaks have been suppressed effectively.

2.4 Discrete polynomial Fourier transform

2.4.1 Definition of DPFT

The $(M + 1)$ th order discrete polynomial Fourier transform (DPFT) is defined as

$$X_{\mathbf{l}}(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) e^{-j2\pi(\sum_{m=1}^M l_m n^{m+1} + \frac{kn}{N})}, \quad (2.23)$$

where $\mathbf{l} = \{l_1, \dots, l_M\}$. It should be mentioned that DPFT is different from PTFT in (2.9) in that $\mathbf{l}$ is fixed in that $\mathbf{l}$ is fixed input parameters. Thus DPFT is of one dimension and PTFT is of $M+1$ dimension. When $M = 0$, DPFT simply becomes DFT. When $M = 1$, DPFT can also be called as discrete chirp Fourier transform (DCFT) [48, 118]. DPFT has the optimal concentration for PPSs because DPFT turns a PPS into a single line, which is shown by the following property.

This important property is illustrated with an N -point PPS $x(n) = e^{j2\pi \sum_{m=0}^{M+1} a_m n^m}$. If $l_m = a_{m+1}$, $1 \leq m \leq M$ and $(a_0 N)_{\text{mod } N} = k_1$, $0 \leq k_1 \leq N - 1$, where k_1 is a positive integer, the DPFT of $x(n)$ becomes

$$X_{\mathbf{l}}(k) = \begin{cases} \sqrt{N}, & k = k_1 \\ 0, & k \neq k_1. \end{cases} \quad (2.24)$$

This property shows that when k and $\mathbf{l}$ in (2.23) match the parameters a_m of the PPS, its DPFT becomes a single peak value in the main lobe with other components being zeros. This can be seen from Figure 2.4 (a) which is the DPFT of a 2nd order PPS.

Figure 2.4 Comparison between DPFT and DFRFT of a linear chirp signal.

2.4.2 Comparison

It can be easily seen from the property in (2.24) that the 2nd order DPFT has the optimal concentration for the 2nd order PPS, i.e., linear FM signal. It is worth mentioning that there exists another transform which can concentrate linear chirp signal as well. It is the fractional Fourier transform (FRFT) [77] with the definition as

$$X_{\varphi}(u) = \begin{cases} k_{\varphi} \int_{-\infty}^{\infty} x(t) e^{(j\pi t^2 \cot \varphi - j2\pi ut \csc \varphi)} dt & \varphi \neq n\pi \\ x(u) & \varphi = 2n\pi \\ x(-u) & \varphi = (2n+1)\pi. \end{cases} \quad (2.25)$$

where $k_{\varphi} = \sqrt{1 - j \cot \varphi} e^{(j\pi u^2 \cot \varphi)}$.

FRFT is a generalization of the classical FT. When $\varphi = \frac{\pi}{2}$, FRFT corresponds to the classical FT. If it is assumed that a linear FM signal $x(t) = e^{j2\pi l_1 t^2}$, $X_{\varphi_0}(u) \sim \delta(u)$, where $\varphi_0 = \arctan(-1/l_1)$. However, the discrete FRFT (DFRFT) of a linear

chirp signal does not become a single line as DPFT when the input signal is localized with a window. Figure 2.4 (b) shows the DFRFT of the same signal used in Figure 2.4 (a). It can be seen that there exist not only one peak but also many small values around that peak.

This property indicates the advantage of DPFT over DFRFT for the application of the polynomial phase interference (PPI) excision in DS-SS system. This is because the interference turns into single line in the DPFT domain and it can be distinguished and excised easily from the desired signals. Whereas, those lines with small values around the peak of DFRFT are difficult to be distinguished and cause significant distortion to the desired signals. This will be further explained in detail in Chapter 4.

2.5 Time frequency transforms

Time frequency transforms (TFTs) describe how the instantaneous frequency (IF) of the signal varies with time. Generally, TFTs fall into two categories, which are linear TFTs and nonlinear TFTs according to whether the transform satisfies the linearity superposition principle. Linearity superposition states that the TFT of a signal which is a linear combination of several components are the combination of the TFTs of the same components. In the following, the discrete time form of the transforms are given with the assumption that the sampling frequency of the discrete data is normalized to be one Hz and τ takes integer values. When τ/d_m arises where d_m is an integer, the analyzed signal has to be oversampled by d_m times.

Linear TFTs, expressed as $T_x(n, f) = \sum_{\tau=-\infty}^{\infty} x(\tau)h(\tau; n, f)$, mainly include the short time Fourier transform (STFT) [24, 34, 68, 76, 80, 91] and the wavelet transform [24, 80, 91]. The performance of linear TFTs largely depends on the choice of window $h(\tau; n, f)$.

According to the nonlinearity of the transforms which is related to the number of the times the analyzed signal appears in the transform, the M th order nonlinear TFT can be written as

$$T_x(n, f) = \sum_{\tau_{1,1}=-\infty}^{\infty} \cdots \sum_{\tau_{1,M/2}=-\infty}^{\infty} \sum_{\tau_{2,1}=-\infty}^{\infty} \cdots \sum_{\tau_{2,M/2}=-\infty}^{\infty} \left\{ \prod_{m=1}^{M/2} x(t + \tau_{1,m})x^*(t + \tau_{2,m})K_T(\tau_{1,1}, \dots, \tau_{1,M/2}, \tau_{2,1}, \dots, \tau_{2,M/2}; n, f) \right\}. \quad (2.26)$$

When $M = 2$, it is the quadratic TFT which includes the Cohen's class [24, 80, 91], the affine class [80], the hyperbolic class [43, 78], the exponential class [80] and the power class [36, 40, 44, 108]. When $M > 2$, it becomes a high order TFT such as polynomial WVD (PWVD) [19] and L-WVD [102]. The disadvantage of nonlinear TFTs is that there are high oscillating cross-terms which severely degrade their performance [24, 80]. Take a quadratic TFT as an example, if the analyzed signal is written as $x(n) = x_1(n) + x_2(n)$, its $T_x(n, f)$ is

$$T_x(n, f) = T_{x_1}(n, f) + T_{x_2}(n, f) + 2Re\{T_{x_1, x_2}(n, f)\}, \quad (2.27)$$

where $T_{x_1, x_2}(n, f)$ is defined as the cross TFT between $x_1(n)$ and $x_2(n)$, and $Re\{x\}$ is the real part of x . The third term in (2.27) is the cross-terms which does not have a clear physical meaning. It can be easily seen that the number of the cross-terms increases with the nonlinearity of the transform. In the following, several basic

transforms in each category are introduced. Then, a newly developed transform called local polynomial time frequency transform is presented.

2.5.1 The traditional linear TFFT - short time Fourier transform

The basic idea behind STFT is straightforward. At each time instant n_0 , the spectrum is represented by the FT of a small segment of the signal $x(n)$ around n_0 . Mathematically, the STFT [91] of the signal $x(n)$ can be described as

$$\text{STFT}(n, f) = \sum_{\tau=-\infty}^{\infty} x(\tau)h^*(\tau - n) \cdot e^{-j2\pi f\tau}, \quad (2.28)$$

where $h(n)$ is a low pass filter and $\|h\|_2 = 1$. In conventional STFT, the window $h(n)$ is Hamming window, Hanning window or Gaussian function. STFT has been applied to different areas including system identification [34], speech pitch and format analysis [68], interference excision in spread spectrum communications [76]. Although STFT is widely used to process non-stationary signals, its performance is subject to the choice of the window $h(n)$. Generally, it is found to use shorter windows to acquire a better time resolution and longer windows to achieve a better frequency resolution. Unfortunately, the uncertainty theorem prohibits the existence of the windows with high resolutions in both the time and frequency domains. Furthermore, STFT is not suitable for the representation of those signals whose components are not parallel with the time or frequency axis in the time frequency plane.

2.5.2 Nonlinear TFTs

A. Quadratic TFTs - Wigner Ville distribution and Cohen's class

For a signal $x(n)$, the Wigner Ville distribution (WVD), which is the most basic TFT in Cohen's class, is defined as

$$W(n, f) = \sum_{\tau=-\infty}^{\infty} x(n + \frac{\tau}{2})x^*(n - \frac{\tau}{2})e^{-j2\pi f\tau}. \quad (2.29)$$

Although the WVD yields the optimum concentration for linear FM signal, it suffers from the cross-terms due to its nonlinearity. In order to suppress the undesired cross-terms, many alternatives, such as smoothed WVD (SWVD), smoothed pseudo WVD (SPWVD), and reduced interference distribution (RID) [116], have been proposed. In general, the solutions are to introduce various windows to filter out the oscillating cross-terms. With fewer cross-terms, these modified WVDs have been used in many applications, such as analysis of the biological signals [116]. Generally, these variations, together with WVD, constitute the Cohen's class defined by

$$C(n, f) = \sum_{\tau=-\infty}^{\infty} \sum_{\theta=-\infty}^{\infty} \sum_{u=-\infty}^{\infty} x(u + \frac{\tau}{2})x^*(u - \frac{\tau}{2})\Phi(\theta, \tau)e^{-j(\theta\tau - \theta u + 2\pi f\tau)}, \quad (2.30)$$

where $\Phi(\theta, \tau)$ is the kernel function. The properties of the TFTs in this class can be readily known by simply examining $\Phi(\theta, \tau)$.

B. High order TFTs - Polynomial WVD

The M th order PWVD of $x(n)$ is defined as

$$\text{PWVD}^M(n, f) = \sum_{\tau=-\infty}^{\infty} \left\{ \prod_{m=1}^{M/2} x(n + d_m\tau)x^*(n + d_{-m}\tau)e^{-j2\pi f\tau} \right\}, \quad (2.31)$$

where the order M of PWVD is an even integer and indicates the order of the non-linearity of PWVD, d_m and d_{-m} are real coefficients. PWVD is originally designed to yield maximal concentration around the IF for the PPS of order smaller than two [13, 19], and higher order PWVD is designed in [13] to provide high concentration for the PPS of order smaller than four. Generally, the computation of high order TFTs always requires interpolation of the original signals. The higher interpolation order, the more computation is required for computing PWVD.

In general, nonlinear TFTs suffer from cross-terms which are introduced by their nonlinearity. Thus, many techniques have been proposed to filter out the cross-terms in the time frequency domain. One kind of method is realized by using windows to filter the high oscillating cross-terms [24, 116]. Unfortunately, smoothing with windows usually brings deterioration to the frequency resolution of the original TFTs because we cannot get a window with high resolution in both time and frequency domains, especially when the time frequency structure of the analyzed signal is complex, such as highly nonlinear signals. Another class of methods developed for the suppression of the cross-terms is the s-method developed by L. Stankovic, et.al [25, 103, 105, 107]. This class of methods performs well when the components are well separated and its performance deteriorates when the signal components overlap in the time frequency plane [25, 103, 105, 107].

2.5.3 Local polynomial time frequency transform

Recently, the local polynomial time-frequency transform (LPTFT) has been developed as a generalized STFT with the form of [53]

$$\text{LPTFT}(n, f) = \sum_{\tau=-\infty}^{\infty} x(n + \tau)h(\tau) e^{-j2\pi[\sum_{m=2}^M l_{m-1}(n) \frac{\tau^m}{m!} + f\tau]}, \quad (2.32)$$

where $x(n)$ is the signal to be analyzed, $h(\tau)$ is the window function used at time n with length Q and $\mathbf{l}(n) = [l_1(n), \dots, l_{M-1}(n)]$ are the parameters related to the derivatives of the instantaneous frequency of the signal being processed [53]. LPTFT is used for non-stationary signal analysis and developed from the traditional STFT to enhance the resolution of the signal representation in the time frequency domain. The basic idea of LPTFT is to find an $(M - 1)$ th order polynomial function approximation in the frequency domain from the signal segment to determine the nonparametric characteristic of LPTFT [53]. The main processing costs of the LPTFT are for the estimation of both the time-varying parameter $\mathbf{l}(n)$ and window length Q . Compared with the traditional linear TFTs, LPTFT has much higher resolution and it has been shown that the LPTFT can yield high resolution of any time varying frequency provided that these parameters are properly estimated and updated [53]. Compared with nonlinear TFTs, LPTFT has much fewer cross-terms due to its linearity.

Chapter 3

Fast maximum likelihood estimation of polynomial phase signals

The estimation of PPSs is of great importance in a variety of applications including communications, radar and sonar signal processing. The following are some examples.

Time selective fading channel is very common in wireless communications. The parameters in time selective fading channels, such as the doppler shift and the doppler rate, cannot be neglected in most cases because the relative velocity of the mobile user or satellite is time varying [39]. For example, for a system with carrier frequency 2 GHz and a satellite with altitude 1000 km, the doppler shifts and doppler rates are as large as 42 kHz and 300 Hz/s, respectively. For a LEO

satellite operating at 2.4 GHz, the doppler shifts and doppler rates are as large as 62 kHz and 13 Hz/s. In these cases, the channel parameters are estimated from the received signals which are modeled as PPSs. In radar applications, the estimation of the parameters of PPSs is necessary in the following situations [86, 96]. The first application is in passive electronic intelligence systems where we are interested in identifying the radar type from the order of the received PPSs through estimating their parameters. The second one is the imaging of moving targets using synthetic aperture radar (SAR) [11, 49, 96, 101]. The relative motion between radar and target introduces a phase modulation characterized by PPSs on the received signal. A proper positioned and focused image of the target requires exact knowledge of the parameters of the PPSs. Similar scenario arises for the inverse synthetic aperture radar (ISAR) imaging [49].

3.1 Review on the parameter estimation of PPSs

Tremendous research work has been done on the estimation of PPSs. They can be categorized as parametric methods and non-parametric methods. Non-parametric methods either employ phase-locked loop (PLL), least mean square (LMS) and recursive least square (RLS) adaptive filters to track the phase of the signals [22, 42] or using time frequency transforms to estimated the instantaneous frequency of the signals [24]. These filter-based methods are generally efficient. However, these methods can not track rapid frequency variation. In addition, the step size has to be adjusted in advance. Time frequency based methods have the advantages of

providing time information about the variations of the instantaneous frequency of the analyzed signals. This category will be discussed in more details in Chapter 5. For the estimation of the parameters of PPSs, parametric methods are more favored than non parametric methods because parametric methods provide unlimited resolution [51]. Generally, the parametric methods fall into two categories which are sub-optimal maximum likelihood methods and maximum likelihood methods.

3.1.1 Sub-optimal maximum likelihood methods

Among the sub-optimal maximum likelihood methods, HAF-based estimation is the most popular candidate due to its computational efficiency. HAF was originally used for the parameter estimation of mono-component PPSs with constant amplitudes [83, 85]. Later, the HAF-based methods were extended to the case of multi-component PPSs [9, 11, 51, 84] and PPSs with random amplitudes [17]. Recently, a variety of methods based on HAF have been proposed to avoid the limitations existed in the previous HAF-based methods.

Firstly, it is found that the lags τ in (2.18) used in HAF are selected according to conflicting requirements because large lags are needed for a better estimation accuracy but severely limit the dynamic range of the estimated parameters [49, 124]. Two types of solutions have been proposed. One defined as the iterative approach [49] uses a small lag to achieve the rough estimation followed by a fine estimate with a large lag. The other approach [124] is to compute two HAFs with two large co-prime lags. The parameters of PPSs are recovered from the positions of the aliased peaks

by solving linear Diophantine equations. This approach was extended to multi-component PPSs in [119]. Secondly, spurious peaks arise when the components share the same highest order phase coefficients for multi-component PPSs [11], which is also defined as the identifiability problem. Product HAF (PHAF) is thus proposed as the multiplications between two or more HAFs with different lags. PHAF is based on the observation that the cross-terms of HAFs with different lags fall in different positions related with the lag parameters, while the auto-terms remain at the same position. In this way, the cross-terms are suppressed and the auto-terms remain. Thirdly, several algorithms are proposed to further improve the statistical performance including the output SNR and asymptotic variances. For example, integrated generalized ambiguity function (IGAF) was proposed in [9] based on the $M - 1$ order integrals on the HIM defined in (2.17). Two parameters instead of one as in the HAF-based method are estimated at each iteration. Because IGAF introduces the coherent integration along each lag in HIM, it provides higher output SNR, thus leads to a lower SNR threshold and a closer statistical performance to the CRLB. Another method [51] was proposed to use a 'bottom-up' approach which firstly estimates the parameters associated with the lowest order instead of the highest order because the number of the cross-terms increases with the order of HAF.

It should be mentioned that the improvement on the output SNR and estimation variances of these reported methods are limited, especially for PPSs with high order and/or low input SNR. This is because these methods do not change the nonlin-

earity of HAF, which is the fundamental reason for the performance deterioration. Furthermore, HAF-based methods suffer from the error propagation phenomena brought by its recursive estimation structure.

3.1.2 Maximum likelihood methods

Compared with those sub-optimal maximum likelihood (ML) methods, ML methods have higher output SNR without SNR threshold and achieve better statistical performance which asymptotically approaches the CRLB [17, 84]. This is because ML method is implemented with a linear transform called polynomial time frequency transform (PTFT) defined in (2.9). The PTFT has the advantages of no cross-terms or deterministic noise for multi-component PPSs due to its linear property. The 2nd order PTFT has been studied in the recent years. It has been found that when the signal length is prime, the ratio of the magnitudes between all the side lobes and the peak equal the square root of the signal length [118].

Although algorithms of fast Fourier transform (FFT) are available for use, the PTFT requires a huge computational complexity to deal with the multidimensional computation. Several fast algorithms for 2nd order PTFT were reported for mono-component linear chirp signals in the literature [1, 17, 75]. Fast quadratic phase transform [48] was proposed as an efficient algorithm for the 2nd-order PTFT to reduce the multiplicative complexity by a factor of $\log_2 N$ compared with that needed by directly using the FFTs. Unfortunately, there has not been a fast algorithm to support higher order of the PTFT in the literature. Recently, there is an increasing

need for high performance estimators for higher order PPSs. For example, longer observation intervals are required for the high resolution at low frequencies to recognize targets hidden from objects in natural environment. In this case, the lower order PPS such as the quadratic phase model is no longer valid [11, 101]. Thus, it is critical to develop fast algorithms for the higher order PTFT. The following Sections in this chapter present such a general fast algorithm for the PTFTs of any order by exploiting the intrinsic symmetric properties of the PTFT to remove the redundancy of the computation. For example, the proposed fast algorithm reduces the computational load by a factor of $2^M \log_2 N$ compared to the algorithm that directly uses the FFTs.

In Section 3.2, the advantages of PTFT are further introduced from the aspects of output SNRs and estimation variances with details. The decimation of FFT and the fast algorithm for the 2nd order PTFT are reviewed in Section 3.3. In Section 3.4, the fast algorithm is extended to the 3rd order PTFT and the analysis of its computational complexity is given. Next, the two important properties of PTFT are introduced for the generalized fast algorithm for PTFT with any order, together with the analysis of its computational complexity, in Section 3.5. Experimental results are given in Section 3.6.

3.2 Advantages of PTFT

It was mentioned in Chapter 2 that PTFT can be expressed as

$$\text{PTFT}_x^{M+1}(k, \mathbf{l}) = \text{DFT}_n[y_q^{(\mathbf{l})}(n)] \quad (3.1)$$

where $y_q^{(\mathbf{l})}(n)$ is a demodulated sequence defined as

$$y_q^{(\mathbf{l})}(n) = x(n) e^{-j2\pi\phi_2(\mathbf{l}, n)} \quad (3.2)$$

and

$$\phi_2(\mathbf{l}, n) = (l_1/N_1)n^2 + (l_2/N_2)n^3 + \cdots + (l_M/N_M)n^{M+1}$$

These equations are rewritten here for the convenience of the readers to have a better understanding on the remaining part of this Chapter.

Compared with those sub-optimal estimation methods, such as HAF, the most important advantage of PTFT is that it is a linear transform with respect to $x(n)$. The linear property is important because it makes the PTFT work well even with low SNR and/or high orders. The PTFT generally provides advantages in the following three aspects.

Firstly, it is proved in [17] that PTFT is the ML estimation of the parameters of PPSs and asymptotically achieves CRLB [86]. In contrast, the asymptotic variances of the HAF-based method derived in [90] shows a constant deviation from the CRLB and the deviation increases with the order of the PPS. Table 3.1 [90] shows the ratio of asymptotic variances between HAF-based estimation methods and PTFT-based estimation methods for a_m defined in (2.1). For example, the asymptotical

variances of HAF-based methods are about 50 percent larger than those of PTFT-based methods when $M = 3$ and even larger when $M = 4$. Furthermore, the ratio of the asymptotic variances between the HAF and the CRLB has a polynomial growth in the noise variance [90]. Therefore, the estimation based on the HAF is comparable to the estimation based on the PTFT only when SNR is high and the order of the PPS is small. The following experiment is conducted to compare the performance of these two methods. The 2nd order PTFT and HAF are applied for the estimation of a_1 and a_2 of the mono-component signal defined as:

$$s(n) = e^{j2\pi(0.001n^2+0.3n)} + w(n), \quad 0 \leq n \leq N-1 \quad (3.3)$$

where $N = 128$ and $w(n)$ is white Gaussian noise. The number of the grids

M+1	m=0	m=1	m=2	m=3	m=4	m=5	m=6
2	1.04	1.06	1.06				
3	1.33	1.49	1.53	1.66			
4	1.80	2.28	2.43	2.46	2.46		
5	3.10	4	4.08	4.08	4.08	4.08	
6	6.55	8.07	7.61	7.12	6.91	6.86	6.86

Table 3.1 The ratio of asymptotic variances between HAF-based methods and PTFT-based methods

for computing HAF and PTFT are selected large enough to keep the maximum quantization errors less than the root of the CRLB. The parameters of the PPS shown in (3.3) are estimated 100 times by HAF and PTFT at each SNR. The

(a) The variances of a_1 (b) The variances of a_2

Figure 3.1 Comparison of variances between PTFT and HAF of a PPS.

variances of the PTFT and HAF are calculated as the average of the square errors from each estimation process. The comparison with CRLB is shown in Figure 3.1. It can be seen easily that the variances acquired by the PTFT is always closer to CRLB than those by HAF. Especially when SNR is low, e.g. below 0 dB, the variances achieved by HAF-based method show a large deviation from CRLB, which means that HAF-based method cannot work under this condition.

Secondly, the output SNR (SNR_{out}), which is an important measurement commonly used in radar and communication systems [82, 93, 100], of the PTFT is higher than that of the HAF, especially when the input SNR (SNR_{in}) is low. The derivation of SNR_{out} for the PTFT is given in Appendix A. Equation (A.3) in Appendix A shows that the ratio between output SNR and input SNR of PTFT increases in proportion to the number of sample N and is not influenced by M . On the contrary, it was shown [85] that this ratio for HAF decreases with the increase of M .

Figure 3.2 Output SNR comparisons.

Figure 3.2 is drawn according to (A.3) and (4.12) in [85] and shows the comparisons between output SNRs of the PTFT and HAF with various orders when $N = 256$. It can be seen that output SNR of the HAF is generally lower than that of the PTFT. For example, the output SNR of PTFT is at least 10 dB higher than that of HAF when the input SNR is 30 dB. The difference is more significant with the increase of order and the decrease of SNR_{in} .

Thirdly, the PTFT does not suffer from error propagation in the recursive structure of the HAF-based estimation algorithms [90]. For multi-component PPSs, the PTFT has no cross-terms which generally exist and gives great difficulties in HAF [11, 51], which can be seen by comparing Figure 2.1 and 2.2 (b).

Although the PTFT provides statistical performance achieving the CRLB, the transform suffers from the huge computational load particularly for high order PPSs. For example, a direct computation of the PTFT generally requires $N^2 \prod_{i=1}^M N_i$ com-

plex multiplications. Although the computational costs can be reduced by using FFT algorithms as shown in (3.1), the PTFT still requires a computational complexity that is often difficult to support even with high speed processors. For example, about $\frac{1}{2}N(\log_2 N) \prod_{i=1}^M N_i$ complex multiplications are needed by a length- N $(M+1)$ th order PTFT.

3.3 DIT radix-2 FFT and the fast algorithm for the 2nd order PTFT

3.3.1 Decimation-in-time radix-2 FFT

For an easy understanding of the procedures used to derive the fast algorithm for the PTFT in the next two sections, let us have a brief review on the derivation of the FFT with a recursive computation structure. The FFT is derived by recursively decomposing the entire DFT computation into DFTs of smaller sequences. For the computation of (3.1), the sequence $y_q^{(1)}(n)$ is generally divided into two length- $(N/2)$ subsequences $y_q^{(1)}(2m)$ and $y_q^{(1)}(2m+1)$ according to even and odd indexing, respectively. Then, the computation of length- N DFT becomes the combinations of the length- $(N/2)$ DFT of these two subsequences. The same indexing procedure is repeatedly applied for the DFT computation of each subsequences. Let us assume the input sequence length is $N = 2^q$. In general, the decimation-in-time (DIT) radix-2 FFT has q computational stages and stage- r computes 2^r length- $(N/2^r)$ DFTs with r from $q-1$ to 0. For an easy presentation, the sequence $y_r^{(1)}(k)$ is

defined to be the output sequence of stage- r .

$$y_r^{(1)}(k) = \text{DFT}_m[y_q^{(1)}(2^r m + s)], \quad (3.4)$$

where $k = 2^r m + s$ with

$$m = 0, 1, \dots, \frac{N}{2^r} - 1,$$

$$s = 0, 1, \dots, 2^r - 1,$$

$$r = q - 1, \dots, 0.$$

The length- N FFT can be achieved by [20]

$$y_r^{(1)}(k) = y_{r+1}^{(1)}(k) + W^h y_{r+1}^{(1)}(k + 2^r), \quad r = q - 1, \dots, 0, \quad (3.5)$$

where $W = e^{-j\frac{2\pi}{N}}$ and h is an integer obtained by shifting and bit-reversing the binary expression of k [20]. It is decided by the following procedures [20].

- (i). Write the index k in binary form with q bits;
- (ii). Slide this binary number r bits to the right and fill zeros in the bit positions on the left;
- (iii). Reverse the order of the bits.

Following (3.5), the computation at stage- r for the length- $(N/2^r)$ DFT of $y_q^{(1)}(n)$ on the left hand side of (3.5) can be recursively calculated from the combination of two length- $(N/2^{r+1})$ DFTs from stage- $(r + 1)$ on the right hand side of (3.5), which are the even-indexed and odd-indexed subsequences in terms of m at stage r .

3.3.2 The fast algorithm for the 2nd order PTFT

The 2nd order PTFT achieved from (3.1) with $M = 1$ is

$$\begin{aligned} \text{PTFT}_x^2(k, l_1) &= \frac{1}{N_0} \sum_{n=0}^{N_0-1} x(n) e^{-j2\pi[(k/N_0)n + (l_1/N_1)n^2]} \\ &= \text{DFT}[y^{(l_1)}(n)], \end{aligned} \quad (3.6)$$

where $k = 0, 1, \dots, N-1$, $l_1 = 0, 1, \dots, N_1-1$, $y_q^{(l_1)}(n) = x(n)e^{-j2\pi(l_1/N_1)n^2}$ and $x(n)$ is the input sequence. It is assumed that the dimension sizes $N = 2^q$, where $q_0 \geq 1$, and $N_1 \geq N_0$ to achieve a satisfactory accuracy for the parameter estimation [48].

It is straightforward to see that the 2nd order PTFT can be computed through the FFT procedures shown in as (3.5)

$$y_r^{(l_1)}(k) = y_{r+1}^{(l_1)}(k) + W^h y_{r+1}^{(l_1)}(k + 2^r), \quad r = q-1, \dots, 0. \quad (3.7)$$

The important property used for the fast algorithm of the 2nd order PTFT is discovered in [48] as

$$y_q^{(l_1+N_1/2^{r+1})}(2^r m + s) = W_{r,s} y_q^{(l_1)}(2^r m + s), \quad (3.8)$$

where

$$W_{r,s} = e^{-j2\pi s^2/2^{r+1}}. \quad (3.9)$$

Since $W_{r,s}$ is independent of m , the following equation is achieved after applying DFT with respect to m to both sides of (3.8),

$$\text{DFT}[y_q^{(l_1+N_1/2^{r+1})}(2^r m + s)] = W_{r,s} \text{DFT}[y_q^{(l_1)}(2^r m + s)], \quad (3.10)$$

which is the same as the following equation by using (3.4),

$$y_r^{(l_1+N_1/2^{r+1})}(k) = W_{r,s} y_r^{(l_1)}(k). \quad (3.11)$$

Equation (3.11) suggests that $y_r^{(l_{11})}(k)$ with $l_{11} = l_1 + N_1/2^{r+1}$ can be obtained by that associated with $l_{11} = l_1$ at each computation stage of the 2nd order PTFT, which can be obtained by using FFT algorithm as shown in (3.5) without computing $y_r^{(l_1+N_1/2^{r+1})}(k)$ at stage higher than r . The detailed procedures of the fast algorithm for the 2nd order PTFT can be found in [48]. Unfortunately, this algorithm cannot be directly used for the higher order PTFT because the scaling factor $W_{r,s}$ is also a function of m besides r and s and the property shown in (3.11) is invalid. Therefore, more general form of the property in (3.11) is to be discovered for the fast algorithm of higher order PTFT. In the next two Sections, such properties are developed, together with the procedures of the fast algorithms and the analysis of computational complexity.

3.4 The fast algorithm for the 3rd order PTFT

The 3rd order PTFT achieved from (3.1) with $M = 2$ is

$$\text{PTFT}_x^3(k, l_1, l_2) = \frac{1}{N_0} \sum_{n=0}^{N_0-1} x(n) e^{-j2\pi[(k/N_0)n + (l_1/N_1)n^2 + (l_2/N_2)n^3]}, \quad (3.12)$$

where $k = 0, 1, \dots, N-1$, $l_i = 0, 1, \dots, N_i-1$ for $i = 1, 2$ and $x(n)$ is the input sequence. It is assumed that the dimension sizes $N = 2^q$, where $q_0 \geq 1$, and $N_2 \geq N_1 \geq N_0$ to achieve a satisfactory accuracy for the parameter estimation [48].

The PTFT of $x(n)$ is equivalent to the DFT of $y_{q_0}^{(l_1, l_2)}(n)$ expressed as

$$\text{PTFT}_x^3(k, l_1, l_2) = \text{DFT}_n[y_{q_0}^{(l_1, l_2)}(n)], \quad (3.13)$$

where

$$y_{q_0}^{(l_1, l_2)}(n) = x(n) e^{-j2\pi[(l_1/N_1)n^2 + (l_2/N_2)n^3]}. \quad (3.14)$$

The 3rd order PTFT requires a huge computational load. A direct computation of the 3rd order PTFT needs $N^2 N_1 N_2$ complex multiplications. One general method is to use FFT to reduce the number of multiplications to be in the order of $N_0 N_1 N_2 (\log_2 N_0)$. New algorithms must be found for further reduction of the computational complexity.

3.4.1 The fast algorithm

To further explore the similar property shown in (3.11) for the 3rd order PTFT, the computational process of PTFT using FFT algorithm is studied. The PTFT can be decomposed in a similar way to that used for the FFT which is reviewed in Section 3.3. The entire computation can be decomposed into q stages. For convenience of presentation, let us consider the computation at the r th-stage

$$y_r^{(l_1, l_2)}(k) = \text{DFT}_m[y_q^{(l_1, l_2)}(2^r m + s)] \quad (3.15)$$

$$r = q - 1, \dots, 0,$$

which defines 2^r length- $(N/2^r)$ DFTs of the input sequence $y_q^{(l_1, l_2)}(n)$ indexed by $2^r m + s$, where $m = 0, 1, \dots$, and $N/2^r - 1$; $s = 0, 1, \dots, 2^r - 1$. If a_1 and a_2 are

defined to be 0 or 1, a property of the PTFT can be described as

$$\begin{aligned}
y_r^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, l_2 + \frac{a_2 N_2}{2^{r+1}})}(k) &= \text{DFT}_m[y_q^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, l_2 + \frac{a_2 N_2}{2^{r+1}})}(2^r m + s)] \\
&= \text{DFT}_m[x(2^r m + s)e^{-j2\pi[\frac{k}{N_0}(2^r m + s) + \frac{2^{r+1}l_1 + a_1 N_1}{2^{r+1}N_1}(2^r m + s)^2 + \frac{2^{r+1}l_2 + a_2 N_2}{2^{r+1}N_2}(2^r m + s)^3}]] \\
&= e^{\frac{-j2\pi\phi(s)}{2^{r+1}}} \text{DFT}_m[x(2^r m + s)e^{-j2\pi[\frac{k}{N_0}(2^r m + s) + \frac{l_1}{N_1}(2^r m + s)^2 + \frac{l_2}{N_2}(2^r m + s)^3]}] \\
&\quad e^{-j\pi(2a_1 ms + 3a_2 ms^2 + 2^r a_1 m^2 + 2^{2r} a_2 m^3)}], \tag{3.16}
\end{aligned}$$

where $\phi(s) = a_1 s^2 + a_2 s^3$ is associated with a phase factor.

One special case of (3.16) when $r = 0$ is derived firstly as

$$\begin{aligned}
y_0^{(l_1 + \frac{a_1 N_1}{2}, l_2 + \frac{a_2 N_2}{2})}(k) &= \text{DFT}_m[y_q^{(l_1 + \frac{a_1 N_1}{2}, l_2 + \frac{a_2 N_2}{2})}(2^r m + s)] \\
&= \text{DFT}_m[x(m)e^{-j2\pi[\frac{k}{N_0}(m) + \frac{2^{r+1}l_1 + a_1 N_1}{2N_1}(m)^2 + \frac{2l_2 + a_2 N_2}{2N_2}(m)^3}]] \\
&= e^{\frac{-j2\pi\phi(s)}{2^{r+1}}} \text{DFT}_m[x(m)e^{-j2\pi[\frac{k}{N_0}(m) + \frac{l_1}{N_1}(m)^2 + \frac{l_2}{N_2}(m)^3]}] \\
&\quad e^{-j\pi(a_1 m^2 + a_2 m^3)}] \\
&= \begin{cases} y_0^{(l_1, l_2)}(k) & (a_1 + a_2) \text{ is even} \\ y_0^{(l_1, l_2)}(k + \frac{N}{2}) & (a_1 + a_2) \text{ is odd,} \end{cases} \tag{3.17}
\end{aligned}$$

which indicates that it is not necessary to compute the PTFT associated with $l_i > N_i/2$, $i = 1, 2$. In the following, the case when $r \neq 0$ is derived.

It can be easily seen that, when $(2a_1 s + 3a_2 s^2)$, which comes from the phase of the third term in $\text{DFT}_m[\cdot]$, is even, (3.16) can be simplified as

$$\begin{aligned}
y_r^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, l_2 + \frac{a_2 N_2}{2^{r+1}})}(k) &= e^{\frac{-j2\pi\phi(s)}{2^{r+1}}} \text{DFT}_m[y_q^{(l_1, l_2)}(2^r(2m) + s)] \\
&= e^{\frac{-j2\pi\phi(s)}{2^{r+1}}} y_r^{(l_1, l_2)}(k). \tag{3.18}
\end{aligned}$$

This is similar to the property utilized for the fast algorithm of 2-nd PTFT as shown

in (3.11). When $(2a_1s + 3a_2s^2)$ is odd,

$$\begin{aligned}
 & y_r^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, l_2 + \frac{a_2 N_2}{2^{r+1}})}(k) \\
 &= e^{\frac{-j2\pi\phi(s)}{2^{r+1}}} \{ \text{DFT}_m[y_q^{(l_1, l_2)}(2^r(2m) + s)] - W^h \text{DFT}_m[y_q^{(l_1, l_2)}(2^r(2m + 1) + s)] \} \\
 &= e^{\frac{-j2\pi\phi(s)}{2^{r+1}}} \{ y_{r+1}^{(l_1, l_2)}(k) - W^h y_{r+1}^{(l_1, l_2)}(k + 2^r) \}
 \end{aligned} \tag{3.19}$$

It is noted that both $y_r^{(l_1, l_2)}(k)$ and $y_r^{(l_1, l_2)}(k + 2^r)$ are length- $(N/2^{r+1})$ DFTs whose input sequences are, respectively, even- and odd-indexed in terms of m . In this way, the property shown in (3.11) is extended to the 3rd order PTFT because $y_r^{(l_1, l_2)}(k)$ is decomposed as even and odd-index sequences according to m during the decimation process of FFT, as shown in (3.5). The main functions used in the fast algorithm for the computation of 3rd order PTFT are

$$y_r^{(l_1, l_2)}(k) = y_{r+1}^{(1)}(k) + W^h y_{r+1}^{(1)}(k + 2^r) \tag{3.20}$$

$$y_r^{(l_1 + \frac{N_1}{2^{r+1}}, l_2)}(k) = \varsigma_{r,s} [y_{r+1}^{(1)}(k) + W^h y_{r+1}^{(1)}(k + 2^r)] \tag{3.21}$$

$$y_r^{(l_1, l_2 + \frac{N_2}{2^{r+1}})}(k) = \begin{cases} \zeta_{r,s} [y_{r+1}^{(1)}(k) + W^h y_{r+1}^{(1)}(k + 2^r)] & s \text{ is even} \\ \zeta_{r,s} [y_{r+1}^{(1)}(k) - W^h y_{r+1}^{(1)}(k + 2^r)] & s \text{ is odd} \end{cases} \tag{3.22}$$

$$y_r^{(l_1 + \frac{N_1}{2^{r+1}}, l_2 + \frac{N_2}{2^{r+1}})}(k) = \begin{cases} \xi_{r,s} [y_{r+1}^{(1)}(k) + W^h y_{r+1}^{(1)}(k + 2^r)] & s \text{ is even} \\ \xi_{r,s} [y_{r+1}^{(1)}(k) - W^h y_{r+1}^{(1)}(k + 2^r)] & s \text{ is odd} \end{cases} \tag{3.23}$$

where

$$\varsigma_{r,s} = e^{-j2\pi \frac{s^2}{2^{r+1}}}; \quad \zeta_{r,s} = e^{-j2\pi \frac{s^3}{2^{r+1}}}; \quad \xi_{r,s} = e^{-j2\pi \frac{s^2 + s^3}{2^{r+1}}}$$

and $l_i = 0, 1, \dots, N_i/2^{r+1}$.

In general, the computation can be decomposed into q stages with the above

decomposition method. At each stage r , the computation can be summarized as follows:

- Compute the length- $(N/2^r)$ DFTs for $l_i = 0, 1, \dots, N_i/2^{r+1} - 1$ and $i = 0, 1, 2$ according to (3.20) where the parameters $(a_1, a_2) = (0, 0)$;
- Compute the length- $(N/2^r)$ DFTs for $l_i = N_i/2^{r+1}, \dots, N_i/2^r - 1$ according to (3.21), (3.22) and (3.23) where the parameters $(a_1, a_2) = (1, 0)$, $(0, 1)$ and $(1, 1)$.

These procedures can be recursively applied for $r = q - 1, \dots, 0$.

Figure 3.3 shows the computational process when $N = 4$ and $N_1 = N_2 = 8$, which also has the similar butterfly structure as that in FFT algorithm. On the left hand side of the figure, $y_2^{(l_1, l_2)}(k)$ for l_1 or $l_2 = 0, 1$ are the four input sequences and each has four points. At the next stage, $y_1^{(l_1, l_2)}(k)$, for l_1 or $l_2 = 0, 1, 2, 3$ are computed according to (3.20) to (3.23). For example, $y_1^{(l_1, l_2)}(k)$ for $k = 0, 1, 2, 3$ are computed by (3.20). Similarly, $y_1^{(l_1+2, l_2)}(k)$ for $k = 0, 1, 2, 3$ are computed by (3.21). At the last stage, $y_0^{(l_1, l_2)}(k)$, where l_1 or $l_2 = 0, 1, 2, 3$ and $k = 0, 1, 2, 3$, are computed by (3.20). Up to now, all the 3rd order PTFT outputs defined on the right hand side of (3.17) are available. Then the property given in (3.17) is used to obtain all the other 3rd order PTFT points on the left hand side of (3.17).

Figure 3.3 Signal flow graph of the fast algorithm for the 3rd order PTFT.

3.4.2 Computational Complexity

Based on the decomposition procedures, let us consider the computational complexity needed by the proposed algorithm. The demodulated signal given in (3.14) requires only $N(N_1/N-1)(N_2/N-1)$ complex multiplications, which reduces about N^2 times from those required by directly using FFT. The multiplicative complexity for the fast algorithm is briefly analyzed as follows.

In (3.20), no complex multiplications are needed at stage $q-1$. Therefore, the

total number of complex multiplications from stage $r = q - 2$ to 0 is

$$N_{1,mul} = \sum_{r=0}^{q-2} \frac{N_1 N_2}{2^{2(r+1)}} \left(\frac{N}{2} \right) = (N N_1 N_2) \sum_{r=0}^{q-2} \frac{1}{2^{(2r+3)}} < \frac{1}{6} N N_1 N_2.$$

Complex multiplications are also needed for the factor $\varsigma_{r,s}$, $\zeta_{r,s}$ and $\xi_{r,s}$ in (3.21), (3.22) and (3.23) with $(a_1, a_2) = (1,0)$, $(0,1)$, $(1,1)$. Because multiplications are not needed when $r = 0, 1$, and $\frac{3N_1 N_2}{2^{2(r+1)}} (\frac{N}{2})$ complex multiplications are needed at the r th stage, the total multiplicative costs from stage $r = q - 1$ to 2 is

$$N_{2,mul} = \sum_{r=2}^{q-1} \frac{3N_1 N_2}{2^{2(r+1)}} \left(\frac{N}{2} \right) = \sum_{r=2}^{q-1} \frac{3N_1 N_2}{2^{2(r+1)}} \left(\frac{N}{2} \right) < \frac{1}{32} N N_1 N_2. \quad (3.24)$$

Finally, the total number of complex multiplications needed by the proposed algorithm is

$$\begin{aligned} N_{mul} &= N_{1,mul} + N_{2,mul} \\ &< \frac{1}{6} N N_1 N_2 + \frac{1}{32} N N_1 N_2 \\ &= \frac{19}{96} N N_1 N_2. \end{aligned} \quad (3.25)$$

It is noted that the above calculation includes the operations for trivial factors that generally do not need any multiplications. Compared with $\frac{N N_1 N_2}{2} \log_2 N$ complex multiplications needed by the directly use of the FFT, the proposed method achieves a reduction of the multiplicative complexity by approximately $2.5 \log_2 N$ times. Similarly, it can be calculated that the proposed algorithm needs about $N N_1 N_2 / 3$ complex additions while the method directly using the FFT needs $N N_1 N_2 \log_2 N_0$ complex additions. The proposed method achieves a reduction of the additive complexity by approximately $3 \log_2 N$ times.

3.5 The generalized fast algorithm for PTFT

This section generalizes the algorithm proposed in Sections 3.3 and 3.4 and presents a fast algorithm for PTFT with any order by using the quasi-periodic and key properties of the PTFT. It can be shown that many redundant operations can be removed from the FFT-based computation defined in (3.1).

3.5.1 Quasi-periodic property

The quasi-periodic property can be used to remove the inherent redundancy of the PTFT which is the extension of (3.17). Based on the definition in (2.9), this property can be expressed as

$$\begin{aligned} \text{PTFT}_x^{M+1} \left(k, l_1 + \frac{m_1 N_1}{2}, \dots, l_M + \frac{m_M N_M}{2} \right) \\ = \begin{cases} \text{PTFT}_x^{M+1}(k, \mathbf{l}) & M_0 \text{ is even} \\ \text{PTFT}_x^{M+1}(k + \frac{N}{2}, \mathbf{l}) & M_0 \text{ is odd.} \end{cases} \end{aligned} \quad (3.26)$$

where $M_0 = \sum_{i=1}^M m_i$ and $m_i = 0$ or 1 . The proof is given in Appendix B.

The quasi-periodic property indicates that the computation for those PTFT points with indices $l_i = \frac{N_i}{2}, \dots, N_i - 1$ is not necessary because they can be replaced by those with indices $l_i = 0, \dots, \frac{N_i}{2} - 1$. For the $(M + 1)$ th-order PTFT, this property means that the entire PTFT can be obtained from those PTFT points indexed with only the first half of the range for each l_i . Thus, the entire PTFT computation needs only $2^{-M} \prod_{i=1}^M N_i$, instead of $\prod_{i=1}^M N_i$, length- N FFTs needed in (3.1). With the quasi-periodic property, for example, the 3rd-order PTFT can be

expressed as:

$$\begin{aligned} \text{PTFT}_x^3(k, l_1 + \frac{N_1}{2}, l_2 + \frac{N_2}{2}) &= \text{PTFT}_x^3(k, l_1, l_2), \\ \text{PTFT}_x^3(k, l_1 + \frac{N_1}{2}, l_2) &= \text{PTFT}_x^3(k + \frac{N}{2}, l_1, l_2), \\ \text{PTFT}_x^3(k, l_1, l_2 + \frac{N_2}{2}) &= \text{PTFT}_x^3(k + \frac{N}{2}, l_1, l_2), \end{aligned} \quad (3.27)$$

where $l_i = 0, \dots, N_i/2 - 1$ and $i = 1, 2$. It can be verified that the total number of length- N FFTs required is only $N_1 N_2 / 4$.

3.5.2 Key property

Inspired by the concept of decimation-in-time FFT and the decimation processes along l_1 and l_2 shown in Section 3.4, decimation processes are also considered along each other dimension of **1**. For easy understanding, we take decimation along only one dimension l_u as a special example. From (3.4), the following equation can be easily acquired after simple mathematical derivation:

$$\begin{aligned} y_r^{(l_u + \frac{N_u}{2^{r+1}})}(k) &= \text{DFT}_m[y_q^{(l_u + \frac{N_u}{2^{r+1}})}(2^r m + s)] \\ &= \text{DFT}_m[x(2^r m + s)e^{-j2\pi(\frac{l_u}{N_u} + \frac{1}{2^{r+1}})(2^r m + s)^{u+1}}] \\ &= e^{-j2\pi\frac{s^{u+1}}{2^{r+1}}} \text{DFT}_m[x(2^r m + s)e^{-j2\pi(\frac{l_u}{N_u})(2^r m + s)^{u+1}}] \\ &\quad e^{-j\pi(u+1)ms^u}. \end{aligned} \quad (3.28)$$

Except the extra terms $e^{-j\pi(u+1)ms^u}$, which is either 1 or -1 depending on the parity of $(u+1)ms^u$, in the last line of (3.28), the rest of (3.28) is the same as $y_r^{(l_u)}(k)$.

When $(u+1)s^u$ is even, (3.28) can be further expressed as:

$$y_r^{(l_u + \frac{N_u}{2^{r+1}})}(k) = e^{-j2\pi\frac{s^{u+1}}{2^{r+1}}} y_r^{(l_u)}(k). \quad (3.29)$$

When $(u + 1)s^u$ is odd,

$$\begin{aligned}
 y_r^{(l_u + \frac{N_u}{2^{r+1}})}(k) &= e^{-j2\pi \frac{s^{u+1}}{2^{r+1}}} \{ \text{DFT}_m[x(2^r((2m) + s)e^{-j2\pi(\frac{l_u}{N_u})(2^r(2m)+s)^{u+1}}] \\
 &\quad - W^h \text{DFT}_m[x(2^r((2m + 1) + s)e^{-j2\pi(\frac{l_u}{N_u})(2^r(2m+1)+s)^{u+1}}] \\
 &= e^{-j2\pi \frac{s^{u+1}}{2^{r+1}}} [y_{r+1}^{(l_u)}(k) - W^h y_{r+1}^{(l_u)}(k)]. \tag{3.30}
 \end{aligned}$$

It can be seen that redundancy along l_u is removed in (3.29) and (3.30) because the middle result $y_r^{(l_u)}(k)$ with higher l_u can be obtained from that with lower l_u .

Next, the key property for all $M + 1$ dimensions is generalized from the above special case along l_u to obtain a significant reduction of computational complexity.

This property can be generally stated as

$$\begin{aligned}
 y_r^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, \dots, l_M + \frac{a_M N_M}{2^{r+1}})}(k) &= \begin{cases} \Phi(r, s)[y_{r+1}^{(1)}(k) + W^h y_{r+1}^{(1)}(k + 2^r)] & \gamma \text{ is even} \\ \Phi(r, s)[y_{r+1}^{(1)}(k) - W^h y_{r+1}^{(1)}(k + 2^r)] & \gamma \text{ is odd} \end{cases} \tag{3.31} \\
 r = q - 1, \dots, 1; \quad k = 0, \dots, N - 1,
 \end{aligned}$$

where k , m , and s have the same definitions as those in Section 3.3, $y_r^{(1)}(k)$ is the output at stage- r of the FFT computation, $a_i = 0$ or 1 for $i = 1, 2, \dots, M$, $\gamma = \sum_{i=1}^M (a_i(i + 1)s^i)$,

$$\Phi(r, s) = e^{-j2\pi \sum_{i=1}^M (\frac{a_i s^{i+1}}{2^{r+1}})},$$

and $l_i = 0, 1, \dots, N_i/N - 1$ for $r = q - 1$. Equation (3.31), proved in Appendix C, suggests that the computation involved with parameters $l_i + \frac{a_i N_i}{2^{r+1}}$ at stage- r on the left hand side of (3.31) can be obtained by combining the outputs, obtained at stage- $(r + 1)$, that are associated with index l_i on the right hand side of (3.31). Because only the outputs associated with index l_i at stage- $(r + 1)$ on the right hand

side of (3.31) are needed, the relationship in (3.31) is known as the key property at stage- r from which the computation $y_r^{(1)}(k)$ associated with larger l_i on the left hand side of (3.31) can be achieved. In this way, the redundancy along the $\mathbf{l}$ axes is removed.

3.5.3 The fast algorithm

Based on the quasi-periodic property in (3.26) and the key property in (3.31), the computation procedures of the proposed algorithm can be generalized into the following steps with $i = 1, 2, \dots, M$ and $k = 0, 1, \dots, N - 1$:

- (a). Compute $y_q^{(1)}(k)$ for $l_i = 0, 1, \dots, N_i/N - 1$ from the input sequence $x(n)$ according to (3.14).
- (b). For stage $r = q - 1$ to 1, compute $y_r^{(1)}(k)$ for $l_i = 0, 1, \dots, \frac{N_i}{2^{r+1}} - 1$ and $y_r^{(1)}(k)$ for $l_i = \frac{N_i}{2^{r+1}}, \dots, \frac{N_i}{2^r} - 1$ from $y_{r+1}^{(1)}(k)$ for $l_i = 0, 1, \dots, \frac{N_i}{2^{r+1}} - 1$ according to (3.5) and (3.31), respectively.
- (c). At stage-0, compute the length- N DFT of $y_r^{(1)}(k)$ according to (3.5) for $l_i = 0, 1, \dots, N_i/2 - 1$.
- (d). Compute the other PTFT points according to (3.26) for $l_i = N_i/2, \dots, N_i - 1$.

In comparison with the direct use of the FFT shown in (3.1), the proposed algorithm makes use of the quasi-periodic property to reduce the total number of PTFT points. The key property in (3.31) allows the number of initial input sequences to reduce N^M times (from original $\prod_{i=1}^M N_i$ to $N^{-M} \prod_{i=1}^M N_i$), and compute

$y_r^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, \dots, l_M + \frac{a_M N_M}{2^{r+1}})}(k)$ at stage- r from $y_{r+1}^{(1)}(k)$, rather than $y_{r+1}^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, \dots, l_M + \frac{a_M N_M}{2^{r+1}})}(k)$

at stage- $(r+1)$. Therefore, significant savings on computational complexity can be achieved, which is to be discussed in the next Section.

3.5.4 Computational complexity

Because the length- N input sequence is converted into an $(M+1)$ dimensional array according to (3.1), a direct use of the FFT algorithm needs to compute $\prod_{i=1}^M N_i$ length- N DFTs. The main contribution of the proposed algorithm is to use the quasi-periodic and the key properties to reduce the number of length- N DFTs and the redundancy along $\mathbf{l}$ during the computation of PTFT by FFT. In general, the computational complexity includes the total number of additions, multiplications and other possible operations. Let us first consider the multiplicative complexity in terms of the number of complex multiplications needed by the proposed algorithm. It is assumed that each length- N DFT need $N/2 \log_2 N$ complex multiplications because only $N/2$ complex multiplications are needed by the twiddle factor W^h at each computational stage. Based on the computation steps listed in the previous section, the multiplicative complexity can be analyzed as follows.

- Step (a) generates the demodulated signal $y_q^{(1)}(k)$ according to (3.14). The number of complex multiplications needed for $l_i = 0, 1, \dots, N_i/N - 1$ and $i = 1, \dots, M$ is

$$N_{1,mul} = N \prod_{i=1}^M \left(\frac{N_i}{N} - 1 \right) = \prod_{i=1}^M (N_i - N).$$

It is noted that when $l_i = 0$, no complex multiplication is needed.

- Step (b) performs the computation stages based on (3.5) and (3.31) for $r = q-1$ to 1. The first part is the complex multiplications that are needed by the twiddle factor W^h in (3.5) to obtain $y_r^{(1)}(k)$, where $l_i = 0, \dots, \frac{N_i}{2^{r+1}} - 1$ and $i = 1, \dots, M$. It should be noted that no complex multiplications are needed at stage $q-1$. Therefore, the required number of complex multiplications is

$$N_{21,mul} = \frac{N}{2} \sum_{r=1}^{q-2} \prod_{i=1}^M \left(\frac{N_i}{2^{r+1}} \right) = \left(N \prod_{i=1}^M N_i \right) \sum_{r=1}^{q-2} \frac{1}{2^{(r+1)M+1}},$$

where $\frac{N}{2}$ is the number of complex multiplications in (3.5) for half range of k because of the twiddle factor W^h at each computational stage. Complex multiplications are also needed for multiplying factor $\Phi(r, s)$ in (3.31) to calculate $y_r^{(1)}(k)$, where $l_i = \frac{N_i}{2^{r+1}}, \dots, \frac{N_i}{2^r} - 1$ and $i = 1, \dots, M$. Considering that trivial complex multiplications are needed at stage 1, the number of complex multiplications in (3.31) becomes

$$N_{22,mul} = \frac{N}{2} \sum_{r=2}^{q-1} \left[\prod_{i=1}^M \frac{N_i}{2^r} - \prod_{i=1}^M \frac{N_i}{2^{r+1}} \right] = \left(N \prod_{i=1}^M N_i \right) \sum_{r=2}^{q-1} \frac{2^M - 1}{2^{(r+1)M+1}},$$

where $\frac{N}{2}$ can be considered to be the number of complex multiplications needed by $\Phi(r, s)$ in (3.31). Therefore, the computational cost for Step (b) from stage

$r = q - 1$ to 1 is

$$\begin{aligned}
 N_{2,mul} &= N_{21,mul} + N_{22,mul} \\
 &= \left(N \prod_{i=1}^M N_i \right) \sum_{r=1}^{q-2} \frac{1}{2^{(r+1)M+1}} + \left(N \prod_{i=1}^M N_i \right) \sum_{r=2}^{q-1} \frac{2^M - 1}{2^{(r+1)M+1}} \\
 &\simeq \left(N \prod_{i=1}^M N_i \right) \left[\left(\frac{1 - 2^{-M(q-3)}}{2^{2M+1} - 2^{M+1}} \right) + \frac{1}{2^{2M+1}} \right] \\
 &< \frac{N \prod_{i=1}^M N_i (2^{M+1} - 1)}{2^{3M+1} - 2^{2M+1}}. \tag{3.32}
 \end{aligned}$$

- Step (c) is to compute $y_0^{(l)}(n)$ with $l_i = 0, \dots, \frac{N_i}{2} - 1$ according to (3.5). The required number of complex multiplications is

$$N_{3,mul} = \frac{N}{2} \prod_{i=1}^M \left(\frac{N_i}{2} \right) = \frac{N \prod_{i=1}^M N_i}{2^{M+1}}$$

Finally, the total number of complex multiplications needed by the proposed algorithm is

$$\begin{aligned}
 N_{mul} &= N_{1,mul} + N_{2,mul} + N_{3,mul} \\
 &< \left\{ \prod_{i=1}^M (N_i - N) \right\} + \frac{N \prod_{i=1}^M N_i (2^{M+1} - 1)}{2^{3M+1} - 2^{2M+1}} + \frac{N \prod_{i=1}^M N_i}{2^{M+1}} \\
 &= \frac{N \prod_{i=1}^M N_i (2^{2M} + 2^{M+1} - 2^M - 1)}{2^{2M+1} (2^M - 1)} + \prod_{i=1}^M (N_i - N). \tag{3.33}
 \end{aligned}$$

The proposed fast algorithm saves the multiplicative complexity by about $2^M \log_2 N$ times, where $M \gg 2$, compared with the multiplicative complexity $\frac{N \log_2 N}{2} \prod_{i=1}^M N_i$ that is required by directly using FFTs. It should be pointed out that the multiplicative complexity given in (3.33) can be considered to be an up bound because the saving from many trivial twiddle factors, which do not need any multiplications, is not considered in (3.33).

The total number of complex additions needed by the proposed algorithm can also be similarly derived to be

$$N_{add} < \frac{N}{(2^M - 1)} \prod_{i=1}^M N_i \quad (3.34)$$

Compared with the additive complexity of $N \log_2 N \prod_{i=1}^M N_i$ that is required by directly using FFTs, the reduction of the additive complexity achieved by the proposed algorithm is about $2^M \log_2 N$ times.

Since the proposed algorithm is derived by using the radix-2 decomposition, it has a regular computational structure that therefore leads to a simple implementation.

3.6 The experiment

To verify the gain achieved by the proposed fast algorithm, a simple experiment is carried out to compute the ratio between the computation times needed by directly using FFT in (3.1) and the proposed fast algorithms for the second and third order of PTFTs, as shown in Figure 3.4. The savings achieved by the proposed fast algorithm is more significant when the order of PTFT or N increases because more redundancy along many dimensions of $\mathbf{l}$ is explored according to the two properties described in Section 3.5.

3.7 Conclusion

A fast algorithm is presented to compute the PTFT for parameter estimation of PPSs. The analysis of multiplicative complexity is also presented. With the use

Figure 3.4 Comparison of computation time.

of the quasi-periodic and the key properties, the proposed fast algorithm achieves significant savings on the computational complexity compared with that directly using FFTs. For example, the proposed algorithm generally achieves a significant reduction of the computational complexity by $2^M \log_2 N$ times. It is worth mentioning that although the computational complexity reduction is achieved by the proposed fast algorithm, the PTFT is still more computationally intensive than the HAF. Since the PTFT provides much better statistical performance than the HAF when the SNR is low and/or the order of PPS is high, it is expected that these two kinds of methods are combined to achieve the best performance in terms of both computational complexity and statistical performance.

Chapter 4

Modified DPFT and its application for the interference excision in DS-SS system

4.1 Introduction

4.1.1 DS-SS communication system

The spread spectrum (SS) technique has been widely used in a variety of military and civilian applications. These applications include code division multiple access (CDMA), interference mitigation, combating with multi-path fading, covert communications and ranging. The transmitted signal should satisfy two criteria in an SS communication system as follows

- The bandwidth of the transmitted signal must be much greater than the mes-

sage bandwidth.

- The transmitted bandwidth must be spread by the function that is independent of the message and known to the receiver.

There are several different types of SS systems and the most commonly used one is the direct sequence (DS) SS system. Figure 4.1 shows the block diagram of the DS-SS system. In DS-SS system, a pseudorandom (PN) sequence is superimposed on the data bits at the transmitter before the modulation process sends the desired signals into the channel. In the channel, the desired signal is disturbed by the intentional/unintentional interference and white Gaussian noise. At the receiver, the correlation with the same PN sequence recovers the original signal.

Figure 4.1 Block diagram of DS-SS system.

4.1.2 Problem description and previous work

One of the most useful advantages of a DS-SS system is interference mitigation (particularly for narrowband interference). The spectrum of the data signal is spread by a particular PN sequence at the transmitter. At the receiver, the received signal

including the interference sequence is re-acquired by multiplying the same PN sequence. The narrowband interference is spread by the PN sequence, but the signal is despread into its original bandwidth. Therefore, the power of the interference is reduced significantly.

However, DS-SS system deteriorates significantly if the energy of the narrowband interference is large or the interference becomes broadband. A general class of interference known as polynomial phase interference (PPI) has attracted increasing interests in recent years [3, 4, 6, 10, 50, 64, 111, 121, 122]. The PPI includes not only the narrowband tone interference, but also a wide range of broadband interferences, e.g. linear chirp interference.

It has been demonstrated that the use of interference excision not only improves the error rate performance, but also leads to increased system capacity and improves the acquisition capability for a commercial CDMA system [89]. Therefore, mitigation of this kind of interference becomes extremely critical and it is necessary to employ advanced signal processing techniques to excise the interference effectively. In recent years, the excision of PPI has attracted much attention and many useful techniques [6, 10, 65, 69, 72, 89, 111, 121, 122] have been proposed. Generally, they fall into three categories: the time varying notch filtering approach, interference synthesis approach and transform domain based excision approach.

A. Time-varying notch filtering approach

Figure 4.2 shows the block diagram of the methods in this category. In this category, an adaptive time-varying notch filter is employed with its filter tap coefficients

updated according to a suitable adaptive algorithm [72, 89], such as the linear mean square (LMS) filter, or the instantaneous frequency (IF) of the interference estimated by time frequency transforms (TFTs). In [89], specifically, the predictability of the

Figure 4.2 Block diagram of IF estimation approach.

narrow-band interference was exploited by using the LMS filter and then a replica of the narrow-band interference is subtracted from the received signal. In [111], the IF of the interference is estimated by using short time Fourier transforms (STFTs). Then the received data are processed by an adaptive finite impulse response (FIR) filter with a notch at the estimated IF. Although the interference can be effectively removed, this method brings correlation to the PN sequence causing significant distortion to the signals, especially when the interference to signal energy ratio (ISR) is small [111]. It is also not possible to use this method for excision of multi-component interference [111]. In [10], the so-called reassigned smoothed pseudo WVD of the received signal is computed, followed by computing the Wigner Hough transform (WHT) to estimate the IF of the interference. Then the received signal is multiplied with the signal that has the conjugate estimated IF, which makes the interference occupy only low frequency band. The interference is then suppressed by a high-pass filter. This method requires a high computational complexity due to the use of reassigned smoothed pseudo WVD and two dimensional search required by the WHT, which makes it difficult to be implemented on line.

Generally, the interference of the methods in this category often requires longer time for the estimation of the interference than that of the interference excision and they cannot track rapidly changing interference [122].

B. Interference synthesis approach

In this category, the interference is first estimated from the corrupted signal. The synthesized interference is then subtracted from the received signal, as shown in Figure 4.3, to suppress the interference.

An excision approach was proposed based on Wigner Ville distribution (WVD) synthesis [64]. The instantaneous frequency (IF) and amplitude of the interference are estimated based on the peak position of WVD and projection method, respectively. The disadvantage of this approach is the high computational complexity due to the use of WVD synthesis. Additionally, WVD inherently yields significant undesirable cross-terms. The performance of the interference mitigation is significantly affected by the existence of even a small portion of these cross-terms. Thus, it is difficult to use this approach if the components of the interference are crossed in the time frequency domain due to the difficulty in distinguishing the interference and the cross-terms. Another interference synthesis approach proposed in [50] is based on parametric estimation, which is known as HAF. In this method, the parameters of the interference are estimated by iterative HAF. The disadvantage of this approach is that its performance in terms of bit error rate (BER) is sensitive to the time the interference begins because the HAF can not estimate this time instant. Additionally, least square estimation is required to estimate the amplitude and the

phase, which increases the computational load of the excision process.

Figure 4.3 Block diagram of interference synthesis approach.

C. Transform domain based method

In transform domain based interference excision, as shown in Figure 4.4, the windowed signal is transformed into a domain where the interference is confined into a few components and the desired signal is spread. The excision process is realized by setting these interference components to be zero in the same domain before the interference-free signal is transformed back into the time domain. Generally, transform domain based method can be easily implemented in hardware and track rapidly changing interference [65, 72, 89]. These advantages are particularly important for applications where the interference is required to be excised quickly. For instance, such a scenario arises for a hybrid DS and frequency hopping SS system where interference is required to be excised after frequency hopping demodulation, where the frequency hopping period can be of only tens or hundreds of micro seconds.

Figure 4.4 Block diagram of transform based approach.

Much research work [26, 37, 60, 61, 65, 72, 89, 94, 95, 121, 122] has been done on the transform domain method for narrow band tone interference excision. For tone interference, *Fourier transform* (FT)-based frequency excision method was studied extensively [26, 29, 37, 60, 61, 72, 94, 95, 121, 122] because of its simple implementation in hardware and the easiness of tracking rapidly changing interference. The FT implemented by surface acoustic wave (SAW) devices was originally applied to excise the narrow band interference [72, 89]. Discrete FT (DFT) based excision technique was developed to provide higher dynamic range than SAW devices [26, 29, 37, 60, 61, 94, 95, 121, 122]. Other transforms, such as the spectrally-contained orthogonal transform [98] and lapped transforms [69], are also proposed for the excision of tone interference.

Because of the applications of time frequency transforms (TFTs), transform-domain based excision method is possible to be extended for broadband PPI suppression. Several TFTs including adaptive wavelet transform [109], adaptive STFT [76] and WVD [64] have been used for the excision of the 2nd order PPI. All these transforms, except WVD, do not have optimal concentration property for the 2nd order PPI, i.e., linear chirp interference. Alternately, the undesirable influence of cross-terms prevents WVD from being used for the multi-component interference excision. In [3], Akay *et. al* proposed an excision method for linear chirp interference excision based on fractional Fourier transform (FRFT).

In practice, a rectangular window function has unfortunately to be used to localize the input signal in all transform-based excision methods before computing

the transform. This windowing operation causes frequency dispersion and leads to undesirable side lobes. It is extremely difficult to remove the interference in the side lobes of the transformed signal because of the difficulties in distinguishing the interference from the desired signal. Previously, the problems of side lobes for the narrowband tone interference were alleviated by time-weighting DFT based method [26, 29, 37, 60, 61, 94, 95, 97, 121, 122]. In this method, different time weightings are applied by using non-rectangular windows to minimize the magnitudes of the side lobes. At the same time, various frequency mapping algorithms, including setting some of the components to zeros and fraction clipping [29, 121, 122], were also developed as the complement of the time-weighting DFT-based methods for the interference excision. The performance of the time-weighting DFT-based methods has been analyzed in detail in [26, 37, 60, 61, 94, 95, 121, 122]. The performance metrics corresponding to different combinations of windows and frequency mapping algorithms were developed [121, 122]. It was found that the optimal use of windows for the frequency mapping algorithms was inevitably influenced by the frequency characteristics of the interference [121, 122]. Unfortunately, there are several disadvantages for the time-weighting methods. Firstly, it is well known that using non-rectangular window to yield lower side lobes results in a broader main lobe. Thus, more components in the main lobe are to be excised, which brings more distortion to the desired signals. Secondly, a demodulation rule adaptive to the time-weighting and excising processing is required to be employed because the operation of time-weighting colors the white noise [97]. This adaptive demodulation

increases both the computational complexity and the level of the implementation difficulty [98].

Because of the existence of side lobes, the most significant disadvantage for transform domain based methods, including those with rectangular and those with non-rectangular weighting, is that their performances are sensitive to the initial frequency of PPI [65,69,121,122] and vary significantly with the parameters of PPI. The main reason for this problem is that the transforms are evaluated with a grid of discrete frequencies and the biases, which are functions of the initial frequency of the interference, are introduced by this grid. The interference is mostly concentrated in the transform-domain when the bias is zero and becomes severely dispersive with the increasing of the bias.

In the following section, a new transform domain excision method based on discrete polynomial Fourier transform (DPFT) is developed together with the detailed discussion on the influence of undesirable side lobes and the bias introduced by the grid. We show that the problem caused by side lobes can be minimized by using the proposed modified DPFT (MDPFT). Then, two examples concerning narrowband and broadband interference excision are given with detailed analysis of performances in Section 4.3 and 4.4, respectively.

4.2 The modified DPFT-based excision method

Let us assume that binary phase-shift-keying (BPSK) signals are used in the DS-SS systems and normalized by the energy of the transmitted signal. The received BPSK

signals in one transmitted symbol duration are sampled at the rate of once per chip and have the form of

$$r(n) = p(n)s + A g(n) + w(n), \quad 0 \leq n < N, \quad (4.1)$$

where A is the amplitude of the interference with $A^2 = \text{ISR}$, $p(n)$ is the PN sequence of length L , s , the transmitted symbol, is either 1 or -1, $w(n)$ is additive white Gaussian noise with zero mean and variance $\sigma_n^2 = 1/\text{SNR}$, SNR is the signal to white Gaussian noise energy ratio and ISR is the interference to white Gaussian noise energy ratio. ISNR defined as $A^2/(1 + \sigma_n^2)$ is the interference to noise plus signal energy ratio and $g(n)$ is the PPI with the form of

$$g(n) = e^{j2\pi(\sum_{m=1}^{M+1} l_m n^{m+1} + k_0 n + p)}, \quad (4.2)$$

where p is the phase of the PPI, and k_0 is defined as the initial frequency of the PPI. For example, the PPI becomes a narrowband tone interference when $\mathbf{l} = \{l_1, \dots, l_{M+1}\}$ are all zeros. The PPI becomes broadband linear chirp interference when only l_1 is nonzero. Since it was mentioned in Section 2.4.1 that DPFT has the optimal concentration for the PPI, we can use DPFT as the transform to excise the interference effectively.

4.2.1 DPFT-based method and the side lobe distortion

To use DPFT for the excision of PPI, the parameter vector $\mathbf{l}$ of the PPI is firstly estimated as $\widehat{\mathbf{l}} = \{\widehat{l}_1, \dots, \widehat{l}_{M+1}\}$ by one of the estimation methods introduced in Chapter 3. Because the energy of the interference is relatively large compared with

that of the noise $w(n)$ when the interference excision is required and on-line operation is needed for this application, HAF-based method is used in this Chapter to achieve a good computational efficiency without much deteriorating the estimation performance. However, it should be noted if the model for the PPI is of high order, the PTFT-based method is more desired. This is due to the fact, shown in Section 3.2, that the performance of HAF-based method deteriorates with the increase of the order of PPI. This scenario may arise when higher order polynomial phase signal modelling is required because longer processing period is needed to obtain high resolution. Since the estimation variances of the PTFT-based method and HAF-based method for $\mathbf{l}$ is close to CRLB when the SNR (ISNR in this case) is high [90], we do not consider the difference between the estimated $\hat{\mathbf{l}}$ and $\mathbf{l}$ and use $\mathbf{l}$ instead in the remaining part of this Section. It should be noted that the influence of the estimation variances is added for the analysis of the theoretical performance in Section 4.4.

After estimating parameters $\mathbf{l}$, the following procedures of DPFT-based excision method can be seen from Figure 4.4, in which T is replaced by DPFT. The DPFT of the windowed $r(n)$ is used to localize the interference $Ag(n)$ in the DPFT domain. Since the DPFT is evaluated with the grid of discrete frequencies, the interference becomes the sampled version of the FT of the window function shifted by a distance decided by the initial frequency of the interference. To effectively remove the interference with a minimum distortion of the desired signal, the interference in the DPFT domain is required to be concentrated within as few components as possible.

The DPFT of the windowed interference $Ag(n)$ generally has a main lobe and a few side lobes. When mixed with the desired signal which is spread in the whole DPFT domain, the interference components in the main lobe are easily distinguished from the desired signal because they have much larger magnitudes than those of the desired signal. The interference energy from the main lobe can be easily suppressed by using spectrum mapping algorithms to either clip or set those interference components to zeros. In the side lobes, however, it is difficult to distinguish the interference components from those of the desired signal.

The removal of interfering frequency components in the side lobes inevitably results in distortion to the desired signal and performance deterioration. Although this problem has been traditionally alleviated by choosing windows having lower side lobes, the bias introduced by grid influences substantially the concentration of the interference in the DPFT domain. Let us consider the DPFT of the PPI. It can be easily seen from the definition of DPFT, shown in (2.23) of Chapter 2, that if the product of the initial frequency of the interference k_0 and the processing block length N is an integer, the DPFT of the PPI becomes mostly concentrated. When a rectangular window is used, for example, the DPFT of the interference is a sampled version of SINC function and becomes a single line in the main lobe and zeros in all side lobes if the bias $\delta = k_0N - \lfloor k_0N \rfloor = 0$ as shown in (2.24) and Figure 4.5 (b), where $\lfloor x \rfloor$ is the operation to round x to the nearest integer. It can be easily seen from (2.23) that the DPFT of the interference becomes more dispersive as the modulus of δ in Figure 4.5 (a) increases and has non-zero frequency components

in the side lobes that leads to distortion of the desired signal when suppressing them. Therefore, the performance of DPFT-based method is sensitive to the initial frequency of the interference. It is a critical issue to find an effective method to obtain the DPFT of the interference with any initial frequency as concentrated as possible.

Figure 4.5 The windowed interference in the DPFT domain.

4.2.2 Modified DPFT-based excision method

A proper type of windows is needed for the proposed interference excision method. Rectangular window is selected for two reasons. The first one is that adaptive demodulation techniques instead of conventional ML receiver are required if non-rectangular windows are used [26, 94, 97]. The use of adaptive demodulation complicates the receiver hardware and increases the total computational cost [98]. The second reason, which is more important, is that the interference with rectangular

window can be mostly concentrated compared with that using other windows. When $\delta = 0$, for example, the interference becomes a single line in the main lobe, as shown in Figure 4.5 (b). Such a property allows the interference to be removed with a minimum distortion of the desired signal, i.e., only one component is changed in the DPFT domain. Thus, the problem to be solved is to find a new transform to convert the PPI into a single component for the interferences with any k_0 . The objective can be achieved by shifting the DPFT in Figure 4.5 (a) by a distance of δ to obtain the DPFT in Figure 4.5 (b). The modified DPFT is particularly designed to utilize the parameter δ , which can be estimated, so that the desired DPFT of the interference can be always achieved. The $(M + 1)$ th order MDPFT is defined as

$$X_1(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} r(n) e^{-j2\pi(\sum_{i=1}^M l_i n^{i+1} + \frac{kn+\delta n}{N})},$$

$$0 \leq k \leq N - 1, \quad -0.5 < \delta \leq 0.5, \quad (4.3)$$

and the inverse $(M + 1)$ th order MDPFT is

$$r(n) = \frac{p(\delta)}{\sqrt{N}} \sum_{k=0}^{N-1} X_1(k) e^{\frac{j2\pi kn}{N}}, \quad (4.4)$$

where

$$p(\delta) = e^{j2\pi(\sum_{i=1}^M l_i n^{i+1} + \frac{\delta n}{N})}, \quad 0 \leq n \leq N - 1. \quad (4.5)$$

The parameter δ is the distance to shift the DPFT to meet the condition that $[k_0 N - \delta]$ is an integer. The MDPFT of the interference in the shifted DPFT of the interfered signal has zero values in the side lobes and the maximum component at the middle of the main lobe, as shown in Figure 4.5 (b).

Let us now consider the estimation of δ from the DPFTs for computing MDPFT. From the definition of DPFT in (2.23), it can be seen that after de-chirping operation (i.e. multiplying $x(n)$ with $e^{-j2\pi(\sum_{i=1}^M l_i n^{i+1})}$), the remaining operation is the DFT computation. Therefore, the value of δ can be estimated by the interpolated DFT method [66] which has many advantages, including low computational complexity, immune to nonzero mean signals and near optimal in colored or non-Gaussian noise [66]. This algorithm uses the peak value in the main lobe of the DPFT and its two neighboring values because most of the interfering energy is contained within them. If it is assumed that the peak value is $X_1(k_1)$ and $R(m) = \text{Real}[e^{-jm\pi/N} X_1(k_1 + m)X_1^*(k_1)]$, then [66]

$$\begin{aligned}\delta &\approx \delta_m - 0.2576 \sin(2\pi\delta_m)/N^2, \\ \delta_m &\approx \frac{\sqrt{1 + 8r^2} - 1}{4r},\end{aligned}\tag{4.6}$$

where

$$r = \frac{R(-1) - R(1)}{2R(0) + R(-1) + R(1)}.$$

It was shown that the worst-case biases in δ estimation is $\pm 0.002/N^2$ bins and the bias may be ignored if they are much smaller than the errors due to noise [66]. This helps to analyze the BER performance for the proposed excision algorithm in the next two Sections. For the signal that contains a mono-component $(M + 1)$ th order PPI, the steps of MDPFT based excision process can be summarized as follows:

- (i). Estimate **1** by using HAF-based method defined by (2.20);
- (ii). Compute the DPFT defined by (2.23) of the interfered signal;

- (iii). Estimate δ defined in (4.6) by the interpolated DFT method;
- (iv). Use the estimated δ to compute the MDPFT of the interfered signal defined in (4.3);
- (v). Remove the frequency component with the highest magnitude and its $M_0 - 1$ neighboring components on both sides in the MDPFT domain; (The details about the optimal M_0 are provided in Section 4.3.)
- (vi). Compute the IMPDFT, according to (4.4), of the output obtained from step (v) to recover the desired signal.

Steps (iii) and (v) need only several simple mathematical operations for computing (4.6) and comparisons to find the peak of $|MDPFT|$, which uses a small portion of the total computational complexity. The main computational cost is from step (i), (ii), (iv) and (vi). In step (i), there are M FFT computations. In each step of (ii), (iv) and (vi), there is one FFT computation. Thus, the total computational complexity is about $(M + 3)/2 \log_2 N$, which is in the order of $N \log_2 N$. Therefore, the proposed MDPFT-based method is computationally efficient and can easily be applied on-line for practical applications. Detailed computational comparisons of the computational complexity are given for narrowband and broadband examples in Sections 4.3 and 4.4, respectively.

4.2.3 Performance analysis

It is assumed that $\{p(n)\}$, $\{s\}$ and $\{w(n)\}$ are uncorrelated and $E[p(i)p^*(j)] = \delta(i - j)$, where $E[\cdot]$ represents the expectation operator and $\delta(i - j)$ represents

the Kronecker delta function. It is also assumed that $N = L$ in our derivation for simplicity of presentation. It is straightforward to derive the BER expression with $N \neq L$ by using the matrix partition technique, which is not presented here. If the transmitted bit $s = 1$ and M_0 is the number of the components excised by setting them to zeros, the output after step (vi) shown in Section 4.2.2 is

$$\mathbf{y} = \mathbf{B}(\mathbf{p} + \sqrt{ISR} \mathbf{g} + \mathbf{w}), \quad (4.7)$$

where $\mathbf{p} = [p(0), \dots, p(L-1)]^T$, $\mathbf{g} = [g(0), \dots, g(L-1)]^T$, $\mathbf{w} = [w(0), \dots, w(L-1)]^T$, and matrix $\mathbf{B} = \mathbf{D}^H \mathbf{A} \mathbf{D}$, where $\mathbf{A}$ represents the binary masks in the MDPFT domain with diagonal elements containing M_0 zeros, $\mathbf{D}$ and $\mathbf{D}^H$ are the coefficient matrices of the MDFT and IMPDFT, respectively. It can be easily achieved from (4.3) and (4.4) that the element of $\mathbf{D}$ is $D(h, k) = e^{-j2\pi(\sum_i^M l_i k^{i+1} + \frac{hk+\delta k}{N})}$. The superscripts T and H in the above expressions represent the transpose and complex conjugate operators. It can be easily derived that

$$\mathbf{B} = \mathbf{D}^H \mathbf{A} \mathbf{D} = \mathbf{D}^H (\mathbf{I} - \overline{\mathbf{A}}) \mathbf{D} = \mathbf{I} - \mathbf{D}^H \overline{\mathbf{A}} \mathbf{D} = \mathbf{I} - \overline{\mathbf{B}}, \quad (4.8)$$

where $\mathbf{I}$ is the unit matrix, $\overline{\mathbf{A}} = \mathbf{I} - \mathbf{A}$ and $\overline{\mathbf{B}} = \mathbf{I} - \mathbf{B}$. Thus, the element of $\overline{\mathbf{B}}$ can be written as

$$\overline{B}(i, h) = \sum_{l=0}^{L-1} \left\{ D^*(l, i) \sum_{k=0}^{L-1} [\overline{A}(l, k) D(k, h)] \right\}. \quad (4.9)$$

If it is assumed that there are M_0 diagonal elements of $\mathbf{A}$ being zeros with index $i_1, \dots, i_{M_0}$, the element of $\overline{\mathbf{A}}$ is

$$\overline{A}(i, h) = \begin{cases} 1, & i = h = i_1 \cdots i_{M_0} \\ 0, & \text{else} \end{cases} \quad (4.10)$$

Using (4.10), $B(i, h)$ can be derived as

$$B(i, h) = \begin{cases} -\frac{1}{L} \sum_{l=[i_1 \dots i_{M_0}]} D^*(l, i) D(l, h), & i \neq h \\ 1 - \frac{M_0}{L}, & i = h \end{cases} \quad (4.11)$$

where $L = N$ is from the scaling factor in (4.3) and (4.4). The decision variable is written as:

$$y_0 = \mathbf{p}^H \mathbf{y} = \mathbf{p}^H \mathbf{B} \mathbf{p} + A_0 \mathbf{p}^H \mathbf{B} \mathbf{g} + \mathbf{p}^H \mathbf{B} \mathbf{w}. \quad (4.12)$$

The mean value of the decision variable is

$$\begin{aligned} E[y_0] &= E[\mathbf{p}^H \mathbf{B} \mathbf{p}] \\ &= E \left[\sum_{i=0}^{L-1} \sum_{h=0}^{L-1} p^*(i) B(i, h) p(h) \right] \\ &= \sum_{h=0}^{L-1} B(h, h) \end{aligned} \quad (4.13)$$

where $p(i)$ and $B(i, h)$ are the elements of $\mathbf{p}$ and $\mathbf{B}$, respectively. Then, (4.13) can be further derived by using (4.11) as

$$E[y_0] = \sum_{h=0}^{L-1} \left(1 - \frac{M_0}{L}\right) = L - M_0. \quad (4.14)$$

Similarly, the variance of the decision variable is expressed as:

$$Var[y_0] = Var[\mathbf{p}^H \mathbf{B} \mathbf{p}] + (A_0^2) Var[\mathbf{p}^H \mathbf{B} \mathbf{g}] + Var[\mathbf{p}^H \mathbf{B} \mathbf{w}]. \quad (4.15)$$

Since $\mathbf{p}$, $\mathbf{g}$ and $\mathbf{w}$ are uncorrelated, the total variance of the decision variable is the sum of the variances of the self noise, remaining interference and AWGN, respec-

tively. These variances are given by

$$\begin{aligned}
Var[\mathbf{p}^H \mathbf{B} \mathbf{p}] &= E[|\mathbf{p}^H \mathbf{B} \mathbf{p}|^2] - |E[\mathbf{p}^H \mathbf{B} \mathbf{p}]|^2 \\
&= E \left[\sum_{i=0}^{L-1} \sum_{h=0}^{L-1} \sum_{m=0}^{L-1} \sum_{n=0}^{L-1} p^*(i) B(i, h) p(h) p(m) B^*(m, n) p^*(n) \right] \\
&\quad - \sum_{i=0}^{L-1} \sum_{h=0}^{L-1} B(i, i) B^*(h, h) \\
&= \sum_{i=0}^{L-1} \sum_{m=0}^{L-1} B(i, i) B^*(m, m) + \sum_{i=0}^{L-1} \sum_{h=0}^{L-1} |B(i, h)|^2 - \sum_{h=0}^{L-1} |B(h, h)|^2 \\
&\quad - \sum_{i=0}^{L-1} \sum_{h=0}^{L-1} B(i, i) B^*(h, h) \\
&= \sum_{i=0}^{L-1} \sum_{h=0}^{L-1} |B(i, h)|^2 - \sum_{h=0}^{L-1} |B(h, h)|^2.
\end{aligned} \tag{4.16}$$

Equation (4.16) is further derived by using (4.11) as

$$\begin{aligned}
Var[\mathbf{p}^H \mathbf{B} \mathbf{p}] &= \sum_{i=0}^{L-1} \sum_{h=0}^{L-1} |B(i, h)|^2 - \sum_{h=0}^{L-1} |B(h, h)|^2 \\
&= \sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} |B(i, h)|^2 \\
&= \sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} \frac{1}{L^2} \left| \sum_{l=[i_0 \dots i_{M0}]} D^*(l, i) D(l, h) \right|^2 \\
&= \sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} \frac{1}{L^2} \sum_{l=[i_0 \dots i_{M0}]} D^*(l, i) D(l, h) \sum_{l'=[i_0 \dots i_{M0}]} D(l', i) D^*(l', h) \\
&= \frac{1}{L^2} \sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} \left[\sum_{l=[i_0 \dots i_{M0}]} |D^*(l, i) D(l, h)|^2 + \sum_{l=[i_0 \dots i_{M0}]} \sum_{l' \neq l} D^*(l, i) D(l, h) D(l', i) D^*(l', h) \right]
\end{aligned} \tag{4.17}$$

Thus, $\text{Var}[\mathbf{p}^H \mathbf{Bp}]$ is the summation of two parts as shown in (4.17). Since the modulus of $D^*(l, i)D(l, h)$ is always 1, the first part is easily achieved as

$$\begin{aligned} \frac{1}{L^2} \sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} \left(\sum_{l=[i_0 \dots i_{M_0}]} |D^*(l, i)D(l, h)|^2 \right) &= \frac{1}{L^2} \sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} M_0 \\ &= \frac{1}{L^2} (L^2 - L) M_0 \end{aligned} \quad (4.18)$$

The second part

$$\begin{aligned} &\frac{1}{L^2} \sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} \sum_{l=[i_0 \dots i_{M_0}]} \sum_{l'=[i_0 \dots i_{M_0}], l' \neq l} D^*(l, i)D(l, h)D(l', i)D^*(l', h) \\ &= \frac{1}{L^2} \sum_{l=[i_0 \dots i_{M_0}]} \sum_{l'=[i_0 \dots i_{M_0}], l' \neq l} \left[\sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} D^*(l, i)D(l, h)D(l', i)D^*(l', h) \right] \\ &= \frac{1}{L^2} \sum_{l=[i_0 \dots i_{M_0}]} \sum_{l'=[i_0 \dots i_{M_0}], l' \neq l} \left[\sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} e^{j2\pi \frac{(l-l')(i-h)}{L}} \right] \end{aligned} \quad (4.19)$$

For each $(l - l')$,

$$\sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} e^{j2\pi \frac{(l-l')(i-h)}{L}} = -L \quad (4.20)$$

according to the following property

$$\sum_{i=1}^{L-1} e^{\frac{j2\pi li}{L}} = -1, \text{ if } l \neq 0 \quad (4.21)$$

Since (4.19) has $M_0(M_0 - 1)$ combinations with different l and l' , the second part of (4.17) is

$$\begin{aligned} &\frac{1}{L^2} \sum_{i=0}^{L-1} \sum_{h \neq i, h=0}^{L-1} \sum_{l=[i_0 \dots i_{M_0}]} \sum_{l'=[i_0 \dots i_{M_0}], l' \neq l} D^*(l, i)D(l, h)D(l', i)D^*(l', h) \\ &= M_0(M_0 - 1) \left(-\frac{1}{L} \right). \end{aligned} \quad (4.22)$$

Thus,

$$\begin{aligned} \text{Var}[\mathbf{p}^H \mathbf{B} \mathbf{p}] &= M_0 \left(\frac{L^2 - L}{L^2} \right) + M_0(M_0 - 1) \left(-\frac{1}{L} \right) \\ &= M_0 \left(1 - \frac{M_0}{L} \right). \end{aligned} \quad (4.23)$$

The third part of the variance shown in (4.15) is derived as

$$\begin{aligned} \text{Var}[\mathbf{p}^H \mathbf{B} \mathbf{w}] &= E[\mathbf{p}^H \mathbf{B} \mathbf{w} \mathbf{w}^H \mathbf{B}^H \mathbf{p}] = \frac{1}{\text{SNR}} E[\mathbf{p}^H \mathbf{B} \mathbf{B}^H \mathbf{p}] \\ &= \frac{1}{\text{SNR}} \sum_{i=0}^{L-1} \sum_{k=0}^{L-1} |B(i, k)|^2 \\ &= \frac{1}{\text{SNR}} \left[\sum_{i=0}^{L-1} \sum_{k=0, k \neq i}^{L-1} |B(i, k)|^2 + \sum_i |B(i, i)|^2 \right]. \end{aligned} \quad (4.24)$$

Since the first part of (4.24) was dealt with in (4.17) and (4.23),

$$\begin{aligned} \text{Var}[\mathbf{p}^H \mathbf{B} \mathbf{w}] &= \frac{1}{\text{SNR}} \left[M_0 \left(1 - \frac{M_0}{L} \right) + L \left(1 - \frac{M_0}{L} \right)^2 \right] \\ &= \frac{1}{\text{SNR}} (L - M_0), \end{aligned} \quad (4.25)$$

The last term of the decision variable in (4.12)

$$\begin{aligned} \text{Var}[\mathbf{p}^H \mathbf{B} \mathbf{g}] &= E[\mathbf{p}^H \mathbf{B} \mathbf{g} \mathbf{g}^H \mathbf{B}^H \mathbf{p}] \\ &= E \left[\sum_{n=0}^{L-1} \sum_{m=0}^{L-1} g_c(n) p(n) g_c^*(m) p^*(m) \right] \\ &= \sum_n |g_c(n)|^2 = \frac{1}{L} \sum_k |X_{gc}(k)|^2, \end{aligned} \quad (4.26)$$

where $g_c(n)$ and $X_{gc}(k)$ are the elements of the remaining interference $\mathbf{g}_c = \mathbf{A} \mathbf{D} \mathbf{g}$ and $\mathbf{X}_{gc} = \mathbf{B} \mathbf{g}$, respectively. They are related with the estimation accuracy of $\mathbf{l}$ and δ , which is to be presented in detail in the next two Sections. Then, we have

$$\begin{aligned} \text{SNR}(M_0) &= \frac{E^2[y_0; M_0]}{\text{Var}(y_0; M_0)} \\ &= \frac{(L - M_0)}{\frac{M_0}{L} + \sigma_n^2 + \frac{A^2}{L(L - M_0)} \sum_k |X_{g\mathbf{l}}(k)|^2}. \end{aligned} \quad (4.27)$$

The BER is expressed as a function of M_0

$$P_e(M_0) = Q\left(\sqrt{\text{SNR}(M_0)}\right), \quad (4.28)$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-y^2/2} dy.$$

In the next two Sections, two examples are given to show the necessity of introducing MDPFT for both the narrowband and broadband interference excision. The analytical BERs of the performance for the proposed methods are derived and the optimal value of M_0 is found. Simulation results show the significant performance enhancement of the proposed method compared with the other existing excision methods.

4.3 Narrowband interference excision

Firstly, we deal with the excision of the narrowband tone interference with $g(n)$ of (4.2) in the form of $e^{j2\pi(k_0n+p)}$ as a special case of PPI. In general, this kind of interference is excised by DFT-based [29, 72] or time-weighting DFT [26, 37, 60, 61, 94, 95, 121, 122] based methods, which have been reviewed in Section 4.1. In our proposed method for this case, MDPFT becomes modified DFT, which can be achieved from (4.3) as

$$X(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} r(n) e^{\frac{-j2\pi(kn+\delta n)}{N}}, \quad 0 \leq k \leq N-1, \quad -0.5 < \delta \leq 0.5. \quad (4.29)$$

The inverse MDFT is

$$r(n) = \frac{p(\delta)}{\sqrt{N}} \sum_{k=0}^{N-1} X(k) e^{\frac{j2\pi kn}{N}}, \quad (4.30)$$

where

$$p(\delta) = e^{\frac{j2\pi\delta n}{N}}, \quad 0 \leq n \leq N-1. \quad (4.31)$$

For the signal that contains a single tone interference, the steps of MDFT-based excision process can be summarized as follows:

- (i). Compute the DFT of the interfered signal;
- (ii). Estimate δ defined in (4.6) by the interpolated DFT method;
- (iii). Use the estimated δ to compute the MDFT of the interfered signal defined in (4.29);
- (iv). Remove the frequency component with the highest magnitude and its $M_0 - 1$ neighboring components on both sides in the MDFT domain;
- (v). Compute the IMDFT, according to (4.30), of the output obtained from step (iv) to recover the desired signal.

Compared with the traditional DFT-based method [89, 121, 122], the proposed one requires the estimation of δ and the inverse DFT is replaced by the inverse MDFT. The extra computational cost is a few multiplications for step (ii) and one DFT computation for step (i). It is worth mentioning that the MDFT-based excision method can be easily extended to multi-component interference case if the components are well separated because the estimation performance of δ does not

deteriorate due to the existence of other components. The examples of the multi-component interference excision is introduced in detail in the next Section.

4.3.1 Performance analysis

The performance analysis of the proposed excision method is presented with respect to the estimation accuracy of δ . The optimal number of the components excised is to be found according to the derived BER performance. We follow the derivation in Section 4.2.3 and continue to derive $Var(\mathbf{p}^H \mathbf{B} \mathbf{g})$ in (4.26). Let $\Delta\delta$ represent the estimation error of δ . To achieve (4.26), the residual energy of the interference is computed from the original energy of $g(n)$ subtracted by the energy of the components excised. The energy of the component with the maximum magnitude and its m th pair of components on both sides are $\frac{\sin^2(\pi(\Delta\delta))}{(\pi(\Delta\delta))^2}$, $\frac{\sin^2(\pi(m-\Delta\delta))}{(\pi(m-\Delta\delta))^2}$ and $\frac{\sin^2(\pi(m+\Delta\delta))}{(\pi(m+\Delta\delta))^2}$, respectively. By using the equation

$$\sin(x+a) = \sin(a) + \cos(a)x - \frac{\sin(a)}{2!}x^2 + \cdots + \frac{\sin(a + \frac{n\pi}{2})}{n!}x^n + \cdots, \quad (4.32)$$

we have

$$\frac{\sin^2(\pi(\Delta\delta))}{(\pi(\Delta\delta))^2} = 1 - \frac{\pi^2}{3}\Delta\delta^2 + O(\Delta\delta^2), \quad (4.33)$$

$$\frac{\sin^2(\pi(m-\Delta\delta))}{(\pi(m-\Delta\delta))^2} = m^{-2}\Delta\delta^2 + O(\Delta\delta^2), \quad (4.34)$$

$$\frac{\sin^2(\pi(m+\Delta\delta))}{(\pi(m+\Delta\delta))^2} = m^{-2}\Delta\delta^2 + O(\Delta\delta^2), \quad (4.35)$$

which show that the energy of the m th pair of components are similar when $\Delta\delta^2$ is small. Without loss of generality, let us assume that the pair of two components related to (4.34) and (4.35) either remain or are removed and thus, M_0 is an odd

value. Equation (4.26) can be further expressed by using (4.33), (4.34) and (4.35) as

$$\begin{aligned}
 & \text{Var}(\mathbf{p}^H \mathbf{B} \mathbf{g}) \\
 &= E \left[L \left(1 - \frac{\sin^2(\pi(\Delta\delta))}{(\pi(\Delta\delta))^2} - \sum_{m=1}^{(M_0-1)/2} \left(\frac{\sin^2(\pi(m-\Delta\delta))}{(\pi(m-\Delta\delta))^2} + \frac{\sin^2(\pi(m+\Delta\delta))}{(\pi(m+\Delta\delta))^2} \right) \right) \right] \\
 &\approx L \left(\frac{\pi^2}{3} - \left(\sum_{m=1}^{(M_0-1)/2} 2m^{-2} \right) \right) E[\Delta\delta^2], \tag{4.36}
 \end{aligned}$$

which shows that the residual energy of the interference decreases with the reduction of $E[\Delta\delta^2]$ and becomes zero when $E[\Delta\delta^2] = 0$. From (4.27), (4.28) and (4.36), we have

$$\begin{aligned}
 \text{SNR}(M_0) &= \frac{E^2[y_0; M_0]}{\text{Var}(y_0; M_0)} \\
 &= \frac{(L - M_0)}{\frac{M_0}{L} + \sigma_n^2 + \frac{LA^2}{L-M_0} \left(\frac{\pi^2}{3} - \left(\sum_{m=1}^{(M_0-1)/2} 2m^{-2} \right) \right) E[\Delta\delta^2]}. \tag{4.37}
 \end{aligned}$$

The BER is expressed as a function of M_0

$$P_e(M_0) = Q \left(\sqrt{\text{SNR}(M_0)} \right), \tag{4.38}$$

Based on (4.37) and (4.38), let us discuss the necessity of using the MDFT and the optimal number of M_0 . It is well known that $E[\Delta\delta^2]$ is the summation of the estimation variances $\text{Var}(\Delta\delta)$ and the square of the bias $E[\Delta\delta]$. For time weighting DFT-based methods [121, 122], the bias is in the range of $(-0.5, 0.5]$ and decided by the frequency of the interference. Thus, the performance changes significantly for different interferences. With the MDFT, the modulus of the bias $\Delta\delta$, limited to $0.002/L^2$, is decided by the worst biases of the interpolated DFT algorithm in (4.6) and is significantly smaller compared with the range of $(-0.5, 0.5]$. Without

considering the estimation variances, Figure 4.6 shows the performance comparison in terms of BER by varying biases and M_0 . It can be seen that the performance deteriorates significantly with the increasing biases regardless of M_0 for DFT-based method. Therefore, it is critical to use MDFT instead of DFT for the excision algorithm.

Figure 4.6 Comparison between BERs of the MDFT-based excision methods.

It is obvious that, when $E[\Delta\delta^2] = 0$, the performance with $M_0 = 1$ yields the best performance for any ISR with the minimum distortion of the desired signal because increasing the number of the excised components only results in decreasing the energy of the desired signal. For other fixed value of $\Delta\delta$, for example $\Delta\delta = 0.001$, Figure 4.6 also shows that the optimal M_0 yielding the lowest BER varies with ISR and there exists a large range of ISR in which the optimum performance is achieved by $M_0 = 1$. This is because, when M_0 is increased and ISR is in this range,

the performance enhancement brought by decreasing the variances of the residual interferences is relatively smaller than the deterioration caused by decreasing the means of the desired signal. When ISR is out of the range, i.e., $\text{ISR} > 50\text{dB}$, the optimum performance is achieved by increasing M_0 . However, the BER performance in such a case significantly degrades. Therefore, the case when $Pe(1)$ is minimum is more desirable than that when $Pe(M_0)$ with $M_0 > 1$ is a minimum.

The condition that $Pe(1)$ becomes the minimum is

$$Pe(M_0) - Pe(1) > 0, \quad 1 < M_0 < L,$$

which is equivalent to

$$\text{SNR}(M_0) - \text{SNR}(1) < 0, \quad 1 < M_0 < L. \quad (4.39)$$

By putting (4.37) into (4.39), we get

$$\begin{aligned} & \left(\frac{3(L-1)^2 \left(\sum_{m=1}^{(M_0-1)/2} 2m^{-2} \right) - \pi^2(M_0-1)(2L-1-M_0)}{3(M_0-1)(L-M_0)(L-1)} \right) E[\Delta\delta^2] \\ & < \frac{1 + \sigma_n^2}{A^2 L}. \end{aligned} \quad (4.40)$$

When the left side of (4.40) is positive, i.e.

$$3(L-1)^2 \left(\sum_{m=1}^{(M_0-1)/2} 2m^{-2} \right) - \pi^2(M_0-1)(2L-1-M_0) > 0, \quad (4.41)$$

we have

$$M_0 < L - (L-1) \sqrt{1 - \frac{3 \left(\sum_{m=1}^{(M_0-1)/2} 2m^{-2} \right)}{\pi^2}}, \quad (4.42)$$

and

$$M_0 > L + (L-1) \sqrt{1 - \frac{3 \left(\sum_{m=1}^{(M_0-1)/2} 2m^{-2} \right)}{\pi^2}} > L. \quad (4.43)$$

Equation (4.43) is ignored because M_0 cannot be greater than L . With some necessary manipulations, the condition in (4.40) is further expressed as

$$\begin{aligned}
 E[\Delta\delta^2] &= \text{Var}(\Delta\delta) + E^2[\Delta\delta] \\
 &< \frac{3(1 + \sigma_n^2)(M_0 - 1)(L - M_0)(L - 1)}{A^2 L (3(L - 1)^2 (\sum_{m=1}^{(M_0-1)/2} 2m^{-2}) - \pi^2 (M_0 - 1)(2L - 1 - M_0))} \\
 &= \frac{(M_0 - 1)(1 + \sigma_n^2)}{A^2 L (\sum_{m=1}^{(M_0-1)/2} 2m^{-2})} + O(L^{-2}), \tag{4.44}
 \end{aligned}$$

where the last line is obtained by decomposing the term in the second line into a parallel summation form, and $O(L^{-i})$ is the summation of the terms containing factors of $1/L^k$ for $k \geq i$. It is also noted that $Pe(1)$ is the minimum regardless of $E[\Delta\delta^2]$ when the condition in (4.42) is not met because (4.40) becomes negative in this case.

It was shown in Figure 3 of [66] that the variance of the δ estimation in (4.6) is upbounded by

$$\text{Var}(\Delta\delta) = 1.96 \text{CRLB}(\delta) = 1.96 L^2 \text{CRLB}(k_0) = \frac{11.76 L (1 + \sigma_n^2)}{4\pi^2 A^2 (L^2 - 1)}, \tag{4.45}$$

where $\text{CRLB}(\delta) = L^2 \text{CRLB}(k_0)$, which can be obtained from (4.29), and [66]

$$\text{CRLB}(k_0) = \frac{6}{4\pi^2 (\text{ISNR}) L (L^2 - 1)} = \frac{6(\sigma_n^2 + 1)}{4\pi^2 A^2 L (L^2 - 1)}.$$

With (4.45) and the worst-case bias $E[\Delta\delta] = \pm 0.002/L^2$, (4.44) is further derived as

$$E[\Delta\delta^2] \approx \frac{11.76 L (1 + \sigma_n^2)}{4\pi^2 A^2 (L^2 - 1)} + \left(\frac{0.002}{L^2} \right)^2 \tag{4.46}$$

$$< \frac{(M_0 - 1)(1 + \sigma_n^2)}{A^2 L (\sum_{m=1}^{(M_0-1)/2} 2m^{-2})} + O(L^{-2}). \tag{4.47}$$

Because Figure 4.6 shows that the optimal M_0 increases with the increase of ISR, the threshold of ISR, when $Pe(1)$ is minimum, is achieved by solving (4.47) in terms of A^2 :

$$A^2 < \frac{10^6(1 + \sigma_n^2)(M_0 - 1 - 0.3 \sum_{m=1}^{(M_0-1)/2} 2m^{-2})}{4 \sum_{m=1}^{(M_0-1)/2} 2m^{-2}} L^3, \quad (4.48)$$

where the terms related to $L^i (i < 3)$ in (4.48) are omitted. Let us define $ISR = A^2$ and

$$Th(M_0) = \frac{10^6(1 + \sigma_n^2)(M_0 - 1 - 0.3 \sum_{m=1}^{(M_0-1)/2} 2m^{-2})}{4 \sum_{m=1}^{(M_0-1)/2} 2m^{-2}} L^3,$$

to be the threshold of ISR for $Pe(1)$ being smaller than $Pe(M_0)$ when the condition in (4.42) is met. It can be easily verified that the $Th(M_0)$ increases as M_0 , L and σ_n^2 increase. When L and σ_n^2 are fixed, therefore, $Pe(1)$ yields the minimum BER if $ISR < Th(3)$. For example, $Th(3)$ is about 92 dB when $L = 16$ and $\sigma_n^2 = 1$, which suggests that excising only one bin achieves the best performance within a large range of ISR.

When $ISR > Th(3)$, there are two possible methods to obtain a better performance. The first one is to reduce $E^2[\Delta\delta^2]$ in (4.46) by increasing L to satisfy (4.48). The other one is to increase M_0 to find the optimal M_0 that satisfies (4.48). Since it was mentioned before that the BER performance degrades severely when ISR is out of the range, the first one providing a better performance is generally much more desired than the second one.

4.3.2 Simulation results

The simulation is implemented according to the block diagram shown in Figure 4.7.

In the simulations, it is assumed that the transmitted signals are sampled at the

Figure 4.7 Block Diagram for simulations.

rate of once per chip and is corrupted by mono-component tone interferences $g(n)$, the length of the PN sequence is 16 and the SNR is zero dB. The transmitted data, which is either 1 or -1, is generated from uniform distributed random variable. The BER achieved from each simulation is computed as the average of 10^6 runs. The ISR is adjusted by changing A . It can be seen that general agreement is achieved between the theoretically calculated and simulated BERs.

The first simulation is to show the necessity for using the MDFT instead of the DFT to excise the interference. The transform block length is 128 and the three tone interferences used in each experiment are

$$g_i(n) = e^{j2\pi(\frac{8-\Delta\delta_i}{128})}, \quad i = 1, 2, 3, \quad (4.49)$$

where $\Delta\delta_1 = 0.01$, $\Delta\delta_2 = 0.2$ and $\Delta\delta_3 = 0.4$, respectively. It can be seen from (4.49) that the three tone interferences are with different grid biases determined by $\Delta\delta_i$.

During the excision process in this experiment, there is no estimation of δ and it is assumed that $E[\Delta\delta^2]$ is introduced only by the biases and the interference excision is performed by turning the maximum component to zero in the frequency domain, as shown in the excision procedures given in the previous Section. Based on the theoretical calculation of (4.28), Figure 4.8 presents the analytical BERs expressed by various lines and simulated BERs represented by markers for different values of $\Delta\delta_i$. It is evident that the BER performance varies considerably with $\Delta\delta$ and is significantly improved with the reduction of $\Delta\delta$. When $\Delta\delta = 0$, for example, the BER is always smaller than 10^{-4} , which is a nearly ideal performance. Therefore, it is critical to limit $\Delta\delta$ as small as possible, which can be achieved by using the proposed MDFT-based method.

Figure 4.8 Comparison of BERs for interferences with different grid biases.

The second experiment shows how the performance of the proposed excision method varies with M_0 . The block length in this experiment is 16 and the interference is defined as

$$g(n) = e^{j2\pi(\frac{8-\Delta\delta}{128})},$$

where $\Delta\delta$ is randomly selected within the range $(-0.5, 0.5]$. The interference is removed by the processing steps described in Section 4.3.

Figure 4.9 Comparisons of BERs achieved by the MDFT-based method.

Figure 4.9 (a) shows that, when ISR is relatively small, for example, < 94 dB, the excision method with $M_0 = 1$ yields the best performance compared with that $M_0 > 1$. As ISR increases, $Pe(3)$ yields the optimal performance with the least BER shown by the diamonds markers in Figure 4.9(a), which satisfies the theoretical ISR threshold shown in (4.48) for the condition that $Pe(1)$ yields the best performance. This is because the ISR is so high that the residual energy of the interference after

excising the bin with the highest magnitude still causes considerably undesirable distortion. It is expected to excise more components though this brings more distortion of the desired signal.

The last simulation shows the robustness of the proposed MDFT-based method compared with the DFT-based method and the time-weighting DFT-based method using Hamming windows. The block length is 128 and the interference is the same as that in the second experiment. For DFT-based and time-weighting method, 9 components are excised. Figure 4.10 shows that the BER obtained by the MDFT-based method is significantly better than that achieved by the original DS-SS system without any excision and that achieved by using DFT-based method with different time-weighting algorithms. This is because, with the estimated δ , the tone interference is confined within a few side lobes by the MDFT so that the interference is effectively suppressed by turning only one MDFT bin to zero. Thus the distortion of the desired signal due to the suppression operation is minimized.

4.4 Broadband interference excision

This Section considers broadband linear chirp interference $g(n)$ with the form of $e^{j2\pi(l_1 n^2 + k_0 n + p)}$ where l_1 is also known as chirp rate. To excise this kind of interference, the 2nd order MDPFT which is also called as modified discrete chirp Fourier transform (MDCFT) is used as

$$X_{l_1}(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} r(n) e^{-j2\pi(n^2 l_1 + \frac{kn + \delta n}{N})}, \quad 0 \leq k \leq N-1, \quad (4.50)$$

Figure 4.10 Comparison of BERs achieved by different excision methods.

where $r(n)$ is the interfered signal. For each fixed l_1 , the inverse MDCFT (IMDCFT) is

$$x(n) = \frac{d(\delta)}{\sqrt{N}} \sum_{k=0}^{N-1} X_{l_1}(k) e^{\frac{j2\pi kn}{N}}, \quad (4.51)$$

where

$$d(\delta) = e^{j2\pi(n^2 l_1 + \frac{\delta n}{N})}, \quad 0 \leq n \leq N-1 \quad (4.52)$$

The parameter δ is the deviation of the discrete chirp Fourier transform (DCFT) to be shifted to meet the condition that $[k_0 N - \delta]_{\text{mod } N}$ is an integer, so that the components of the interference in the side lobes are located at the zero points and the maximum component is at the middle point of the main lobe.

For the signal that contains a single chirp interference, the steps of MDCFT based excision process can be summarized by the following steps:

- (i). Estimate the chirp rate of the interference by HAF method in (2.3);
- (ii). Compute DCFT with the estimate chirp rate;
- (iii). Estimate δ defined in (4.6) by the interpolated DFT method;
- (iv). Use the estimated δ to compute the MDCFT defined in (4.50);
- (v). Remove the highest peak in the main lobe of the MDCFT;
- (vi). Compute the IMDCFT, according to (4.51), of the output obtained from step (v) to recover the desired signal.

It was derived in the previous Section that the performance with $M_0 = 1$ is much more desired than those $M_0 > 1$. If the estimation deviation of the interference is small enough, only the highest peak is removed in step (v). The main computational cost of the process is four fast Fourier transform (FFTs) operations for steps (i), (ii), (iv) and (vi), which is $2N \log_2 N$ in terms of complex multiplications. Therefore, the computational cost of the proposed method is in the order of $O(N \log_2 N)$, which makes the proposed method attractive for on-line applications. In contrast, other reported methods [7, 21, 64] for linear chirp-alike interference excision generally require a higher order of computational complexity because of using instantaneous frequency estimation of the interference [7] and matching pursuit algorithm [21], or time frequency transform (TFT) synthesis [64]. Table 4.1 compares the computational complexity of these methods.

Excision methods	Computational complexity
MDCFT based	$O(N \log N)$
open-loop adaptive filter [7]	$O(N^2 \log N)$
TFT-synthesis based [64]	$O(N^2 \log N)$
Chirplet decomposition [21]	$O(N^2 \log N)$

Table 4.1 Comparison of the computational complexity

4.4.1 Performance Analysis

The performance analysis of the proposed excision method is presented with respect to the estimation accuracy of l_1 and δ . To calculate the BER, (4.26) is firstly computed. Let Δl_1 and $\Delta \delta$ represent the estimation deviation of l_1 and δ , respectively. Since only the highest peak in the MDCFT domain is removed, (4.26) becomes

$$\begin{aligned} \text{Var}(\mathbf{p}^H \mathbf{B} \mathbf{g}) &= \frac{1}{L} E[|X_{gl_1}(k)|^2] \\ &= E \left[L \left(1 - \frac{1}{L^2} \left| \sum_{n=1}^{(L-1)} e^{j2\pi(\Delta l_1 n^2 + \frac{\Delta \delta}{L} n)} \right|^2 \right) \right], \end{aligned} \quad (4.53)$$

By using the equation

$$e^x = 1 + x + \frac{x^2}{2!} + \cdots + \frac{x^n}{n!} + \cdots, \quad (4.54)$$

and the assumption that $\Delta \delta$ and Δl_1 are uncorrelated, (4.53) can be further derived as

$$\text{Var}(\mathbf{p}^H \mathbf{B} \mathbf{g}) \approx L \pi^2 \left(\frac{1}{3} E[\Delta \delta^2] + \frac{16}{45} E[\Delta l_1^2] L^4 \right). \quad (4.55)$$

It can be easily seen that the variance is directly related with the estimation accuracy of the deviation $\Delta \delta$ and Δl_1 . Then, the upper bound BER can be acquired by

putting the worst-case biases and variances of the $\Delta\delta$ and Δl_1 into (4.26). The worst-case biases and variances of $\Delta\delta$ have been derived as in (4.46) except that

$$\text{CRLB}(k_0) = \frac{24}{(\text{ISNR})\pi^2 L^3} = \frac{24(1 + \sigma_n^2)}{A^2 \pi^2 L^3}. \quad (4.56)$$

as shown in (2.3) with $M = 1$ and $m = 1$. Since the HAF-based method shown in (2.20) with $\tau = L/2$ is used to estimate the chirp rate, l_1 is then estimated as $l_1 = f_0/L$ if it is supposed that the estimated frequency of the HAF is f_0 . It has been derived in [90] that the statistical variances of the Δl_1 estimated by HAF is

$$\text{Var}(\Delta l_1) = \left(\frac{16}{15} + \frac{8}{15(\text{ISNR})}\right) \text{CRLB}(\Delta l_1) \quad (4.57)$$

$$= \left(\frac{16}{15} + \frac{8A^2}{15(1 + \sigma_n^2)}\right) \frac{45(1 + \sigma_n^2)}{2\pi^2 L^5 A^2}. \quad (4.58)$$

Since interpolated DFT [66] is also used to refine the estimation of l_1 by HAF-based method, the worst case bias of f_0 is $0.002/L^3$, and the worst case bias of Δl_1 is

$$E[\Delta l_1] = E[\Delta f_0/L] = 0.002/L^4. \quad (4.59)$$

By combining (4.55), (4.58), (4.59), and the BER obtained in (4.28), the upper bound BER performance of the MDCFT based excision method can be readily obtained, as shown in Figure 4.13 in Section 4.4.3.

4.4.2 Multi-Component PPI

The MDCFT based excision method can also be extended to effectively deal with the multi-component¹ interference. The signal containing multi-component interference

¹The term 'components' used here is different from those used before. The components here refer to those with different phase parameters in the time domain

has more than one chirp rate, each is associated with a particular chirp-alike component. The chirp rates, thus, cannot be estimated directly by using the HAF-based method as the mono-component interference because there may exist spurious peaks when the interference share with the same chirp rates [11]. The product HAF-based method shown in (2.21) instead of the HAF-based method is used to estimate the chirp rates for multi-component PPI.

In the MDCFT domain, these components can be overlapping or non overlapping. If the MDCFT of the signal is computed by matching l_1 with a particular chirp rate, Figure 4.11 shows the MDCFT of an interference containing two chirps. The spectrum has one peak value, which is the component whose chirp rate is used in the MDCFT computation, and a wide band of small values that belong to other component. When the components are non-overlapping, as shown in Figure 4.11 (a), the previously presented method can be recursively used to deal with each interfering component, respectively. For the overlapping case, as shown in Figure 4.11 (b), however, the interpolated DFT approach to estimating δ is not effective due to the existence of the other component near the location of the peak value of a particular component. When we adjust δ to get the maximum value for the peak of MDCFT, the side lobes of MDCFT with that particular δ are asymptotically the smallest. Thus, this problem can be solved by using a searching technique. The following steps describe this searching method for each chirp-alike component

- Define $\delta_i = i/M_0$, $0 \leq i < M_0$;
- Compute the MDCFT that corresponds to $\delta_i = i/M_0$, $0 \leq i < M_0$;

- Find the δ_i , $0 \leq i < M_0$, that leads to the maximum value in all MDCFTs.

These steps generally require more computational complexity than the interpolated DFT used for the mono-component chirp interference.

The BER performance and computational complexity of the proposed method for these two cases are also different. The performance in the overlapping case generally has a small degradation compared to that in the non-overlapping case because removing one interference component has generally some undesirable effects on the removal process for the remaining components, which can be observed in Section 4.4.3. If it is assumed that the number of the interference components is V , the computational complexity of the non-overlapping case is V times of that for the mono-component interference excision. Compared with the non-overlapping case, the complexity for the overlapping case increases by about $(M_0 + 4)/4$ times because the δ estimation for each component by the searching method requires extra multiplicative complexity of $\frac{M_0}{2} N \log_2 N$.

4.4.3 Simulation Results

This section presents the performance of the proposed excision methods based on the simulation results. The simulation condition is similar to that in Section 4.3.2. The only difference is that we use broadband interference in this Section. In the simulation, it is assumed that the signal is corrupted by chirp-alike interferences, the length of the PN sequence generated is 16, the SNR is zero dB and $N = 128$.

The first simulation is to demonstrate the performance of MDCFT based method

Figure 4.11 MDCFT of the signal containing multiple chirp-like components.

without the δ estimation in terms of the bias $\Delta\delta$. The chirp interferences are

$$g_i(n) = e^{j2\pi(0.001n^2 + \frac{8-\Delta\delta_i}{N})}, \quad i = 1, 2, 3.$$

Figure 4.12 Comparisons of BERs for the interference with different grid biases

Figure 4.13 Comparisons of BERs achieved by using different excision methods.

Based on (4.28), Figure 4.12 presents the analytical BERs expressed by various curves for $\Delta\delta_1 = 0$, $\Delta\delta_2 = 1.28/N$ and $\Delta\delta_3 = 12.8/N$. It can be seen that the BERs (expressed by the cross-hairs) obtained by simulation have a very good agreement with the theoretical results. It is evident that the BER performance is improved with the reduction of $\Delta\delta$. When $\Delta\delta = 0$, for example, the BER is lowest and always smaller than 10^{-4} , which is a very much desired performance because there is only a small deterioration compared with that when there is no interference represented by the dotted curve in Figure 4.12. With the increase of $\Delta\delta$, the ISR range, in which the BER performance is nearly the optimal when $\Delta\delta = 0$, decreases. For example, the BER has a large deviation around 5 dB and 25 dB from the optimal one when $\Delta\delta = 12.8/N$ and $\Delta\delta = 1.28/N$, as shown in Figure 4.12.

In the second simulation, comparisons are made between the proposed MDCFT

based method and the FRFT based method [3]. The interference is defined by

$$g(n) = e^{j2\pi(0.001n^2 + \alpha n)},$$

where α is randomly selected within one frequency bin of the MDCFT. The interference is removed by the steps given in the last Section. The chirp rate is estimated by (2.20) with $M = 1$ and $\tau = N/2$, and the parameter δ is estimated by using the interpolated DFT. Figure 4.13 shows the general agreement between the analytical upper bound obtained from (4.28) and the simulated result. It can be easily seen that the simulation result is lower than the analytical upper-bound. It also shows that the BER of the MDCFT based method is significantly better than that achieved by the original DS-SS system without any excision and that achieved by using the FRFT based method. The main reason on the improvements achieved by the MDCFT based method over the FRFT based method is that the FRFT lacks the solution to reduce the distortion effects of the side lobes. It is worth mentioning that the analytical performance in the range 0 – 10 dB deteriorates by using the HAF-based estimation method because its estimation variances of the chirp rate deviate largely from CRLB under low ISNR [90] due to its nonlinearity.

The next two simulations consider the case in which the signal is corrupted by two chirps. The interference in the third simulation has two non-overlapping chirps defined by

$$g_i(n) = e^{j2\pi(0.001n^2 + \frac{8 - \Delta\delta_i}{N})} + e^{j2\pi(0.002n^2 + \frac{8 - \Delta\delta_i}{N})}, \quad i = 1, 2,$$

where $\Delta\delta_1 = 0$ and $\Delta\delta_2 = 6.4/N$. Figure 4.14 shows that the FRFT based method

Figure 4.14 BERs for signal containing non-overlapping chirp interference.

(a) Non-overlapping

(b) Overlapping

Figure 4.15 The WVDs of the multi-component interference.

becomes almost useless mainly due to the undesirable influence of the side lobes which makes the excision of each chirp component more difficult. Because the two chirp components are overlapping, as seen in Figure 4.15 (b), excision of one component inevitably remove some information of the remaining components, which

has some undesirable effects on the linear chirp property of these components. Figure 4.14 also presents that the performance of MDCFT based method changes significantly with different values of $\Delta\delta$.

In the last simulation, we compare the performance of the interpolated DFT and the searching based methods presented in the last Section. The interferences used in the simulation is defined by

$$\begin{aligned} g_1(n) &= e^{j2\pi(0.001n^2+\alpha_1n)} + e^{j2\pi(0.002n^2+\alpha_2n)}, \\ g_2(n) &= e^{j2\pi(0.001n^2+\alpha_1n)} + e^{j2\pi(-0.002n^2+\alpha_2n)}, \end{aligned}$$

whose Wigner-Ville distributions (WVDs) [64] are shown in Figure 4.15. It is noted that $g_1(n)$ contains two non-overlapping chirps and $g_2(n)$ contains two overlapping chirps. The parameters α_1 and α_2 are randomly selected from one MDCFT bin. The chirp rates of the two components are estimated by PHAF shown in (2.21).

When the chirp components are non-overlapping both methods are effective to achieve a similar BER performance. Figure 4.16 (a) shows that the BER performance achieved by using the interpolated-DFT method is better than the searching method. However, the interpolated DFT method fails to produce a good performance when the two components overlap, as shown in Figure 4.16 (b). It should be noted that the multi-component interference used here is not separable in the time-frequency domain. It is difficult to use the WVD synthesis method [64] to handle such kind of complicated interferences.

Finally, the MDCFT based excision process, including estimation of the chirp rate based on HAF and the computation of MDCFT and IMDCFT, requires a

Figure 4.16 BER performance comparison.

computational complexity in the order of $O(N \log_2 N)$. Furthermore, the MDCFT based method can provide a better BER performance as shown in the previous Figures. Figure 4.13 (b) shows the BER of the MDCFT based method is below 10^{-4} within a wide range of ISR, from 0 dB to 100 dB for mono-component interference. Figures 4.16 (a) and 4.16 (b) show that the MDCFT based method provides about 20 dB to 35 dB more tolerance on ISR for multi-component interference. However, the main limitation of MDCFT based method is that it can not deal with all kinds of interferences, compared with the open-loop adaptive filter based excision method [7]. The MDCFT based method is suitable only for linear chirp interference or the interference that can be approximated by linear chirps.

4.5 Conclusion

In this Chapter, simple and effective methods are presented for both mono- and multi-component PPI excision in DS-SS communication systems. In order to reduce the undesirable influence of the side lobes, the bias δ is introduced and estimated by interpolated DFT or the proposed searching method. With both the bias and the estimated parameters of PPI, the interference can be effectively removed in the MDPFT domain with minimum undesirable influence of the side lobes. It is shown by theoretical analysis and computer simulations that the proposed methods outperform other existing transform based methods in terms of the reduced computational complexity and improved BER performance within a large range of ISR. Therefore, this algorithm is particularly suited to practical applications for polynomial phase interference excision because of its efficiency and easy implementation by digital signal processing hardware. The only drawback of our method is that the performance deteriorates if the amplitude of the interference varies greatly because δ estimation proposed in this Chapter is not accurate.

Chapter 5

The efficient analysis of time-varying signals with LPTFT

5.1 Introduction

In the previous two Chapters, new algorithms are proposed for the analysis of signals which are modeled as PPSs. However, there are occasions that the time-varying signals, such as sinusoidal FM signals and hyperbolic FM signals, cannot be directly modeled as PPSs, or the parameters of the PPSs vary with time. For example, sinusoidal FM signals arise from vibrating or rotating parts of the targets if an oscillating object is illuminated with an incident laser, RF, or acoustic wave [15]. Validated by real data collected in pulse-Doppler radars, sinusoidal FM signals also appear as a result of the so-called jet engine modulation (JEM) phenomenon, which occurs when a radar observes a jet airplane at an aspect angle that scatters electro-

magnetic radiation from the moving parts of the compressor and blade assembly of the engine [45]. Other occurrences can be found in helicopter recognition problems, where the reflected signal from the helicopter is characterized by sinusoidal FM signals [115]. Hyperbolic FM signals arise from the cetacean whistles emitted from the long-finned pilot whales [78]. This kind of signals has also been used in a biological active sonar system as an effective method of hunting and survival [8].

There are many kinds of methods for the analysis of the signals mentioned above. They, generally, can be categorized as filter-based or time-frequency transform based methods. For the filter-based methods [22, 42], the filter structure is adjusted according to the analyzed signals, e.g., phase locked loop tracks the phase of the analyzed signal. However, this method can not track rapid frequency variation. It has been shown in [18] that the performance of these filter-based methods deteriorates with the increasing degree of the non-stationary of the signals and the decreasing of SNR. Because of the superior performance in dealing with time-varying signals, time frequency transforms (TFTs) have been used to analyze those signals [18, 24]. Generally, TFTs can fall into two categories according to whether the transform satisfies linear property. The first category is the linear TFT whose definition uses the analyzed signals only once. Basically, there are STFT [33, 52, 63, 104, 112], Gabor transforms which are the discrete form of STFT [92], and wavelets [92] in this category. The other category is the nonlinear TFT such as the Cohen's class [24, 116, 117], the power classes [44], the hyperbolic class [43], polynomial Wigner Ville Distribution (PWVD) [19] and polynomial L Wigner-Ville distribution (PLWD) [102]. It

has been shown that one class of the nonlinear TFTs performs quite well for signals with a particular type of phases [81]. For example, the Cohen's class performs best for the linear chirp signal [24] and the power class yields the best performance for the signals with time frequency structure as power functions [44, 81]. It is also shown that the mismatch between TFTs with the phase of the analyzed signals causes significant distortion [81]. The Cohen's class has cross-terms for mono-component nonlinear chirp signals brought by this mismatch.

In this Chapter, we focus on linear TFTs because nonlinear TFTs introduce high oscillating cross-terms for multi-component signals due to their nonlinearity. One of the most popular linear TFTs is short time Fourier transform (STFT) because of its simple concept and easy implementation. The STFT has been widely used for many practical applications [63, 92] such as system identification [34] and speech analysis [68]. Nevertheless, STFT suffers from poor resolution and its performance is subject to the window used since it is impossible to have windows providing high resolutions in both time and frequency domains. In most of the application, high resolution is desired. One example is instantaneous frequency estimation which arises in a variety of application including FM demodulation [52] and interference excision for those interference which cannot be modeled as PPSs [64, 76]. Thus, the STFTs with adaptive window length have been proposed in the literature [5, 52, 63, 92] to provide higher resolution. The STFTs with different window lengths are firstly computed at each time instant, and the final window length to be used at each time instant is decided according to the adaptation criteria such as concentration

measurement [24, 52]. To reduce the computational complexity, the STFTs with different window lengths are computed in a recursive manner in [5, 76].

Although the STFTs with adaptive window length have improved the resolution in some sense, they cannot represent desirably the component which is not parallel with the time or frequency axes, e.g., chirp-like (linear FM) signals. To further enhance the resolution of the STFT, an extra parameter related to the first derivative of the instantaneous frequency (IF) of the analyzed signal is included in the computational process. For example, the STFT with adaptive window length is further developed by using new window sets [63]. The window sets are formed by multiplying the traditional windows with different linear chirp signals. Because windows with different chirp rates shear different angles in the time-frequency plane compared with the traditional windows. The STFT with this kind of windows can provide a better performance for the chirp-like signals. In [33, 104], the STFTs employ the fractional Fourier transform (FRFT) of the traditional windows. The appropriate direction of the fractional domain is also related to the first derivative of the IF of the analyzed signal. The adaptation criteria used are minimizing the second-order fractional Fourier transform moments [104] and the generalized bandwidth product (GTBP) [33]. To extend the above mentioned STFTs for the analysis of multi-component signal, we propose a modified STFT as the numeric mean of several STFTs with their window lengths and extra parameters for each component [112]. This is because the above mentioned adaptive techniques utilized some particular adaptation criteria, such as maximum correlation rule [63] and concentra-

tion measurement [52], as a ruler to select windows. Such criteria are suitable only when the signal is mono-component. If the signal contains multiple components, the final result is a kind of compromise between the optimizations for individual components since the optimal window for one component is not optimal for the other components.

By introducing an extra parameter, in general, the resolution of these newly developed STFTs is enhanced especially for the linear chirp-like signal. Unfortunately, it is observed that the performance of the STFTs with only one extra parameter deteriorates with the increase of the nonlinearity of the signal in the time frequency domain. As a result, the local polynomial time frequency transform (LPTFT), referred as the generalization of STFT, was reported to provide high resolution for non-stationary signals [30,53] with a local polynomial function approximating to the IF characteristics of the analyzed signal. To realize the local polynomial function approximation, the LPTFT introduces more extra parameters including the first order derivative and other higher order derivatives of the IF of the analyzed signal. Theoretically, the LPTFT can provide high resolution for any kind of continuous phase signal if the extra parameters required by the LPTFT computation is properly estimated and updated [53]. Unfortunately, the estimation of these extra parameters results in a heavy computational load. This is mainly due to the long overlap between consecutive signal segments on which the estimation process of the extra parameters is based. In order to reduce the computational complexity, attempt is made to reduce the length of overlap between the consecutive segments. However,

there are problems of reduced resolution because windows with small lengths are required for a large length of overlap in the time-frequency domain. This problem is solved by using adaptive window with length adjusted according to the characteristic of the analyzed signals.

This Chapter presents analysis algorithms for non-polynomial-phase time varying signals with multiple components containing white Gaussian and/or impulse noises. Sections 5.2 and 5.3 provide a brief introduction of the modified LPTFTs for the analysis of multi-component signals and the robust form of modified LPTFT to deal with the signals embedded in impulse noises. The key concept of the segmentation used in the efficient realization of modified LPTFT and robust modified LPTFT is introduced in Section 5.4. The implementation details of the proposed algorithm for the modified LPTFT is presented in Section 5.5 from two issues, which are parameter estimation and window length selection. Different from the previously reported algorithms, the proposed one reduces the overlap length between consecutive segments to minimize the number of segments to be processed for the parameter estimation. Effective methods of estimating the parameters from each signal segment are presented. The window lengths used are adaptive with each component of the analyzed signal to avoid the deterioration of resolution due to the reduction of overlap length. Simulations on various kinds of signals with a single or multiple components are conducted to show the validity of the proposed algorithm in Section 5.6. Conclusion is given in Section 5.7.

5.2 Modified LPTFT for multi-component signals

The LPTFT defined in (2.5.3) cannot be directly used for signals containing multiple components because individual signal components have their own parameter $\mathbf{l}(n)$ and window length Q . Let us define the modified LPTFT_{*p*} for signals containing p components with sets of parameters $\mathbf{L}(n) : \{\mathbf{l}_i(n); 1 \leq i \leq p\}$ and window length $Q : \{Q_i; 1 \leq i \leq p\}$ as

$$\text{MLPTFT}_p(n, f) = \sum_{\tau=-\infty}^{\infty} \frac{1}{a} x(n + \tau) e^{-j2\pi f\tau} \sum_{i=1}^p h_i(\tau) e^{-j2\pi \sum_{m=2}^M l_{i,m-1}(n) \frac{\tau^m}{m!}}, \quad (5.1)$$

where

$$a = \left\| \sum_{i=1}^p h_i(\tau) e^{-j2\pi \sum_{m=2}^M l_{i,m-1}(n) \frac{\tau^m}{m!}} \right\|_2,$$

is the scaling factor keeping the signal energy unchanged and $\|\cdot\|_2$ is the 2-norm operation in terms of τ .

Now let us consider the performance of the modified LPTFT_{*p*} for signals containing p non-parallel components with M th-order polynomial phase defined as

$$x(n) = \sum_{i=1}^p x_i(n) = \sum_{i=1}^p A_i e^{j2\pi \sum_{m=0}^M k_{i,m} n^m},$$

where A_i is the amplitude of the i th component $x_i(n)$. It was shown that the optimal $\{\mathbf{l}_i(n)\}$ for the i th component is given as [53]

$$l_{i,s-1}(n) = \frac{d^s \sum_{m=2}^M k_{i,m} n^m}{dn^s} = \sum_{m=s}^M m(m-1) \cdots (m-s+1) k_{i,m} n^{m-s}, \quad (5.2)$$

where $1 \leq i \leq p$ and $2 \leq s \leq M$ is the index of the phase parameters in $x_i(n)$. From

(5.1), the modified LPTFT_p of $x(n)$ can be expressed as

$$\begin{aligned} \text{MLPTFT}_p(n, f) = & \frac{1}{a} \sum_{i=1}^p \{ A_i \delta(f - \sum_{m=1}^M m k_{i,m} n^{m-1}) * \text{FT}_\tau(h_i(\tau)) \\ & + \sum_{\substack{i < q \leq p \\ q=1, q < i}} A_i \delta(f - \sum_{m=1}^M m k_{q,m} n^{m-1}) * \text{FT}_\tau(h_i(\tau) e^{-j2\pi \sum_{m=2}^M [k_{i,m} - k_{q,m}] \frac{\tau^m}{m!}}) \}, \end{aligned} \quad (5.3)$$

where $*$ is the convolution operator and FT_τ represents the Fourier transform in terms of τ . The first term in (5.3) is the useful auto-term and the second one contains the undesirable cross-terms. Generally, the cross-terms have much smaller magnitudes, compared with those of the auto-terms because the phase factor of $e^{-j2\pi \sum_{m=2}^M [k_{i,m} - k_{q,m}] \frac{\tau^m}{m!}}$ is spread in the frequency domain [99]. The modified LPTFT_p generally has fewer cross-terms than the nonlinear TFT [92] and can be approximately viewed as the sum of the optimal LPTFTs of each component. Furthermore, the constant scaling factor a for each window in (5.1) or (5.3) keeps the signal energy ratio between components approximately unchanged after the modified LPTFT_p computation because the influence of the cross-terms is trivial. Also a similar modified form of the LPTFT, which also uses summation of the LPTFTs with several parameters suitable for each component, was proposed in [30]. However, the estimation of $\mathbf{L}(n)$ is based on maximizing the concentration measure [30], which is different from our proposed modified LPTFT_p.

The following experiment shows the validity of the modified LPTFT_p to analyze the multi-component signals. Let us consider the signal containing two linear chirps. If only one set of parameters optimized for one component is used for the computation of LPTFT as shown in Figure 5.1(a) or 5.1(b), only one of the two

chirps can be truthfully represented, but the other one is smeared significantly. Figure 5.1(c) shows the result computed by using the modified LPTFT_p with two sets of the parameters matching for both components, respectively. A good representation is achieved for both components. Next we compare the modified LPTFT_p with WVD, which is the most popular candidate of nonlinear TFTs. It can be seen from Figure 5.1(d) that the cross-terms introduced in the modified LPTFT_p are much smaller than those of WVD.

5.3 Robust modified LPTFT

In most analysis, the noise is assumed to be Gaussian distributed. Although this assumption is valid in many applications, practitioners in various fields of signal processing and communications have found that this assumption sometimes is inappropriate. For example, impulse noises appear in outdoor mobile and portable radio communications, radar and sonar signal processing and financial time-series modeling. New techniques to deal with impulse noises have to be developed. Various impulse noise models have been well studied for several decades in the statistics literature. One of the most popular model is the α -stable distribution [73] introduced as early as 1925 by Paul Levy and has been widely used in variance applications including modeling the noise on telephone lines and financial time-series. Another popular model belongs to the Middleton class A [71] which has been used widely in wireless communications [123].

For the analysis of signal embedded in impulse noises, the traditional maximum

Figure 5.1 The performance comparisons between different TFTs. (a) LPTFT with one set of parameters; (b) LPTFT with one set of parameters; (c) Modified LPTFT_p with two sets of parameters; (d) WVD.

likelihood estimation (MLE) loses its advantages. To deal with the impulse noises, the minimax Huber's robust statistics have been developed as alternatives to the traditional MLE [47]. Recently, the robust form of FT has been proposed using the robust statistics [55], since FT is one of the key tools used in the signal processing society. In this Section, the modified LPTFT is extended to the robust form to

deal with the signals embedded in impulse noises. The robust FT is reviewed in the following subsection.

5.3.1 Robust FT

Generally, DFT can be written as the summation of the product between the signal and the bases of this transform as

$$X(k) = \sum_{n=0}^{N-1} x(n)\phi_k(n), \quad 0 \leq k \leq N-1, \quad (5.4)$$

where $x(n) = s(n) + w(n)$, $s(n)$ is the signal which is generally embedded in the noise $w(n)$, and $\phi_k(n)$, $k \in [0, N-1]$ are the transform bases defined as $\phi_k(n) = e^{-j2\pi kn/N}$ to form a set of orthogonal functions, i.e.

$$\sum_{k=0}^{N-1} \phi_k(n)\phi_k^*(n') = N\delta(n - n'). \quad (5.5)$$

To estimate the signal transform coefficients $S(k)$ from noisy $x(n)$, it is shown in [32] that the following minimization is to be solved

$$L_\phi(m, k) = \sum_{n=0}^{N-1} F(Nx(n)\phi_k(n) - m) \quad (5.6)$$

where $F(e)$ is a loss function and $e(m; n, k) = Nx(n)\phi_k(n) - m$ is the error function. This problem is also known as orthogonal robust regression [46, 113]. Equation (5.6) can be determined from

$$\frac{L_\phi(m; k)}{\partial m^*} \Big|_{m=m_0} = 0, \text{ with } \hat{S}(k) = m_0. \quad (5.7)$$

Solution of (5.7) is defined as M -estimate. For Gaussian noise, the MLE approach

suggests $F(e) = |e|^2$. In this way, the standard DFT is achieved by

$$\widehat{S}(k) = \sum_{n=0}^{N-1} x(n)\phi_k(n) \quad (5.8)$$

However, the performance of the estimation method in (5.8) deteriorates significantly if the noise is not Gaussian, such as the impulse noises because $|e|^2$ is an unsuitable loss function. It has been proved that the suitable loss function for this case should be $|e|$ when the noise is Laplacian. Since the Laplacian noise, which has the longest tail, is the worst case of α -stable impulse noises, robust estimation is achieved by minimizing $F(e) = |e|$ for real signals and $F(e) = |Re(e)| + |Im(e)|$ for complex signals. It has been shown in [55] that this minimization can be solved using the following recursive steps:

- Initialization:

$$S^{(0)}(k) = \sum_{n=0}^{N-1} x(n)\phi_k(n),$$

$$D^{(0)}(n, k) = Nx(n)\phi_k(n) - S^{(0)}(k),$$

$$d^{(0)}(k) = 1 / \sum_{n=0}^{N-1} (1/|D^{(0)}(n, k)|);$$

- For $m = 1, 2, \dots, M$, compute

$$D^{(m-1)}(n, k) = Nx(n)\phi_k(n) - S^{(m-1)}(k),$$

$$d^{(m-1)}(k) = 1 / \sum_{n=0}^{N-1} (1/|D^{(m-1)}(n, k)|),$$

$$\text{and } S^{(m)}(k) = \sum_{n=0}^{N-1} Nd^{(m-1)}(k)\phi_k(n)/|D^{(m-1)}(n, k)|,$$

until

$$m^* = \min_m \{m : |S^{(m)}(k) - S^{(m-1)}(k)|/S^{(m-1)}(k) \leq \eta, m \leq M\},$$

where η and M are predefined values;

- $S(k) = S^{(m^*)}(k)$.

The solution of (5.7) when $F(e) = |Re(e)| + |Im(e)|$ can also be approximated by using the marginal median filter [56, 88] as

$$X_M(k) = median_n \{Nx(n)\phi_k(n)\} \quad (5.9)$$

to avoid the above mentioned iterative procedures, where $median_n$ refers to selecting the median value with regard to n on the real and imaginary parts, respectively. With the help of the robust FT, robust modified LPTFT_p which is extended from modified LPTFT_p defined in (5.1) is proposed in the next subsection.

5.3.2 Robust modified LPTFT_p

The modified LPTFT_p defined in (5.1) is only suitable for signals containing Gaussian noise and achieves poor performance for signals containing impulse noises. To deal with the non-stationary signals embedded in impulse noises, the robust form of modified LPTFT_p is required to be developed. It can be easily found that the modified LPTFT_p defined in (5.1) is also expressed as

$$MLPTFT_p(n, f) = FT_\tau \left\{ \frac{1}{a} \sum_{i=1}^p [x(n + \tau) h_i(\tau) e^{-j2\pi \sum_{m=2}^M l_{i,m-1}(n) \frac{\tau^m}{m!}}] \right\}. \quad (5.10)$$

The RFT [55, 56] can be conveniently used to replace the FT in (5.10) to define robust modified LPTFT_p as follows. We use the suboptimal marginal-median form of the RFT defined in (5.9) [56] and replace FT_τ with median operation as

$$RMLPTFT_p(n, f) = median_\tau \left\{ \left(\frac{1}{a} \sum_{i=1}^p [x(n + \tau) h_i(\tau) e^{-j2\pi \sum_{m=2}^M l_{i,(m-1)}(n) \frac{\tau^m}{m!}}] \right) e^{-j2\pi f\tau} \right\}. \quad (5.11)$$

The computation for the median operation for MLPTFT at each time instance is in the order of N , which is higher compared with mean operation. This is because the mean operation can utilize the fast algorithm of DFT.

5.4 Segmentation

Figure 5.2 Segmentation for $\alpha = 4, 2$ and 0 , and $Q = 5$.

The signal with noise to be analyzed is defined as

$$x(n) = s(n) + w(n), \quad 0 \leq n \leq N - 1, \quad (5.12)$$

where $w(n)$ represents white Gaussian noise or impulse noise and $s(n)$ contains mono- or multi-non-stationary components in the time frequency domain. In this Chapter, the impulse noise either belongs to Middleton class A [71] or belongs to α stable distribution [58, 73]. The input signal $x(n)$ is divided into many small

segments with a window function $h(\tau)$ in the time domain. The j th signal segment is defined as

$$x_j = x[j(Q - \alpha) + \tau] h(\tau), \quad (5.13)$$

$$0 \leq j \leq \lfloor N/(Q - \alpha) \rfloor - 1, \quad 0 \leq \alpha \leq Q - 1, \quad -(Q - 1)/2 \leq \tau \leq (Q - 1)/2,$$

where $\lfloor x \rfloor$ is the function to return the largest integer that is equal to or smaller than x , N is the length of signal $x(n)$, Q , which is assumed to be an odd number without loss of generality, is the length of the window $h(\tau)$ or equivalently the length of the signal segment and α represents the length of the overlap between the consecutive signal segments. Figure 5.2 shows examples for $\alpha = 0$, $Q - 1$ and $(Q - 1)/2$, where $Q = 5$. Heavy computational complexity is needed for estimating the extra parameters required by the LPTFT computation if the overlap length is large because the number of signal segments to be processed is accordingly increased. For the computation of the traditional LPTFT, the number of segments required for the estimation of the extra parameters are N since a new segment is produced at each time instant for the estimation of the parameter $\mathbf{L}(n)$.

5.5 The efficient realization

Major steps about the estimation of the parameter $\mathbf{L}(n)$ and window length Q for computing modified LPTFT in (5.10) and robust modified LPTFT in (5.11) are presented in detail in the following two subsections, respectively.

5.5.1 Estimation of $\mathbf{L}(n)$

This section considers methods estimating $\mathbf{L}(n)$ in (5.10) and (5.11) from the segments achieved by (5.13) to minimize the computational complexity without deteriorating the smoothness of the spectrum. As mentioned previously, the signal is generally divided into many segments and the $\mathbf{L}(n)$ estimation is based on the idea of finding an $(M - 1)$ th order polynomial function to approximate the frequency characteristics of each signal segment. In the previously reported methods [30, 53, 54, 63], the overlap factor α equals $Q - 1$ which means that there are N segments of length Q for an N -point input sequence. In general, several modified LPTFT _{p} s with different sets of parameters are computed for each signal segment. For segment x_j , for example, the $\mathbf{L}(j)$ that yields the maximum value [63] or values larger than a threshold [30] is selected. Because two consecutive signal segments overlap heavily in the reported method [30], i.e., two adjacent signal segments differ by only one data point, this method introduces great redundancy and thus, requires a heavy computational load.

To reduce the overall computational load, it is necessary to minimize the length of overlap between consecutive segments, such as $\alpha < Q - 1$. For segment x_j , the set of parameters $\mathbf{L}(j(Q - \alpha + \tau))$ within the duration $\lfloor -(Q - \alpha - 1)/2 \rfloor \leq \tau \leq \lfloor (Q - \alpha - 1)/2 \rfloor$ is estimated simultaneously. As shown in Figure 5.2, the parameters for the shaded data intervals are estimated from the corresponding signal segment. For example, $\mathbf{L}(2)$, $\mathbf{L}(3)$ and $\mathbf{L}(4)$ are estimated from segment x_2 when $\alpha = 2$ in Figure 5.2 (b). In this way, only $\lfloor N/(Q - \alpha) \rfloor$ instead of N signal segments

are processed to acquire $\mathbf{L}(n)$ at all time instants. Generally, α controls the trade-off between the computational load and the smoothness of the spectrum. When $\alpha = 0$, there is no overlap and only N/Q segments are processed, which reduces the computational load by Q times compared with that with $\alpha = Q - 1$ in the previously reported methods. In general, the modified LPTFT_{*p*} with $\mathbf{L}(n)$ estimated with $\alpha = 0$ yields satisfactory performance to achieve a good polynomial function approximation to the frequency components if the window length Q is small enough, which is further illustrated in the first experiment of Section 5.6. Otherwise, this arrangement may cause problems of un-smoothness of the frequency components where consecutive segments are connected because longer windows are prone to larger differences between the estimated parameters for the consecutive segments. This case occurs when the SNR is extremely low. Because the estimation of $\mathbf{L}(n)$ in each segment cannot work if the SNR is below a certain threshold decided by the statistical performance of the estimation method, windows with longer length have to be used. Under this circumstance, the overlapping factor α is required to be increased. This is further illustrated by the 2nd order modified LPTFT_{*p*} of a sinusoidal FM signal with long window. The signal used has the form of

$$x(n) = e^{-j(256/\pi)\cos(\sqrt{2}\pi n/256)} + w(n), \quad (5.14)$$

where $w(n)$ is the white Gaussian noise with unit variance. We compute the 2nd modified LPTFT_{*p*}s with different overlap factors $\alpha = Q - 1, \lfloor Q/2 \rfloor, \lfloor Q/3 \rfloor, 0$, where $Q = 103$ and $N = 504$, which can be seen from Figure 5.3. When α is large, e.g. $\lfloor Q - 1 \rfloor, \lfloor Q/2 \rfloor$, the signal can be represented well without any smearing. On the

contrary, un-smoothness occurs between consecutive segments e.g. around $n = 100$, as shown in Figure 5.3 (a) and (b), when $Q = 0, \lfloor Q/3 \rfloor$ because the longer window is used with small overlap factors. In the following, we consider the estimation method for $\mathbf{L}(\mathbf{n})$ with Gaussian noises or impulse noises, respectively.

Figure 5.3 The PTFT of a sinusoidal FM signal with different overlap lengths.

(a). Estimation for signals with Gaussian noise

The coefficients of the polynomial function model are estimated by searching the peak locations of the polynomial time frequency transform (PTFT) of the signal segment x_j introduced in Chapter 3. It is assumed that p peaks are found in the

PTFT indicating the p components located at positions $\mathbf{a}_i = \{a_{i,2}, \dots, a_{i,M}\}$, $1 \leq i \leq p$. The parameters in $\mathbf{L}(n)$ needed for the computation of the modified LPTFT _{p} of the i th component are calculated by (5.2) with $k_{i,m}$ being replaced with the estimates $a_{i,m}$ for $m = 2, \dots, M$. Figure 5.4 shows the $\mathbf{L}(n)$ estimation of the segmented sinusoidal FM signal with the same form as (5.14), where $Q = 63$ and $N = 504$. It can be seen that one peak is clearly shown with different positions representing different optimal $\mathbf{L}(n)$ for each segment of the signal $x(n)$. Figure 5.5

(a) PTFT of the 2nd segment

(b) PTFT of the 4th segment

(c) PTFT of the 6th segment

(d) PTFT of the 8th segment

Figure 5.4 The 2nd order PTFT of a segmented sinusoidal FM signal.

shows the PTFT of the segmented signal containing two components as

$$x(n) = e^{-j(256/\pi)\cos(\sqrt{2}\pi n/256)} + e^{-j.002\pi n^2} + w(n), \quad (5.15)$$

where $w(n)$ is the white Gaussian noise with unit variance. It can be seen that two peaks are clearly shown in the figures which indicating two sets of parameters are required for the computation of modified LPTFT_p.

(a) PTFT of the 2nd segment

(b) PTFT of the 4th segment

(c) PTFT of the 6th segment

(d) PTFT of the 8th segment

Figure 5.5 The 2nd order PTFT of a segmented signal.

b. $L(n)$ estimation of robust modified LPTFT_p for signals with impulse noise

The robust RPTFT is derived for the estimation of coefficients from signals with impulse noises. The estimation of phase parameters is the same as that presented in the previous section except that the robust PTFT uses the RFT [55, 56] instead of the FT in the computation of PTFT. Figure 5.6 compares the performances achieved by the PTFT and RPTFT of the segmented signal expressed in (5.15) with α -stable impulse noise $w(n) = 0.7[w_1^3(n) + jw_2^3(n)]$, where $w_1(n)$ and $w_2(n)$ are independent Gaussian random variables with unit variances. The probability density function of this kind of noise is $g(x) = 1/3\sqrt{2\pi}e^{-|x|^{2/3}/2}|x|^{-2/3}$. This noise model has been used in a lot of publications [30, 31, 55, 56] and has been proved to be effective in modeling many real-life engineering problems such as outliers and impulse signals [55]. It is shown that two peaks are easily identified from the robust PTFTs in Figure 5.6 (b), (d) rather than from the PTFTs in Figure 5.6 (a), (c).

5.5.2 Window Length Estimation

In the previous section, $L(n)$ is estimated based on the idea of modeling each segment as an M th-order polynomial phase signal. Therefore, the window length used in the modified LPTFT_p or robust modified LPTFT_p is the same as the length of the segment. It is known that there is a tradeoff between the window length and the resolution of the modified LPTFT_p [54]. In general, approximation errors increase with the window length if the order of the modified LPTFT_p is lower than

(a) PTFT of the 2nd segment

(b) RPTFT of the 2nd segment

(c) PTFT of the 8th segment

(d) Robust PTFT of the 8th segment

Figure 5.6 Comparisons between PTFT and robust PTFT of a segmented signal.

that of the phase of the signal segment. For polynomial phase component whose order is not higher than that of modified LPTFT_p , on the other hand, the modified LPTFT_p gives a better resolution if longer window (or segment) is used. For a good compromise, it is always desired that the length of the segment is adaptively matched to the characteristics of the signal components. In our analysis, the initial window length is selected to be small enough to provide an acceptable accuracy of

the approximation and the actual length of the window is increased according to the properties of the consecutive signal segments.

We intend to increase the window length if consecutive segments have the same polyphase model. Let us assume that two consecutive segments, the j th and $(j+1)$ th segments, belong to the same polynomial phase model. The estimated parameters from the consecutive segments are different due to the existence of the delay. If the j th segment has the phase $2\pi \sum_{m=0}^M k_{i,m} n^m$, the phase of the $(j+1)$ th segment should be $2\pi \sum_{m=0}^M k_{i,m} (n + (Q - \alpha))^m$ because the $(j+1)$ th segment is delayed by a time interval of the segment overlap compared with that of the j th segment. To achieve the difference between the coefficients of the consecutive segments, the following equation is computed firstly

$$\begin{aligned}
 & \left[\sum_{m=0}^M k_{i,m} (n + (Q - \alpha))^m \right] - \left[\sum_{m=0}^M k_{i,m} n^m \right] \\
 &= \sum_{m=0}^M k_{i,m} \sum_{s=0}^m C_m^s n^s (Q - \alpha)^{m-s} - \sum_{m=0}^M k_{i,m} n^m \\
 &= \sum_{m=0}^M k_{i,m} \left[\sum_{s=0}^{m-1} C_m^s n^s (Q - \alpha)^{m-s} + n^m \right] - \sum_{m=0}^M k_{i,m} n^m \\
 &= \sum_{m=0}^M k_{i,m} \sum_{s=0}^{m-1} C_m^s n^s (Q - \alpha)^{m-s}, \tag{5.16}
 \end{aligned}$$

where $C_m^s = s!/(m!(m-s)!)$. For clarity of presentation, we define

$$\sum_{m=0}^M b_m n^m = \sum_{m=0}^M k_{i,m} \sum_{s=0}^{m-1} C_m^s n^s (Q - \alpha)^{m-s}, \tag{5.17}$$

where b_m is the constant coefficient associated with n^m term on the left side of (5.17).

Let us represent the coefficients of the polynomial function estimated from the j th and $(j+1)$ th segments with $a_{j,m}$ and $a_{j+1,m}$, respectively, where $1 \leq m \leq M$. The

difference $(a_{j+1,m} - a_{j,m})$ is compared with b_m . If each $|a_{j+1,m} - a_{j,m} - b_m|$ is smaller than a predefined threshold T_m , these two segments are considered to have the same polyphase model and the length of the two combined windows becomes $2Q - \alpha$. The final window length is the total length of the consecutive segments that have the same polynomial function model. In general, T_m is defined as the summation of two values. The first value is the deviation caused by the estimation of $k_{j,m}$ in the consecutive segments due to the noise influence. This value is decided by the statistical performance of the estimation method introduced in Section 5.5.1. The second one is the difference defined by the user. This factor controls the tradeoff between the resolution and distortion from the real time-frequency characteristics of the signal. Larger value is used if the resolution is the main consideration of the applications. In our simulations in the next subsection, we define the first value as the bias introduced, by the gridding operation during the estimation of $\mathbf{L}(n)$, and the second value as zero.

From the previously described estimation methods of $\mathbf{L}(n)$ and window lengths, the main computational complexity is the computation of several PTFTs so that $\mathbf{L}(n)$ leading to the maximum peak values can be selected. It can be easily seen that compared with algorithm reported in [53], the computational complexity for $\mathbf{L}(n)$ estimation is significantly reduced. This is because, with the segmentation method shown in Figure 5.2, the number of segments for an N -point sequence is reduced to be $\lfloor N/(Q - \alpha) \rfloor$ in comparison with N segments. The estimation of window lengths requires overheads for computation of (5.16) and the costs of comparison with the

given threshold is trivial.

5.6 Experimental results

Two types of signals, which contain a single component and multiple components, respectively, are used to test the performance of the proposed algorithms. For simplicity, the 2nd-order modified LPTFT_p or robust modified LPTFT_p is used in all the experiments dealing with the input sequence $x(n)$ with $N = 504$. The $\mathbf{L}(n)$ is estimated from the positions ($\mathbf{a}_i = \{a_{i,2}, \dots, a_{i,M}\}$, $1 \leq i \leq p$ and $M = 2$) of the peaks in the PTFT.

The first type of signals contains a single component defined as

$$x_1(n) = e^{-j(256/\pi)\cos(\sqrt{2}\pi n/256)} + w(n), \quad (5.18)$$

$$x_2(n) = e^{-j\pi(0.000000003n^4 - 0.001n^2)} + w(n), \quad (5.19)$$

$$x_3(n) = e^{-j2\pi n(\ln(100n+10))/5} + w(n). \quad (5.20)$$

Modified LPTFT_ps with $\alpha = 0$ of these three signals are shown in Figure 5.7 (a), 5.8 (a) and 5.9 (a), respectively. It can be seen that modified LPTFT_p provides good time frequency representation for each kind of signals, including the sinusoidal FM signal $x_1(n)$, the high order PPS $x_2(n)$ and the hyperbolic FM signal $x_3(n)$.

The estimation of the instantaneous frequencies of $x_i(n)$, $1 \leq i \leq 3$, is conducted with Gaussian noise $w(n)$ of different variances. Monte Carlo simulations are performed to obtain the mean square error (MSE) for each estimator. The MSE is defined by $\frac{1}{N} \sum_{n=0}^{N-1} (\tilde{f}(n) - f(n))^2$, where $f(n)$ is the true instantaneous frequency

(a) Modified LPTFT_p of $x_1(n)$

(b) IF estimation of $x_1(n)$

Figure 5.7 The comparisons of MSEs between modified LPTFT_ps.

(a) Modified LPTFT_p for $x_2(n)$

(b) IF estimation of $x_2(n)$

Figure 5.8 The comparisons of MSEs between modified LPTFT_ps.

and $\tilde{f}(n)$ is the estimation of $f(n)$ according to the curve peak positions in the modified LPTFT_p of $x_1(n)$. The MSEs are compared for different overlap lengths $\alpha = 0$, $\lfloor Q/2 \rfloor$, $\lfloor Q/3 \rfloor$ and $Q - 1$, as shown in (b) of Figures 5.7, 5.8 and 5.9. A large range of window lengths ($6 < Q < 80$) have been tried. Only the results using $Q = 23$,

(a) modified LPTFT_p for $x_3(n)$ (b) IF estimation of $x_3(n)$ Figure 5.9 The comparisons of MSEs between modified LPTFT_ps.

$Q = 33$ and $Q = 43$ are shown because the MSEs achieved with these windows are mostly below 10^{-2} .

We take Figure 5.7 (b) as an example. It is observed that for high SNRs, smaller window length generally gives smaller MSEs, for example, the curve for $Q = 23$ gives the best performance when $\text{SNR} > 7$ dB. With low SNRs, i.e., $\text{SNR} \leq 7$ dB, larger window length yields lower MSEs. This is because that MSEs are mainly influenced by the biases and the variances of the input sequence [53]. When SNR is high, the MSEs are mainly affected by the biases which increase with the increase of window length. When SNR is low, the variances of the signal are the dominant factor affecting the MSEs. The variances decrease with the increase of window length so that the MSEs achieved with the longer window become relatively small. It is worth mentioning that, when SNR is extremely low, e.g., below 0 dB, MSEs deteriorate significantly. This is because the windows used in our proposed modified LPTFT_p

are generally with smaller length. The use of narrow window leads to the increasing influence of noise especially for low SNR.

The most important observation made in Figure 5.7 (b) is that the MSE performances for different overlap lengths are very close to each other regardless of the window lengths, which leads to the conclusion that the decrease of overlap length between segments does not significantly deteriorate the performance of modified LPTFT_p. For the MSE comparisons of the other signals which are shown in Figure 5.8 (b) and 5.9 (b), we reach the similar conclusions.

Figure 5.10 The comparison between modified LPTFT_ps.

The second type of signal contains multiple components, which is defined as:

$$x(n) = e^{-j(256/\pi)\cos(\sqrt{2}\pi n/256)} + e^{-j(.002\pi n^2)} + e^{-j(.002\pi n^2 + 0.06\pi n)} + w(n), \quad (5.21)$$

where $w(n)$ is the Gaussian noise with unit variance. This signal is used to test the performance of modified LPTFT_p using adaptive window length. Figure 5.10 shows the modified LPTFT_ps of $x(n)$ that are computed with and without using adaptive

window lengths. In each segment with $\alpha = 0$, three peaks are detected from the corresponding PTFTs. By comparing the difference between the estimated parameters in the consecutive segments with b_m in (5.17), the algorithm finds that two peaks in the consecutive segments belong to the same components and the window length is enlarged to N to obtain a high resolution. It is shown that the resolution in Figure 5.10 (b) for the linear component of $x(n)$ is improved significantly and the two parallel chirp components are clearly distinguished in comparison with Figure 5.10 (a) in which the window length is fixed.

Figure 5.11 The comparison for the signal with α stable impulse noises.

In the previous experiments, all $w(n)$ is white Gaussian noise. Next, $w(n)$ becomes the impulse noise. Two types of impulse noises are used. The first one is α -stable impulse noise where $w(n) = 0.9[w_1^3(n) + jw_2^3(n)]$. The second one belongs to Middleton class A model which is used widely in wireless communications [123]. It is the summation of a Gaussian process and an output of a linear

(a) The modified LPTFT_p(b) The robust modified LPTFT_p

Figure 5.12 The comparison for the signal with impulse noises.

filter excited by a Poisson process [71, 123]. The one we used here is with the form of $w(n) = w_1(n) + \sum_{i=1}^{40} w_3(n - \tau_i)$, where $w_3(n) = 10\delta(n)$ and τ_i forms poisson processes. Figures 5.11 and 5.12 present the modified LPTFT_p and robust modified LPTFT_p for signal $x_1(n)$ in (5.18). It is clear that a good performance is achieved by using the RFT in robust modified LPTFT_p instead of the FT in modified LPTFT_p.

The signal $x(n)$ used in the next experiment is

$$x(n) = e^{-j(256/\pi)\cos(\sqrt{2}\pi n/256)} + e^{-j(.002\pi n^2)} + w(n), \quad (5.22)$$

where $w(n) = 0.9[w_1^3(n) + jw_2^3(n)]$. The components smear significantly in Figure 5.13 (a) because the FT used in the modified LPTFT_p is not able to substantially suppress the impulse noise. However, the robust modified LPTFT_p yields much better performance and the two components can be clearly seen in Figure 5.13 (b).

Figure 5.13 The comparison for signals with α stable impulse noises.

5.7 Conclusion

This Chapter presents analysis algorithms effectively dealing with time varying multi-component signals embedded in either Gaussian or impulse noise. In particular, these algorithms allow the reduction of computational complexity by minimizing the length of overlap between consecutive signal segments. In this way, the whole computational cost is reduced significantly without compromising the performance of traditional LPTFT in terms of MSEs of IF for the analyzed signals. The window length is adjusted according to the characteristics of each component. Experiments show that by using the proposed algorithms of parameters estimation and adaptive window length, the signals containing a single or multiple components with Gaussian and impulse noises can be more accurately represented in the time-frequency domain.

Chapter 6

Conclusion and Future work

6.1 Conclusion

In Chapter 3, a fast algorithm of the polynomial time frequency transform (PTFT) is proposed for the maximum likelihood estimation (MLE) of polynomial phase signals (PPSs). The computational complexity is reduced by $2^M \log_2(N)$ times for the $(M+1)$ th order PTFT with N -point input. Two properties of PTFT are discovered for this computational savings. The first is the quasi-periodic property which is used to reduce the total number of PTFT points to be computed. The second is the key property which exploits the redundancy along each dimension of the PTFT. Based on these two properties, the computation of PTFT is decomposed into the same stages as in the decimation-in-time algorithm of FFT. The computational structure for the fast algorithm similar to that of FFT is also developed. Experiments show that the time of computation is significantly saved by using the proposed fast

algorithm especially for higher order PTFT.

A new transform-based method is proposed in Chapter 4 for the excision of the interference, which can be modeled as PPSs, to minimize the undesirable influence brought by the side lobes. This method is based on the proposed modified discrete polynomial Fourier transform (MDPFT) which provides optimal concentration capability for PPI with any parameters. The MDPFT is developed from the discrete polynomial Fourier transform (DPFT) with an extra parameter representing the bias introduced by the gridding operation during the computation of the DPFT. Two examples are introduced for the excision of narrowband and broadband PPI, respectively. Detailed performance analysis is given in terms of analytical BERs. For narrowband PPI excision, it is shown that excising one component in the MDPFT domain yields the best performance in a large range of ISR. Out of this range, the performance of the excision method deteriorates rapidly and the optimal number of components excised increases with ISR. The MDPFT-based excision method is also extended for the excision of multi-component PPI. Two cases of multi-component PPI, which are overlapped or non-overlapped in the MDPFT domain, have been dealt with and different estimation methods for the extra fractional factor are proposed for computing the MDPFT. Several simulations have been conducted to show the necessity of using the MDPFT for the application of interference excision and the performance enhancement compared with other transform-domain based excision methods.

In Chapter 5, modified LPTFT is proposed with several sets of parameters opti-

mized for each component of the analyzed signal. Robust form of modified LPTFT is extended from modified LPTFT by using robust FT to deal with signals embedded in impulse noises. An efficient realization of modified LPTFT and robust modified LPTFT is proposed based on the idea of reducing the overlap between the segments required for estimating the parameters needed by computing modified LPTFT and robust modified LPTFT. Effective parameter estimation for computing modified LPTFT and robust modified LPTFT is realized by using PTFT and RPTFT to deal with Gaussian and impulse noises, respectively. A simple adaptive window length algorithm is further developed to enhance the resolution of the proposed efficient realization of modified LPTFT and robust modified LPTFT. Experiments on the simulated time-varying signals including sinusoidal FM signals and hyperbolic FM signals validate the good performance of the proposed algorithms.

6.2 Recommendations for further research

6.2.1 Estimation of PPSs

More details about the properties of PTFT are to be further studied. For example, it will be interesting to exploit the capability of PTFT to discriminate multi-component PPSs, which is different along each of its dimension. More effort is also needed for analyzing the side lobes of PTFT, which are more complicated than those of DFT.

Another valuable research direction is to combine the PTFT and HAF for a good

compromise between computational complexity and statistical performances since the computation complexity of PTFT is still higher than that of HAF, especially for higher order PPSs. One possible combination is to use lower order HIM than that of HAF-based estimation method. Then, PTFT instead of FFT is computed for the lower order HIM to estimate more than one parameter at the same time.

Estimation of PPSs has a wide range of applications. This thesis has exploited two in Chapters 4 and 5. Other applications can be found in filtering and watermarking. The research on these applications is another future research direction.

6.2.2 PPI excision in spread spectrum communication systems

In this thesis, the proposed MDPFT-based excision method has been extended to the two-component PPI. More experiments are needed for PPI containing more than two components. The analytical performance analysis for mono-component PPI is expected to be extended to multi-component cases. The optimal number of components excised for multi-component PPI is also required to be derived corresponding to the estimation performance of the parameters of PPI.

Another possible extension of the proposed excision method is for the PPI with time-varying amplitudes. This kind of PPI often occurs in the channels where fading exists [17]. For the PPI with time-varying amplitudes, the performance of the proposed method can still work well if the variation of the amplitudes is moderate. On the other hand, when the amplitude is highly time varying, the performance

of the proposed method deteriorates significantly and new techniques are to be developed by introducing the estimated time-varying amplitude to the MDPFT.

The third direction of the proposed excision method is to deal with time-varying PPI where the parameters of PPI change with time. This scenario arises in the wireless communications where mobiles users are moving in most cases.

6.2.3 Time frequency analysis of time-varying signals

In this thesis, robust form of modified LPTFT has been proposed to deal with signals embedded in two kinds of impulse noises. More experiments are required to be done on the signals embedded in other kinds of impulse noises. Furthermore, the mixture of Gaussian and impulse noises may also arise in applications. In this case, neither robust FT nor FT can provide a good performance. It is expected to use L-estimation method introduced in [32]. The challenge of using L-estimation method is the decision of the parameters used for the median form filter [32].

Another possible direction is the potential applications of the proposed algorithm in speech analysis. The synthesis algorithm for the proposed modified LPTFT is also needed, which is important for other potential applications such as audio synthesis.

Author's Publication List

- [1] Guoan Bi and Yongmei Wei, "Split-Radix Algorithms for Arbitrary Order of Polynomial Time Frequency Transforms", Accepted by *IEEE Trans on Signal Processing*, 2006.
- [2] Yongmei Wei and Guoan Bi, "Efficient Analysis of Time-Varying Multi-Component Signals with Modified LPTFT", *EURASIP Journal on Applied Signal Processing*, Issue 8, 1261-1268, 2005.
- [3] Yongmei Wei and Guoan Bi, "Broadband Interference Suppression in DS-SS System with Modified Discrete Chirp Fourier Transform", *Signal Processing*, Volume 84, Issue 10, PP. 1749-1758, October 2004.
- [4] Guoan Bi and Yongmei Wei, "Fast Computation for Third-order Polynomial Time Frequency Transforms", *Electronic letters*, Volume 40, No. 5, PP. 349-351, March, 2004.
- [5] Yongmei Wei and Guoan Bi, "Narrowband Interference Suppression in DS-SS System with Modified Discrete Fourier Transform", Revised and Submitted to *IEEE Trans on Circuits And System*, August, 2005.

-
- [6] Yongmei Wei and Guoan Bi, "Fast Algorithm for Polynomial Time Frequency Transform", Submitted to *Signal Processing*. August, 2005.
 - [7] Yongmei Wei and Guoan Bi, Efficient Algorithm for Modified Local Polynomial Time Frequency Transform", Accepted by *ICASSP*, 2006.
 - [8] Yongmei Wei and Guoan Bi, "Broadband Interference Suppression in DS-SS System with Modified Discrete Chirp Fourier Transform", Proceedings of *Ninth International conference on communication systems*, PP. 640-644, Sept. 2004.
 - [9] Yongmei Wei and Guoan Bi, "Robust STFT with Adaptive Window Length and Rotation Direction", Proceedings of *Fourth International Conference on Information, Communications and Signal Processing*, Oct. 2003

Bibliography

- [1] T. J. Abatzoglou, “Fast maximum likelihood joint estimation of frequency and frequency rate,” in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1986, pp. 1409–1412.
- [2] K. Abed-Meraim and A. Beghdadi, “Multi-line fitting using polynomial phase transform and downsampling,” in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2001, pp. 1701–1704.
- [3] O. Akay and G. F. Boudreaux-Bartels, “Broadband interference excision in spread spectrum communication systems via fractional Fourier transform,” in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 1998, pp. 3265–3268.
- [4] M. G. Amin and K. D. Feng, “Short-time Fourier transforms using cascade filter structures,” *IEEE Trans. on Signal Processing*, vol. 42, no. 3, pp. 631–641, 1995.
- [5] M. G. Amin and K. D. Feng, “Short-time Fourier transforms using cascade filter structures,” *IEEE Transactions on Signal Processing*, vol. 42, no. 3, pp.

631–641, 1995.

- [6] M. G. Amin and G. R. Mandapati, “Nonstationary interference excision in spread spectrum communications using projection filtering methods,” in *The Thirty-Second Asilomar Conference on Signals, Systems and Computers*, 1998, pp. 827–831.
- [7] M. G. Amin, C. Wang, and A. R. Lindsey, “Optimum interference excision in spread spectrum communications using open-loop adaptive filters,” *IEEE Trans. on Signal Processing*, vol. 47, no. 7, pp. 1966–1976, 1999.
- [8] W. W. L. Au, *The Sonar of Dolphins*. Berlin, Germany: Springer-Verlag, 1993.
- [9] S. Barbarossa and V. Petrone, “Analysis of polynomial-phase signals by the integrated generalized ambiguity function,” *IEEE Trans. on Signal Processing*, vol. 45, no. 2, pp. 316–327, 1997.
- [10] S. Barbarossa and A. Scaglione, “Adaptive time-varying cancellation of wideband interferences in spread spectrum communications based on time-frequency distributions,” *IEEE Trans. on Signal Processing*, vol. 47, no. 4, pp. 957–965, 1999.
- [11] S. Barbarossa, A. Scaglione, and G. B. Giannakis, “Product high-order ambiguity function for multicomponent polynomial-phase signal modeling,” *IEEE Trans. on Signal Processing*, vol. 46, no. 3, pp. 691–708, 1998.

-
- [12] S. Barbarossa and R. Torti, “Chirped-ofdm for transmissions over time-varying channels with linear delay/doppler spreading,” in *Proc. of IEEE Signal Processing Workshop on Higher order Statistics*, 2001, pp. 2377–2380.
- [13] B. Barkat and B. Boashash, “Design of higher order polynomial time-frequency distributions,” *IEEE Trans. on Signal Processing*, vol. 47, no. 9, pp. 2608–2611, 1999.
- [14] B. Barkat and B. Boashash, “Instantaneous frequency estimation of polynomial FM signal using the peak of the PWVD: Statistical performance in the presence of additive Gaussian noise,” *IEEE Trans. on Signal Processing*, vol. 47, no. 9, pp. 2480–2490, 1999.
- [15] M. R. Bell and R. A. Grubbs, “Jem modeling and measurement for radar target identification,” *IEEE Trans. Aerosp. Electron. Syst*, vol. 29, no. 1, pp. 73–87, 1993.
- [16] A. J. Bemi and W. D. Gregg, “On the utility of chirp modulation for digital signaling,” *IEEE Trans. on Communications*, vol. 21, no. 6, pp. 748–751, 1973.
- [17] O. Besson, M. Ghogho, and A. Swami, “Parameter estimation for random amplitude chirp signals,” *IEEE Trans. on Signal Processing*, vol. 47, no. 12, pp. 3208–3219, 1999.
- [18] B. Boashash, “Estimating and interpreting the instantaneous frequency of a signal-part 2: Algorithms and applications,” *Proceedings of the IEEE*, vol. 80, no. 4, pp. 540–568, 1992.

-
- [19] B. Boashash and P. O'shea, "Polynomial Wigner-ville distributions and their relationship to time-varying higher order spectra," *IEEE Trans. on Signal Processing*, vol. 42, no. 1, pp. 216–220, 1994.
- [20] E. O. Brigham, *The fast Fourier transform*. Englewood Cliffs, NJ: Prentice Hall, 1974.
- [21] A. Bultan and A. N. Akansu, "A novel time-frequency exciser in spread spectrum communications for chirp-like interference," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 1998, pp. 3265–3268.
- [22] K. K. Clarke and D. T. Hess, *Communication Circuits: Analysis and Design*. Addison-Wesley: Wesley, 1971.
- [23] I. Cohen and I. Herlin, "A motion computation and interpretation framework for oceanographic satellite images," in *Proc. of International Symposium on Computer Vision*, 1995, pp. 13–18.
- [24] L. Cohen, *Time-Frequency Analysis*. Englewood Cliffs, NJ: Prentice Hall, 1995.
- [25] M. Dakovic, V. N. Ivanovic, and L. Stankovic, "On the S-method based instantaneous frequency estimation," in *Proc. of Seventh International Symposium on Signal Processing and Its Applications*, 2003, pp. 605–608.

-
- [26] S. Davidovici and E. G. Kanterakis, "Narrow-band interference rejection using real-time Fourier transforms," *IEEE Trans. on Communications*, vol. 37, no. 7, pp. 1292–1294, 1989.
- [27] P. J. Davis, *Interpolation and approximation*. New York: Dover, 1975.
- [28] Y. Ding and X. Qian, "Estimating sinusoidal parameters of musical tones based on global waveform fitting," in *Proc. of IEEE First Workshop on Multimedia Signal Processing*, 1997, pp. 95–100.
- [29] R. C. Dipietro, "An FFT based technique for suppressing narrow-band interference in PN spread-spectrum communication systems," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1989, pp. 1360–1363.
- [30] I. Djurovic, "Robust adaptive local polynomial Fourier transform," *IEEE Signal Processing letters*, vol. 11, no. 2, pp. 201–204, 2004.
- [31] I. Djurovic and L. Stankovic, "Realization of robust filters in the frequency domain," *IEEE Signal Processing letters*, vol. 9, no. 10, pp. 333–335, 2002.
- [32] I. Djurovic, L. Stankovic, and J. F. Bohme, "Robust L-estimation based forms of signal transforms and time-frequency representation," *IEEE Trans. on Signal Processing*, vol. 51, no. 7, pp. 1753–1761, 2003.

- [33] L. Durak and O. Arikan, “Short-time Fourier transform: two fundamental properties and optimal implementation,” *IEEE Trans. on Signal Processing*, vol. 51, no. 5, pp. 1231–1242, 2003.
- [34] L. G. Durand, J. Genest, Jr., and R. Guardo, “Modeling of the transfer function of the head-thorax acoustic systems in dogs,” *IEEE Trans. on Biomedical Engineering*, vol. 32, pp. 592–601, 1973.
- [35] M. F. Erden, M. A. Kutay, and H. M. Ozaktas, “Repeated filtering in consecutive fractional Fourier domains and its application to signal restoration,” *IEEE Trans. on Signal Processing*, vol. 47, no. 5, pp. 1458–1462, 1999.
- [36] M. J. Freeman, M. E. Dunham, and S. Qian, “Trans-ionospheric signal detection by time-scale representation,” in *Proc. IEEE UK Symp. Appl. Time-freq. Time-scale methods*, 1995, pp. 152–158.
- [37] J. Gevargiz and et al., “A comparison of weighted and nonweighted transform domain processing systems for narrowband interference excision,” in *Proc. of IEEE Military Communication Conference*, 1984, pp. 32.3.1–32.3.4.
- [38] F. Gini and G. B. Giannakis, “Hybrid FM-polynomial phase signal modeling: parameter estimation and Cramer-Rao bounds,” *IEEE Trans. on Signal Processing*, vol. 47, no. 2, pp. 363–377, 1999.
- [39] F. Gini, M. Luise, and R. Reggiannini, “Cramer-Rao bounds in the parametric estimation of fading radiotransmission channels,” *IEEE Trans. on Communications*, vol. 46, no. 10, pp. 1390–1398, 1998.

-
- [40] P. Guillemain and P. White, “Wavelet transforms for the analysis of dispersive systems,” in *Proc. IEEE Sympos. Time-Freq. Time-Scale Anal.*, 1996, pp. 429–432.
- [41] C. Gupta and A. Papandreou-Suppappola, “Wireless CDMA communications using time-varying signals,” in *Proc. of international symposium on signal processing and its applications*, 2001, pp. 242–245.
- [42] S. Haykin, *Adaptive filter theory*. Prentice Hall, 2001.
- [43] F. Hlawatsch, A. Papandreou-Suppappola, and G. F. Boudreaux-Bartels, “The hyperbolic class of quadratic time-frequency representations -part ii: subclasses, intersection with the affine and power classes, regularity and unitary,” *IEEE Trans. on Signal Processing*, vol. 45, no. 2, pp. 303–315, 1997.
- [44] F. Hlawatsch, A. Papandreou-Suppappola, and G. F. Boudreaux-Bartels, “The power classes-quadratic time-frequency representations with scale covariance and dispersive time-shift covariance,” *IEEE Trans. on Signal Processing*, vol. 47, no. 11, pp. 3067–3083, 1999.
- [45] S. R. Huang, R. M. Lerner, and K. J. Parker, “On estimating the amplitude of harmonic vibration from the doppler spectrum of reflected signals,” *J. Acoust. Soc. Amer.*, vol. 88, pp. 2702–2712, 1990.
- [46] P. J. Huber, “Robust regression: asymptotics, conjecture and monte carlo,” *Ann. Math. Stat.*, vol. 1, no. 5, pp. 799–821, 1973.

-
- [47] P. J. Huber, *Robust statistics*. New York: Wiley, 1981.
- [48] M. Z. Ikram, K. Abed-Meraim, and Y. Hua, "Fast quadratic phase transform for estimating the parameters of multicomponent chirp signals," *Digital Signal Processing*, vol. 7, no. 2, pp. 127–135, 1997.
- [49] M. Z. Ikram, K. Abed-Meraim, and Y. Hua, "Estimating the parameters of chirp signals: an iterative approach," *IEEE Trans. on Signal Processing*, vol. 46, no. 12, pp. 3436–3440, 1998.
- [50] M. Z. Ikram, A. Belouchrani, D. Gesbert, and K. Abed-Meraim, "Parametric estimation and suppression of non-stationary interference in DS-spread spectrum communications," in *The Thirty-Second Asilomar Conference on Signals, Systems and Computers*, 1998, pp. 1401–1405.
- [51] M. Z. Ikram and G. T. Zhou, "Estimation of multicomponent polynomial phase signals of mixed orders," *Signal Processing*, vol. 81, no. 11, pp. 2293–2308, 2001.
- [52] D. L. Jones and R. G. Baraniuk, "A simple scheme for adapting time-frequency representations," *IEEE Trans. on Signal Processing*, vol. 42, no. 12, pp. 3530–3535, 1994.
- [53] V. Katkovnik, "A new form of the Fourier transform for time-varying frequency estimation," *Signal Processing*, vol. 47, no. 2, pp. 187–200, 1995.

-
- [54] V. Katkovnik, "Local polynomial periodograms for signals with the time-varying frequency and amplitude," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1996, pp. 1399–1402.
- [55] V. Katkovnik, "Robust M-periodogram," *IEEE Trans. on Signal Processing*, vol. 46, no. 11, pp. 3104–3109, 1998.
- [56] V. Katkovnik, I. Djurovic, and L. Stankovic, "Median filter based realization of the robust time-frequency distributions," *Signal Processing*, vol. 81, no. 7, pp. 1771–1776, 2001.
- [57] S. M. Kay, *Fundamentals of Statistical Signal Processing-detection theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
- [58] S. M. Kay, *Fundamentals of Statistical Signal Processing-estimation theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
- [59] R. S. Kennedy, *Fading dispersive communication channels*. New York: Wiley, 1960.
- [60] R. R. Kurth, "Performance of FFT-based interference excision for hf communications," in *Proc. of IEEE Military Communication Conference*, 1990, pp. 21.5.1–21.5.5.
- [61] R. R. Kurth and et al., "Windowed FFT processing for narrow-band interference excision," in *Proc. of IEEE Military Communication Conference*, 1991, pp. C5.2.1–C5.2.5.

-
- [62] M. A. Kutay, H. M. Ozaktas, O. Arikan, and L. Onural, "Optimal filtering in fractional Fourier domains," *IEEE Trans. on Signal Processing*, vol. 45, no. 5, pp. 1129–1142, 1997.
- [63] H. K. Kwok and D. L. Jones, "Improved instantaneous frequency estimation using an adaptive short-time Fourier transform," *IEEE Trans. on Signal Processing*, vol. 48, no. 10, pp. 2964–2972, 2000.
- [64] S. R. Lach, M. G. Amin, and A. R. Lindsey, "Broadband interference excision for software-radio spread-spectrum communications using time-frequency distribution synthesis," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 4, pp. 704–714, 1999.
- [65] C. N. Li, G. R. Hu, and M. J. Liu, "Narrow-band interference excision in spread-spectrum systems using self-orthogonalizing transform-domain adaptive filters," *IEEE Journal on selected areas in Communications*, vol. 18, no. 3, pp. 403–406, 2000.
- [66] M. D. Macleod, "Fast nearly ML estimation of the parameters of real complex single tones or resolved multiple tones," *IEEE Trans. on Signal Processing*, vol. 46, no. 1, pp. 141–148, 1998.
- [67] M. Martone, "A multicarrier system based on the fractional Fourier transform for time-frequency selective channels," *IEEE Trans. on Signal Processing*, vol. 49, no. 6, pp. 1011–1020, 2001.

- [68] R. J. McAulay and T. F. Quatieri, "Pitch estimation and voicing detection based on a sinusoidal speech model," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1990, pp. 249–252.
- [69] M. J. Medly and G. J. Saulnier, "Narrow-band interference excision in spread spectrum systems using lapped transform," *IEEE Trans. on Signal Processing*, vol. 45, no. 11, pp. 1444–1455, 1997.
- [70] A. Mezer and B. Porat, "Parametric estimation algorithms for the modeling and restoration of geometrically deformed images," in *Proc. of Nineteenth Convention of Electrical and Electronics Engineers in Israel*, 1996, pp. 224–227.
- [71] D. Middleton, "Channel modeling and threshold signal processing in underwater acoustic: An analytical overview," *IEEE J. Oceanic Eng.*, vol. 12, no. 1, pp. 4–28, 1987.
- [72] L. B. Milstein, "Interferences rejection techniques in spread spectrum communications," *Proceedings of IEEE*, vol. 76, no. 6, pp. 657–671, 1988.
- [73] C. Nikias and M. Shao, *Signal Processing with α -stable distributions and applications*. New York: Wiley, 1995.
- [74] A. G. Orazco-Lugo, M. M. Lara, D. C. McLernon, and H. J. Muro-Lemus, "Multiple packet reception in wireless ad hoc networks using polynomial phase-modulating sequences," *IEEE Trans. on Signal Processing*, vol. 51, no. 8, pp. 2093–2110, 2003.

- [75] P. O'shea, "Fast parameter estimation algorithms for linear FM signals of frequency and frequency rate," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1994, pp. 17–20.
- [76] X. Ouyang and M. G. Amin, "Short-time Fourier transform receiver for non-stationary interference excision in direct sequence spread spectrum communications," *IEEE Trans. on Signal Processing*, vol. 49, no. 4, pp. 851–863, 2001.
- [77] H. M. Ozaktas, Z. Zalevsky, and M. A. Kutay, *The Fractional Fourier Transform: with Applications in Optics and Signal Processing*. Chichester: John Wiley and Sons Press, 2000.
- [78] A. Papandreou-Suppappola, "Sonar echo ranging using signals with nonlinear time-frequency characteristics," *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 393–396, 2004.
- [79] A. Papandreou-Suppappola and S.B.Suppappola, "Wideband weyl symbols for dispersive time-varying processing of systems and random signals," *IEEE Trans. on Signal Processing*, vol. 50, no. 5, pp. 1077–1090, 2002.
- [80] A. Papandreous-Suppappola, *Applications in time-frequency signal processing*. Boca Raton, FL: CRC Press, 2004.
- [81] A. Papandreous-Suppappola and G. F. Boudreaux-Bartels, "The effect of mismatching analysis signals and time-frequency representations," in *Proc. of*

the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis, 1996, pp. 224–227.

- [82] A. Papoulis, *Signal analysis*. New York: McGraw-Hill, 1977.
- [83] S. Peleg and B. Friedlander, “The discrete polynomial-phase transform,” *IEEE Trans. on Signal Processing*, vol. 43, no. 8, pp. 1901–1914, 1995.
- [84] S. Peleg and B. Friedlander, “Multicomponent signal analysis using the polynomial-phase transform,” *IEEE Trans. on Aerospace and Electronic systems*, vol. 32, no. 1, pp. 378–386, 1996.
- [85] S. Peleg and B. Porat, “Estimation and classification of polynomial-phase signals,” *IEEE Trans. on Information Theory*, vol. 37, no. 2, pp. 422–430, 1991.
- [86] S. Peleg and B. Porat, “The Cramer-Rao lower bound for signals with constant amplitude and polynomial phase,” *IEEE Trans. on Signal Processing*, vol. 39, no. 3, pp. 749–752, 1991.
- [87] H. Permuter and J. M. Francos, “A parametric approach for estimating the orientation of planar surfaces,” in *Proc. of International Conference on Image Processing*, 1998, pp. 181–185.
- [88] I. Pitas and A. N. Venetsanopoulos, *Nonlinear digital filters: principles and applications*. Boston, MA: Kluwer, 1990.

-
- [89] H. V. Poor and L. A. Rusch, "A promising multiplexing technology for cellular telecommunications: Narrowband interference suppression in spread spectrum CDMA," in *IEEE Personal Communications, third quarter*, 1994, pp. 14–27.
- [90] B. Porat and B. Friedlander, "Asymptotic statistical analysis of the high-order ambiguity function for parameter estimation of polynomial-phase signals," *IEEE Trans. on Information Theory*, vol. 42, no. 3, pp. 995–1001, 1996.
- [91] S. Qian, *Introduction to Time-Frequency and Wavelet Transforms*. New Jersey: Prentice Hall PTR, 2002.
- [92] S. Qian and D. P. Chen, *Joint Time-Frequency Analysis: methods and applications*. Upper Saddle River, NJ: Prentice Hall, 1996.
- [93] H. R. Raemer, *Statistical Communication Theory and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1969.
- [94] R. Rifkin, "Comments on 'narrow-band interference rejection using real-time Fourier transforms'," *IEEE Trans. on Communications*, vol. 39, no. 9, pp. 277–283, 1991.
- [95] R. Rifkin and E. R. Laderman, "Analysis of DFT-based frequency excision algorithms for direct-sequence spread-spectrum communications," in *Proc. of IEEE Military Communication Conference*, 1991, p. 422.
- [96] A. W. Rihaczek, *Principles of high-resolution radar*. Los Altos, CA: Peninsula publishing, 1985.

-
- [97] S. D. Sandberg, "Adapted demodulation for spread-spectrum receivers which employ transform-domain interference excision," *IEEE Trans. on Communications*, vol. 43, no. 9, pp. 2502–2510, 1995.
- [98] S. D. Sandberg, K. J. S. D. Marco, and M. A. Tzannes, "Some alternatives in transform-domain suppression of narrow-band interference for signal detection and demodulation," *IEEE Trans. on Communications*, vol. 43, no. 12, pp. 3025–3036, 1995.
- [99] A. Scaglione and S. Barbarossa, "On the spectral properties of polynomial-phase signals," *IEEE Signal Processing Letters*, vol. 5, no. 9, pp. 237–240, 1998.
- [100] M. I. Skolnik, *Radar Handbook*. New York: McGraw-Hill, 1970.
- [101] M. Soumekh, "Reconnaissance with ultra wideband UHF synthetic aperture radar," *IEEE Signal Processing Magazine*, vol. 12, no. 4, pp. 21–40, 1995.
- [102] L. Stankovic, "L-class time-frequency distributions," *IEEE Signal Processing letters*, vol. 3, no. 1, pp. 22–25, 1996.
- [103] L. Stankovic, "S-class of time-frequency distributions," *IEE Proceedings of Vision, Image and Signal Processing*, vol. 144, no. 2, pp. 57–64, 1997.
- [104] L. Stankovic, T. Alieva, and M. J. Bastiaans, "Time frequency signal analysis based on the windowed fractional Fourier transform," *Signal Processing*, vol. 83, no. 11, pp. 2459–2468, 2003.

-
- [105] L. Stankovic, S. Stankovic, and I. Djurovic, "An architecture for realization of the cross-terms free polynomial Wigner-ville distribution," in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1997, pp. 2053–2056.
- [106] S. Stankovic and I. Djurovic, "Motion parameter estimation by using time-frequency representations," *Electronic letters*, vol. 37, no. 24, pp. 1446–1448, 2001.
- [107] S. Stankovic and L. Stankovic, "An architecture for the realization of a system for time-frequency signal analysis," *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 44, no. 7, pp. 600–604, 1997.
- [108] V. Szekely, *Distributed RC networks*. Boca Raton, FL: CRC, 1995.
- [109] M. V. Tazebay and A. N. Akansu, "Adaptive subband transforms in time-frequency excisers for DSSS communications systems," *IEEE Trans. on Signal Processing*, vol. 43, no. 11, pp. 2776–2782, 1995.
- [110] M. V. Tazebay and A. N. Akansu, "A performance analysis of interferences excision in direct sequence spread spectrum communications," *IEEE Trans. on Signal Processing*, vol. 46, no. 9, pp. 2530–2534, 1998.
- [111] C. Wang and M. G. Amin, "Performance analysis of instantaneous frequency-based interference excision techniques in spread spectrum communications," *IEEE Trans. on Signal Processing*, vol. 46, no. 1, pp. 70–82, 1998.

-
- [112] Y. Wei and G. Bi, "Robust STFT with adaptive window length and rotation direction," in *Proc. of Fourth International Conference on Information, Communications and Signal Processing*, 2003.
- [113] D. P. Wiens, "Minimax- variance L- R-estimators location," *Can. J. Stat.*, vol. 18, no. 1, pp. 47–57, 1990.
- [114] D. P. Wiens, "Watermarking in the space/spatial-frequency domain using two-dimensional Radon-Wigner distribution," *IEEE Trans. on Image Processing*, vol. 10, no. 4, pp. 650–658, 2001.
- [115] J. E. Wilbur and R. J. McDonald, "Nonlinear analysis of cyclically correlated spectral spreading in modulated signals," *J. Acoust. Soc. Amer.*, vol. 92, no. 1, pp. 219–230, 1993.
- [116] W. J. Williams, "Reduced interference distributions: Biological applications and interpretations," *Proceedings of IEEE*, vol. 84, no. 9, pp. 1264–1280, 1996.
- [117] J. C. Wood and D. T. Barry, "Tomographic time-frequency analysis and its application toward time-varying filtering and adaptive kernel design for multicomponent linear-FM signals," *IEEE Trans. on Signal Processing*, vol. 42, no. 8, pp. 2094–2104, 1994.
- [118] X. G. Xia, "Discrete chirp-Fourier transform and its application to chirp rate estimation," *IEEE Trans. on Signal Processing*, vol. 48, no. 11, pp. 3122–3133, 2000.

-
- [119] X. G. Xia, "Dynamic range of the detectable parameters for polynomial phase signals using multiple-lag diversities in high-order ambiguity function," *IEEE Trans. on Information theory*, vol. 47, no. 4, pp. 1378–1384, 2001.
- [120] J. Xu, L. Durand, and P. Pibarot, "Extraction of the aortic and pulmonary components of the second heart sound using a nonlinear transient chirp signal model," *IEEE Trans. on Biomedical Engineering*, vol. 48, no. 3, pp. 277–283, 2001.
- [121] J. A. Young and J. S. Lehnert, "Analysis of DFT-based frequency excision algorithms for direct-sequence spread-spectrum communications," *IEEE Trans. on Communications*, vol. 46, no. 8, pp. 1076–1087, 1998.
- [122] J. A. Young and J. S. Lehnert, "Performance metrics for windows used in real-time DFT-based multiple-tone frequency excision," *IEEE Trans. on Signal Processing*, vol. 47, no. 3, pp. 800–812, 1999.
- [123] S. M. Zabin and H. Poor, "Efficient estimation of the class A parameters via the EM algorithm," *IEEE Trans. on information theory*, vol. 37, no. 1, pp. 60–72, 1991.
- [124] G. T. Zhou and Y. Wang, "Exploring lag diversity in the high-order ambiguity function for polynomial phase signals," *IEEE Signal Processing Letters*, vol. 4, no. 8, pp. 240–242, 1997.

Appendix A

Derivation of the output SNR of PTFT

It is assumed that the signal $s(n)$ is defined in (2.1) with constant amplitude A and the variance of $w(n)$ is σ^2 . When k and $\mathbf{l}_0$ match the parameters of the estimated PPS $s_0(n) = Ae^{j2\pi \sum_{m=0}^{M+1} a_m n^m}$ (i.e. $\hat{k}_0/N = a_1, \hat{l}_{i0}/N_i = a_{i+1}, 1 \leq i \leq M, \hat{l}_{i0}$ is one of the element of $(\hat{\mathbf{l}}_0)$), the mean of the PTFT at $(\hat{k}_0, (\hat{\mathbf{l}}_0))$ is

$$\begin{aligned} E[\text{PTFT}_s^M(\hat{k}_0, (\hat{\mathbf{l}}_0))] &= E\left[\frac{1}{N} \sum_{n=0}^{N-1} (s_0(n) + w(n)) e^{-j2\pi\phi_1(\hat{k}_0, (\hat{\mathbf{l}}_0), n)}\right] \\ &= A \end{aligned} \tag{A.1}$$

and the variance is

$$\begin{aligned}
 & \text{Var}(\text{PTFT}_s^M(\widehat{k}_0, \widehat{\mathbf{l}}_0)) \\
 &= E\left[\left|\frac{1}{N} \sum_{n=0}^{N-1} (s_0(n) + w(n)) e^{-j2\pi\phi_1(\widehat{k}_0, \widehat{\mathbf{l}}_0), n}\right|^2\right] - E^2[\text{PTFT}_s^M(\widehat{k}_0, \widehat{\mathbf{l}}_0)] \\
 &= (A^2 + \frac{1}{N}\sigma^2) - A^2 \\
 &= \frac{1}{N}\sigma^2
 \end{aligned} \tag{A.2}$$

Thus, the output SNR of PTFT

$$\begin{aligned}
 \text{SNR}_{out} &= \frac{E[\text{PTFT}_s^M(\widehat{k}_0, \widehat{\mathbf{l}}_0)]^2}{\text{Var}(\text{PTFT}_s^M(\widehat{k}_0, \widehat{\mathbf{l}}_0))} \\
 &= N \frac{A^2}{\sigma^2} = N \cdot \text{SNR}_{in}
 \end{aligned} \tag{A.3}$$

where $E[\cdot]$ and $\text{Var}(\cdot)$ are the mean and variance operators, respectively.

Appendix B

Proof of the quasi-periodic property

According to the definition in (2.9), the left side of (3.26) can be expressed as

$$\begin{aligned} & \text{PTFT}_x^{M+1}(k, l_1 + \frac{m_1 N_1}{2}, \dots, l_M + \frac{m_M N_M}{2}) \\ &= \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j2\pi(\phi_1(k, \mathbf{l}, n))} e^{-j\pi\phi_3(\mathbf{m}, n)} \end{aligned} \quad (\text{B.1})$$

where $\phi_3(\mathbf{m}, n) = m_1 n^2 + \dots + m_M n^{M+1}$, $\mathbf{m} = (m_1, \dots, m_M)$ and $m_i = 0$ or 1 .

Since $\phi_3(\mathbf{m}, n)$ is an integer, the value of $e^{-j\pi\phi_3(\mathbf{m}, n)}$ only depends on the parity of $\phi_3(\mathbf{m}, n)$.

Let us represent those non-zero terms, $m_i n^{i+1}$, in $\phi_3(\mathbf{m}, n)$, arranged in an ascending polynomial order, $m_{j(1)}, m_{j(2)}, \dots, m_{j(M_0)}$. When M_0 , which is the number of nonzero terms in $\phi_3(\mathbf{m}, n)$, is even, $\phi_3(\mathbf{m}, n)$ can be grouped into

$$n^{j(1)+1}(1 + n^{j(2)-j(1)}) + \dots + n^{j(M_0-1)+1}(1 + n^{j(M_0)-j(M_0-1)})$$

It can be easily seen that $\phi_3(\mathbf{m}, n)$ is always even because each of the terms, i.e., $n^{j(t)+1}(1 + n^{j(t+1)-j(t)})$ is always even. Therefore, $e^{-j\pi\phi_3(m_i, n)}$ is always equal to 1 and (B.1) becomes $\text{PTFT}_x^{M+1}(k, \mathbf{l})$.

When M_0 is odd, $\phi_3(m_i, n)$ can be expressed as

$$n^{j(1)+1}(1 + n^{j(2)-j(1)}) + \dots + n^{j(M_0-2)+1}(1 + n^{j(M_0-1)-j(M_0-2)}) + n^{j(M_0)+1}$$

Because the summation of all the terms (except the last one) in the above equation is always even, (B.1) can be expressed as

$$\begin{aligned} \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j2\pi(\phi_1(k, \mathbf{l}, n))} e^{-j\pi\phi_3(m_i, n)} &= \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j2\pi(\phi_1(k, \mathbf{l}, n))} e^{-j\pi n^{j(M_0)+1}} \\ &= \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j2\pi(\phi_1(k, \mathbf{l}, n))} e^{-j\pi n} \\ &= \text{PTFT}_x^{M+1}\left(k + \frac{N}{2}, l_1, l_2, \dots, l_M\right) \quad (\text{B.2}) \end{aligned}$$

Appendix C

Proof of the key property

With the definition of $y_q^{(l)}(n)$, it is straightforward to achieve:

$$\begin{aligned}
 y_q^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, \dots, l_M + \frac{a_M N_M}{2^{r+1}})}(2^r m + s) &= e^{-j2\pi \sum_{i=1}^M (\frac{a_i N_i}{2^{r+1} N_i} (2^r m + s)^{i+1})} y^{(l)}(2^r m + s) \\
 &= e^{-j2\pi \sum_{i=1}^M (\frac{a_i s^{i+1}}{2^{r+1}})} e^{-j\pi \sum_{i=1}^M (a_i (i+1)^1 s^i m)} y_q^{(l)}(2^r m + s) \\
 &= \Phi(r, s) e^{-j\pi \gamma m} y_q^{(l)}(2^r m + s)
 \end{aligned} \tag{C.1}$$

with $r = 1, \dots, q-1$. Let us consider two different cases as follows.

(a). When γ is even:

$$y_q^{(l_1 + \frac{a_1 N_1}{2^{r+1}}, \dots, l_M + \frac{a_M N_M}{2^{r+1}})}(2^r m + s) = \Phi(r, s) y_q^{(l)}(2^r m + s) \tag{C.2}$$

The first line on the right hand side of (3.31) can be achieved by applying the DFT in terms of m on sides of (C.2) and then replacing the right hand side by the DIT decomposition as shown in (3.5). It is noted that $\Phi(r, s)$ and $e^{-j\pi \gamma m}$ do not depend on the value of m .

(b). When γ is odd:

It should be noticed that the value of $e^{-j\pi\gamma m}$ is only influenced by the parity of m , not the values of m . As shown in (3.5) in the DIT decomposition, $y_r^{(1)}(k)$ is obtained by the combination of the subsequences associated with odd and even m , respectively. The second line on the right hand side of (3.31) can be achieved by applying the DFT in terms of m on both sides of (C.1) and then replacing the right hand side by the DIT decomposition as shown in (3.5).