TIGHT FRAME METHODS FOR DECONVOLUTION

CHAI ANWEI

(B.Sc., ECNU)

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF MATHEMATICS

NATIONAL UNIVERSITY OF SINGAPORE

2005

Acknowledgements

The two-year's study and research at the National University of Singapore is a valuable experience for me. Beyond the treasures I acquired from academic life, I have gained important knowledge of the society and interpersonal communication skills.

First, I would like to thank my supervisor, Professor Shen Zuowei. The discussion with Prof. Shen during my study for master's degree, especially the discussion in the last three months, helps me get a much clearer picture on wavelet theory and what a real research life looks like, which stimulates me to reconsider my future plan in academy. I would also like to thank Associate Professor Lin Ping for his valuable suggestions in and beyond study. Dr. Xia Tao, an active research scientist in CWAIP, also helped me a lot. I give my great thanks to him on the guidance of programming.

Second, I would like to express my thanks to my friends. I would like to thank Mr. Pan Suqi, my senior, from whom I got quick and accurate answers to my questions; thank Mr. Chen Libing, Mr. Dong Bin and Mr. Xu Yuhong for their helpful discussion during my research; thank all my other friends, the friendship with whom gives me a great memory of the time in Singapore.

In the end, I would like to give my greatest appreciation to my parents. I owe a lot to their consecutive support of my life in the past twenty-four years.

Anwei Chai

July 2005

Contents

Summary

Wavelet analysis has been proved to be a powerful tool in both theoretical and applied mathematics. It cuts up data or functions or operators into different frequency components and then studies each component with a resolution matched to its scale. In order to give such analysis more flexibility, the concept of frame was introduced into this area. Frame is a redundant system which preserves more useful information for analysis. In 1997, Ron and Shen [34] gave a systematical way for constructing tight affine frame system based on multiresolution analysis which makes the construction of tight frame painless. The application using tight frame system also becomes much easier.

However, the tight affine system is not shift invariant and hence restricts the application of the tight frame system in some aspects where shift invariance is a key requirement. To take over this matter, Ron and Shen put forward the concept of quasi-affine system. This system is shift invariant and satisfies the tight frame property if and only if its affine counterpart does. The first aim of this work is to give a systematical study of the quasi-affine tight frame system, to give the explicit formula of decomposition and reconstruction in such a system. We also connect this system with filter bank representation to give a discrete description of the quasi-affine tight frame system, which is desirable in application. Moreover, we give a necessary and sufficient condition on the initial low pass filter, from which a tight frame system can be constructed.

Next we use the properties of quasi-affine tight frame system to analyze deconvolution problem, which is the second aim of this work. Deconvolution problem is an important topic in inverse problems and arises in many applications, especially in those visual communication related areas. We start from the low pass filter of the given convolution equation to construct a corresponding tight frame system. The convolution equation is then interpreted in the quasi-affine system derived from the constructed tight frame using the idea of multiresolution analysis and its approximation. In such formulation, deconvolution becomes a process of filling missing wavelet coefficients. This approach is different from other available methods and give a new angle of view to deconvolution problem. We analyze the convergence of the algorithms derived from this new approach and the minimization properties of the solutions. Numerical simulation is conducted to show the effectiveness of the algorithms. Furthermore, as a direct application of the deconvolution algorithms, the connection with high resolution image reconstructions is briefly discussed.

Introduction

This thesis mainly discusses the tight wavelet frame system and the deconvolution problem. In this chapter, we briefly review these two areas and describe our research problems.

Wavelet Frame: Redundant Wavelet System

The development of wavelet theory cannot go without Fourier analysis. Fourier analysis is the core of pure and applied mathematics and the orthogonal property of the Fourier series plays a crucial role in various applications. Analogous to such classic theory, wavelet analysis inherits the key properties of its ancestor and at the same time, it is able to locate the information of functions or signals in both time and frequency domains. Since there exist real-time algorithms to obtain the coefficients of wavelet series and recover the original functions from such coefficients, wavelet analysis becomes popular in many applications and in some aspects, performs better than Fourier analysis. The examples of application can be found in signal processing (denoising, singularity detecting), image compression (JPEG 2000) and numerical analysis (numerical integration).

Although most applications of wavelets use orthonormal wavelet bases, especially the wavelet family constructed by Daubechies [18], we do have some types of applications which desire a redundant wavelet family. The typical examples are those applications related to signal denoising and image compression. Moreover, the orthonormal "restriction" makes it difficult to construct wavelets adaptive to specific application problems. These, together with other reasons, motivate people to find redundant wavelet family, in which more information is hoped to be kept.

As an easy way, the concept of frame is brought into wavelet analysis. Similar to the orthonormal one, the wavelet frame (also called affine frame) guarantees the perfect decomposition and reconstruction of the given signals or functions. We are particularly interested in the tight wavelet frames (tight affine frames), especially those constructed by the multiresolution analysis, since such wavelet frames guarantee the existence of fast decomposition and reconstruction algorithms. A systematic study of the construction of such frames can be found in [15, 20, 34]. The unitary extension principle (see Theorem 1.1) in [34] and more generally, the oblique extension principle in [15, 20] make the construction of tight wavelet frames painless once the low pass filter is given. Further, we can get symmetric or antisymmetric tight wavelets (framelets) from such constructions with only one multiresolution analysis, which is difficult (even impossible) for orthonormal wavelets.

The tight wavelet frame system is dilation invariant but not shift invariant. There is a sampling process in the data sequence. However, in some applications the size of data set needs to be stationary during the decomposition and reconstruction process. This leads to the Algorithm à Trous [33, Chapter 5] in which the sampling process is transferred to the filters and the size of data is kept. One the other hand, Ron and Shen introduced the concept of quasi-affine system [34] to overcome the difficulties in the study of construction of wavelet frames. Such system is then restudied in [16] to remove a minor assumption. But in neither paper is the decomposition and reconstruction algorithm in quasi tight wavelet frame (quasi-affine tight frame) discussed. Our first main goal of this work is to fully study the decomposition and reconstruction of a function in a quasi-system. As shown by our result, the decomposition and reconstruction algorithm in a quasi-affine tight frame system coincide with the Algorithm à Trous. We also give a discrete form of the decomposition and reconstruction process. This is important not only for our later analysis but also for real-time implementation.

Deconvolution: Ill-posed Inverse Problem

The second purpose of this work is to give a new approach to reconstruct a solution of the convolution equation

$$\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{b} + \boldsymbol{\epsilon} = \boldsymbol{c} \qquad (\clubsuit)$$

where $\boldsymbol{h}_0$ is a low pass filter (i.e. $\sum_{k\in\mathbb{Z}} \boldsymbol{h}_0[k] = 1$) and $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{\epsilon}$ are sequences in $\ell_2(\mathbb{Z})$ and $\boldsymbol{\epsilon}$ being the error term satisfying $\|\boldsymbol{\epsilon}\|_{\ell_2(\mathbb{Z})} \leq \varepsilon$.

There are many real life problems which can be modelled by a deconvolution process. For example, measurement devices and signal communication can introduce distortions and add noise to the original signal. Inverting the degradation is often modelled by a deconvolution process, i.e. a process of finding a solution in ($\clubsuit$). In fact, the deconvolution problem is a critical factor in many applications, especially visual-communication related applications including remote sensing, military imaging, surveillance, medical imaging and high resolution image reconstructions.

Solving equation ($\clubsuit$) is an inverting process, which is often numerically unstable and thus amplifies the noise considerably. Hence, an efficient process of noise removal must be built in the numerical algorithms. The earlier formulation of the problem was proposed in [37] using linear algorithm and in [26] and [36] applying the regularization idea to solve a system of linear equations the coefficient matrix of which is ill-conditioned. Since then, there are many papers devoted to this method in the literature. Because this approach is not the focus of this thesis, instead of a detailed count, we simply refer readers to [25] and [30] and the references there for a complete reference.

The focus of this thesis is to use wavelet (more generally, tight wavelet frame) to solve ($\clubsuit$). Recently, there are several papers on using wavelet methods to solve inverse problems, and in particular, deconvolution problems. One of the main ideas is to construct a wavelet or "wavelet inspired" basis that can almost diagonalize the given operator. The underlying solution has a sparse expansion with respect to the chosen basis. The Wavelet-Vaguelette decomposition proposed in [21], [23] and [24] and the deconvolution in mirror wavelet bases in [30] and [31] can be both viewed as examples of this strategy. Another approach is to apply Galerkin-type methods to inverse problems using an appropriate, but fixed wavelet basis (see e.g. [1] and [17]). Again, the idea there is that if the given operator has a sparse representation and the solution has a sparse expansion with respect to the wavelet basis, then the inversion is reduced approximately to the inversion of a truncated operator. A few new iterative thresholding algorithms which are different from other wavelet approaches and are developed simultaneously and independently are proposed in [8, 10, 11, 19, 21]. It only requires that the underlying solution has a sparse expansion with respect to a given system without any attempt to "almost diagonalize" the convolution operators.

The main idea of [19, 21] is to expand each iteration with respect to the chosen orthonormal basis for a given algorithm such as the Landweber method. Then a thresholding algorithm is applied to the coefficients of this expansion. The result is used to form the next iteration. The algorithm is shown to converge to the minimizer of certain cost functional.

In the studies of high resolution image reconstructions, the wavelet-based (in fact the frame-based) reconstruction algorithms are developed in [7, 8, 9], and later [10, 11] through the perfect reconstruction formula of a bi-frame or tight frame system which has $\boldsymbol{h}_0$ as its primary low pass filter. The algorithms approximate iteratively the coefficients of wavelet frame folded by the given low pass filter. By this approach, many available techniques developed in the wavelet literatures, such as wavelet-based denoising schemes, can be built in the iteration. When there are no displacement errors, the high resolution image reconstruction is exactly the deconvolution problem. Here, we extend the algorithms in the above mentioned papers to solve the equation ($\clubsuit$). Algorithm 5.1 is used in papers mentioned above, in particular in [8, 10]. This method has been extended to algorithms for high resolution image reconstructions with displacement errors in [10] and [11]. Algorithm 4.1 is given in [11] as one of the options which is motivated by the approaches taken by [19, 21]. Algorithms given in [12, 13] is based on Algorithm 4.2 where high resolution images are constructed from a series of video clips. The main ideas of all three algorithms are the same. i.e. an iterative algorithm combined with a denoising scheme applied to each iterate. The differences are different denoising schemes applied to different algorithms which in turn minimizes different cost functionals.

The convergence analysis of Algorithm 2.1 (the iteration without built-in denoising scheme) has already been established in [8] and [11]. However, the convergence of Algorithm 2.2, 2.3, 4.1, 4.2, 5.1 has not been discussed so far. The current work aims to build up a complete theory for these algorithms. We will first give a solid and complete formulation of reconstructions of a solution to equation (♣) in terms of multiresolution analysis and its associated frame system. Then the convergence of all algorithms will be given. A complete analysis of minimization properties, i.e. in which sense the solution derived from the algorithms attains its optimal property, will be given. Finally, the stability of the algorithms are also given, which shows that numerical solution approaches the exact solution when the noise level decreases to zero. As it has already been shown many times in the papers [8, 10, 11, 12, 13], algorithms are very numerically efficient, easy to implement and adaptive to many different applications such as high resolution image reconstructions with displacement errors (see e.g. [10] and [11]). In this thesis, a theoretical foundation of the underlying algorithms used in those papers is fully laid out.

The thesis is organized as follows: Chapter 1 gives the notation and proves basic results of tight frame system that will be used in this thesis. Chapter 2 devotes a formulation of the deconvolution problem in terms of multiresolution analysis and its associated wavelet frame. Algorithms will be derived from this formulation. Chapter 3 gives a complete analysis of the algorithms, including the convergence and minimization properties of the algorithms. Chapter 4 focuses on the finite dimensional data set, i.e. the data set has only finitely many entries. Algorithms 2.2 and 2.3 for infinite dimensional data set can be converted for this case by imposing proper boundary conditions. Since any numerical solution of deconvolution ultimately deals with finite dimensional data sets, such conversion is necessary. As we will see, in many cases, the discussion will be simpler and we are able to obtain better results for many cases. Since the numerical implementations and simulations are discussed in details in [8, 10, 11] and our focus here is to lay the foundation of the algorithms, we omit the detailed discussions of numerical implementations here. Finally, the deconvolution algorithms are connected with high resolution image reconstructions in Chapter 5, where the way to generalize algorithms for data in higher dimensional spaces is also included.

Chapter 1

Affine and Quasi-affine Tight Frame

This chapter focuses on the properties of affine and quasi-affine tight frame systems. We first review the definition and basic properties of affine tight frame system. After that, the quasi-affine tight frame system is introduced and the decomposition and reconstruction algorithm is given paralleled to the affine counterpart.

1.1 Tight Wavelet Frame

We give here a brief introduction of the tight wavelet frame and its quasi-affine counterpart. The decompositions and reconstructions for the affine tight frame system are known (e.g. [20]); however, the analysis of decomposition and reconstruction of quasi-affine systems is not systematically given. Since these results are crucial for our analysis, we introduce them here and give out the proofs in details. At the same time, we set the notations used in this thesis.

By the space $L_p(\mathbb{R})$, we mean that all the functions $f(x)$ satisfy

$$\|f\|_{L_p(\mathbb{R})} := \begin{cases} \left(\int_{\mathbb{R}} |f(x)|^p \,\mathrm{d}x\right)^{\frac{1}{p}} < \infty, & 1 \le p < \infty; \\ \operatorname{ess\,sup}_{x\in\mathbb{R}} |f(x)| < \infty, & p = \infty; \end{cases}$$

and $\ell_p(\mathbb{Z})$ is the set of all sequences defined on $\mathbb{Z}$ which satisfy that

$$\|\boldsymbol{h}\|_{\ell_p(\mathbb{Z})} := \begin{cases} \left(\sum_{k\in\mathbb{Z}} |\boldsymbol{h}[k]|^p\right)^{\frac{1}{p}} < \infty, & 1 \le p < \infty; \\ \sup_{k\in\mathbb{Z}} |\boldsymbol{h}[k]| < \infty, & p = \infty. \end{cases}$$

The Fourier transform of a function $f \in L_1(\mathbb{R})$ is defined as usual by:

$$\widehat{f}(\omega) = \int_{\mathbb{R}} f(x)\mathrm{e}^{-i\omega x}\,\mathrm{d}x, \quad \omega \in \mathbb{R},$$

and its inverse transform is

$$f(x) = \frac{1}{2\pi}\int_{\mathbb{R}} \widehat{f}(\omega)\mathrm{e}^{i\omega x}\,\mathrm{d}\omega, \quad x \in \mathbb{R}.$$

They can be extended to the functions in $L_2(\mathbb{R})$. Similarly, we can define the Fourier series for a sequence $\boldsymbol{h} \in \ell_2(\mathbb{Z})$ by

$$\widehat{\boldsymbol{h}}(\omega) = \sum_{k\in\mathbb{Z}} \boldsymbol{h}[k]e^{-ik\omega}, \quad \omega \in \mathbb{R}.$$

For any function $f \in L_2(\mathbb{R})$, the dyadic dilation operator D is defined by $Df(x) := \sqrt{2}f(2x)$ and the translation operator T is defined by $T_a f(x) := f(x-a)$ for $a \in \mathbb{R}$. Given $j \in \mathbb{Z}$, we have $T_a D^j = D^j T_{2^j a}$. Further, a space V is said to be integer-shift invariant if given any function $f \in V$, $T_j f \in V$ for $j \in \mathbb{Z}$.

A system $X \subset L_2(\mathbb{R})$ is called a tight frame of $L_2(\mathbb{R})$ if

$$\|f\|^2_{L_2(\mathbb{R})} = \sum_{g\in X} |\langle f, g\rangle|^2,$$

holds for all $f \in L_2(\mathbb{R})$, where $\langle\cdot,\cdot\rangle$ is the inner product in $L_2(\mathbb{R})$ and $\|\cdot\|_{L_2(\mathbb{R})} = \sqrt{\langle\cdot,\cdot\rangle}$. This is equivalent to

$$f = \sum_{g\in X} \langle f, g\rangle\, g, \;\; f \in L_2(\mathbb{R}).$$

It is clear that an orthonormal basis is a tight frame.

For given $\Psi := \{\psi_1, \ldots, \psi_r\} \subset L_2(\mathbb{R})$, define the affine system

$$X(\Psi) := \{\psi_{\ell,j,k} : 1 \le \ell \le r;\; j,k \in \mathbb{Z}\},$$

where $\psi_{\ell,j,k} = D^j T_k \psi_\ell = 2^{j/2}\psi_\ell(2^j \cdot -k)$. When $X(\Psi)$ forms an orthonormal basis of $L_2(\mathbb{R})$, then ψ_ℓ, $\ell = 1, \dots, r$, are called the orthonormal wavelets. When $X(\Psi)$ forms a tight frame of $L_2(\mathbb{R})$, then ψ_ℓ, $\ell = 1, \dots, r$, are called the tight framelets.

The tight framelets can be constructed by the unitary extension principle (UEP) given in [34], which uses the multiresolution analysis (MRA). The MRA starts from a refinable function ϕ. A compactly supported function ϕ is refinable if it satisfies a refinement equation

$$\phi(x) = 2\sum_{k\in\mathbb{Z}} \boldsymbol{h}_0[k]\phi(2x-k), \tag{1.1}$$

for some sequence $\boldsymbol{h}_0 \in \ell_2(\mathbb{Z})$. By the Fourier transform, the refinable equation (1.1) can be given as

$$\widehat{\phi}(\omega) = \widehat{\boldsymbol{h}_0}(\omega/2)\widehat{\phi}(\omega/2), \quad \text{a.e.} \quad \omega \in \mathbb{R}.$$

We call $\boldsymbol{h}_0$ the refinement mask of ϕ and $\widehat{\boldsymbol{h}_0}(\omega)$ the refinement symbol of ϕ.

For given finitely supported $\boldsymbol{h}_0$ with $\widehat{\boldsymbol{h}_0}(0) = 1$, the refinement equation (1.1) always has distribution solution which can be written in the Fourier domain as

$$\widehat{\phi}(\omega) = \prod_{j=1}^{\infty} \widehat{\boldsymbol{h}_0}(2^{-j}\omega), \quad \text{a.e.} \quad \omega \in \mathbb{R}.$$

In the following discussion, we require $\boldsymbol{h}_0$ being finitely supported. Then the corresponding refinable function ϕ satisfies that

$$\operatorname{ess\,sup}_{\omega\in\mathbb{R}} \sum_{k\in\mathbb{Z}} |\widehat{\phi}(\omega + 2k\pi)|^2 < \infty, \tag{1.2}$$

whenever $\phi \in L_2(\mathbb{R})$ (see [28]).

To build up a multiresolution analysis, we need the refinable function $\phi \in L_2(\mathbb{R})$. For a compactly supported refinable function $\phi \in L_2(\mathbb{R})$, let V_0 be the closed shift invariant space generated by $\{\phi(\cdot - k) : k \in \mathbb{Z}\}$ and $V_j := \{f(2^j\cdot) : f \in V_0\}$, $j \in \mathbb{Z}$. It is known that when $\phi \in L_2(\mathbb{R})$ is a compactly supported refinable function, then $\{V_j\}_{j\in\mathbb{Z}}$ forms a multiresolution analysis. Recall that a multiresolution analysis is a family of closed subspaces $\{V_j\}_{j\in\mathbb{Z}}$ of $L_2(\mathbb{R})$ that satisfies: (i) $V_j \subset V_{j+1}$, (ii) $\bigcup_j V_j$ is dense in $L_2(\mathbb{R})$, and (iii) $\bigcap_j V_j = \{0\}$ (see [6] and [29]).

For given MRA of nested spaces V_j, $j \in \mathbb{Z}$ with the underlying refinable function ϕ and the refinement mask $\boldsymbol{h}_0$, it is well known that (e.g. see [6]) for any $\psi \in V_1$, there exists a 2π periodic function ϑ, such that

$$\widehat{\psi}(2\cdot) = \vartheta\widehat{\phi}.$$

Let $\Psi := \{\psi_1, \ldots, \psi_r\} \subset V_1$, then

$$\widehat{\psi}_\ell(2\cdot) = \widehat{\boldsymbol{h}_\ell}\widehat{\phi}, \quad \ell = 1, \ldots, r, \tag{1.3}$$

where $\widehat{\boldsymbol{h}_1}, \ldots, \widehat{\boldsymbol{h}_r}$ are 2π periodic functions and are called framelet symbols. In the time domain, (1.3) can be written as

$$\psi_\ell(x) = 2\sum_{k\in\mathbb{Z}} \boldsymbol{h}_\ell[k]\phi(2x-k). \tag{1.4}$$

We call $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ framelet masks. We also call the refinement mask $\boldsymbol{h}_0$ the low pass filter and $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ the high pass filters of the system. The UEP gives conditions on $\{\widehat{\boldsymbol{h}_\ell}\}_{\ell=0}^r$, such that Ψ becomes a set of tight framelets with $X(\Psi)$ being a tight frame of $L_2(\mathbb{R})$.

Theorem 1.1 (Unitary Extension Principle, [34])**.** *Let $\phi \in L_2(\mathbb{R})$ be the refinable function with the refinement mask $\boldsymbol{h}_0$ satisfying $\widehat{\boldsymbol{h}_0}(0) = 1$ that generates an MRA $\{V_j\}_{j\in\mathbb{Z}}$. Let $(\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r)$ be a set of sequences with $\left(\widehat{\boldsymbol{h}_1}, \ldots, \widehat{\boldsymbol{h}_r}\right)$ being a set of 2π-periodic measurable functions in $L_\infty[0, 2\pi]$. If the equalities*

$$\sum_{\ell=0}^{r} |\widehat{\boldsymbol{h}_\ell}(\omega)|^2 = 1 \quad \textit{and} \quad \sum_{\ell=0}^{r} \widehat{\boldsymbol{h}_\ell}(\omega)\overline{\widehat{\boldsymbol{h}_\ell}(\omega+\pi)} = 0 \tag{1.5}$$

hold for almost all $\omega \in [-\pi, \pi]$, then the system $X(\Psi)$ where $\Psi = \{\psi_1, \ldots, \psi_r\}$ defined in (1.3) *by $(\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r)$ and ϕ forms a tight frame in $L_2(\mathbb{R})$.*

We will use (1.5) in terms of sequences $\boldsymbol{h}_0, \ldots, \boldsymbol{h}_r$. The first condition

$$\sum_{\ell=0}^{r} |\widehat{\boldsymbol{h}_\ell}(\omega)|^2 = 1$$

in terms of corresponding sequences is

$$\sum_{\ell=0}^{r} \sum_{k\in\mathbb{Z}} \overline{\boldsymbol{h}_\ell[k]}\boldsymbol{h}_\ell[k-p] = \delta_{0,p},\ p \in \mathbb{Z}, \tag{1.6}$$

where $\delta_{0,p}$ equals 1 when $p = 0$ and 0 otherwise. The second condition

$$\sum_{\ell=0}^{r} \widehat{\boldsymbol{h}_\ell}(\omega)\overline{\widehat{\boldsymbol{h}_\ell}(\omega+\pi)} = 0$$

can be written in terms of the sequences as

$$\sum_{\ell=0}^{r}\sum_{k\in\mathbb{Z}}(-1)^{k-p}\overline{\boldsymbol{h}_\ell[k]}\boldsymbol{h}_\ell[k-p] = 0, \quad p \in \mathbb{Z}. \tag{1.7}$$

With the UEP, the construction of tight framelets become painless. For example, one can construct tight framelets from spline easily. Next, we give some examples that will be used in our numerical simulations and the high resolution image reconstructions in [8, 10, 11].

Example 1.1. *Let* $\boldsymbol{h}_0 = [\frac{1}{4}, \frac{1}{2}, \frac{1}{4}]$ *be the refinement mask of the piecewise linear function* $\phi(x) = \max(1 - |x|, 0)$*. Define* $\boldsymbol{h}_1 = [-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}]$ *and* $\boldsymbol{h}_2 = [\frac{\sqrt{2}}{4}, 0, -\frac{\sqrt{2}}{4}]$*. Then* $\widehat{\boldsymbol{h}_0}$*,* $\widehat{\boldsymbol{h}_1}$ *and* $\widehat{\boldsymbol{h}_2}$ *satisfy* (1.5)*. Hence, the system* $X(\Psi)$ *where* $\Psi = \{\psi_1, \psi_2\}$ *defined in the way of* (1.3) *by using* $\boldsymbol{h}_1$*,* $\boldsymbol{h}_2$ *and* ϕ *is a tight frame of* $L_2(\mathbb{R})$*. This is the first example constructed via the UEP in [34].*

Example 1.2. *Let* $\boldsymbol{h}_0 = [\frac{1}{16}, \frac{1}{4}, \frac{3}{8}, \frac{1}{4}, \frac{1}{16}]$ *be the refinable mask of* ϕ*. Then* ϕ *is the piecewise cubic B-spline. Define* $\boldsymbol{h}_1$, $\boldsymbol{h}_2$, $\boldsymbol{h}_3$, $\boldsymbol{h}_4$ *as follows:*

$$\boldsymbol{h}_1 = [\tfrac{1}{16}, -\tfrac{1}{4}, \tfrac{3}{8}, -\tfrac{1}{4}, \tfrac{1}{16}], \quad \boldsymbol{h}_2 = [-\tfrac{1}{8}, \tfrac{1}{4}, 0, -\tfrac{1}{4}, \tfrac{1}{8}],$$
$$\boldsymbol{h}_3 = [\tfrac{\sqrt{6}}{16}, 0, -\tfrac{\sqrt{6}}{8}, 0, \tfrac{\sqrt{6}}{16}], \quad \boldsymbol{h}_4 = [-\tfrac{1}{8}, -\tfrac{1}{4}, 0, \tfrac{1}{4}, \tfrac{1}{8}].$$

Then $\widehat{\boldsymbol{h}_0}, \ldots, \widehat{\boldsymbol{h}_4}$ *satisfy* (1.5) *and hence the system* $X(\Psi)$ *where* $\Psi = \{\psi_\ell\}_{\ell=1}^4$*, defined in the way of* (1.3) *by* $\boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3, \boldsymbol{h}_4$ *and* ϕ *is a tight frame of* $L_2(\mathbb{R})$*. This is also first constructed in [34].*

The UEP construction is also true for the dilation other than 2. Following is an example constructed by the UEP with dilation 4. The low pass filter $\boldsymbol{h}_0$ is modelled as the convolution kernel for the case of 4×4 sensor arrays in high resolution image reconstruction. The tight frame system was constructed in [11] to use their framelet approach to reconstruct high resolution images.

Example 1.3. *Let* $\boldsymbol{h}_0 = [\frac{1}{8}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{8}]$ *be the refinable mask of* ϕ. *The other seven filters are given by:*

$$\boldsymbol{h}_1 = [\tfrac{1}{8}, 0, 0, 0, -\tfrac{1}{8}], \quad \boldsymbol{h}_2 = [\tfrac{1}{8}, 0, -\tfrac{1}{4}, 0, \tfrac{1}{8}], \quad \boldsymbol{h}_3 = [\tfrac{1}{8}, -\tfrac{1}{4}, 0, \tfrac{1}{4}, -\tfrac{1}{8}],$$
$$\boldsymbol{h}_4 = \tfrac{\sqrt{2}}{8}\cos(\tfrac{\pi}{8})[1, \sqrt{2}, 0, -\sqrt{2}, -1], \quad \boldsymbol{h}_5 = \tfrac{\sqrt{2}}{8}\sin(\tfrac{\pi}{8})[1, -\sqrt{2}, 0, \sqrt{2}, -1],$$
$$\boldsymbol{h}_6 = \tfrac{\sqrt{2}}{8}[\cos(\tfrac{\pi}{8}), -\sqrt{2}\sin(\tfrac{\pi}{8}), -2\sin(\tfrac{\pi}{8}), -\sqrt{2}\sin(\tfrac{\pi}{8}), \cos(\tfrac{\pi}{8})],$$
$$\boldsymbol{h}_7 = \tfrac{\sqrt{2}}{8}[\sin(\tfrac{\pi}{8}), -\sqrt{2}\cos(\tfrac{\pi}{8}), 2\cos(\tfrac{\pi}{8}), -\sqrt{2}\cos(\tfrac{\pi}{8}), \sin(\tfrac{\pi}{8})].$$

Then $\boldsymbol{h}_0, \boldsymbol{h}_1, \boldsymbol{h}_2, \boldsymbol{h}_3, \boldsymbol{h}_4, \boldsymbol{h}_5, \boldsymbol{h}_6, \boldsymbol{h}_7$ *satisfy that*

$$\sum_{\ell=1}^{7} \widehat{\boldsymbol{h}_\ell}(\omega)\overline{\widehat{\boldsymbol{h}_\ell}(\omega + \frac{2p\pi}{4})} = \delta_{0,p}, \quad p = 0, 1, 2, 3.$$

The UEP implies the corresponding $\{\psi_1, \psi_2, \psi_3, \psi_4, \psi_5, \psi_6, \psi_7\}$ *defined by*

$$\widehat{\psi}_\ell(\omega) = \widehat{\boldsymbol{h}_\ell}(\frac{\omega}{4})\widehat{\phi}(\frac{\omega}{4}), \quad \ell = 1, 2, 3, 4, 5, 6, 7,$$

are the tight framelets and the system $X(\Psi)$ *is a tight frame system of* $L_2(\mathbb{R})$.

The deconvolution process has to be formulated by quasi-affine systems that were first introduced in [34]. A quasi-affine system from level J is defined as

Definition 1.1. Let $\Psi = \{\psi_1, \ldots, \psi_r\}$ be a set of functions. A quasi-affine system from level J is defined as

$$X_J^q(\Psi) = \{\psi_{\ell,j,k}^q : \ 1 \le \ell \le r; j, k \in \mathbb{Z}\},$$

where $\psi_{\ell,j,k}^q$ is defined by

$$\psi_{\ell,j,k}^q := \begin{cases} D^j T_k \psi_\ell, & j \ge J; \\ 2^{\frac{j-J}{2}} T_{2^{-J}k} D^j \psi_\ell, & j < J. \end{cases}$$

The quasi-affine system is obtained by over sampling the affine system. More precisely, we over sample the affine system starting from level $J-1$ and downward to a 2^{-J}-shift invariant system. Hence, the whole quasi-affine system is a 2^{-J}-shift invariant system. The quasi-affine system from level 0 was first introduced in [34] to convert a non-shift invariant affine system to a shift invariant system. Further, it was shown in [34, Theorem

5.5] that the affine system $X(\Psi)$ is a tight frame of $L_2(\mathbb{R})$ if and only if $X_J^q(\Psi)$ is a tight frame of $L_2(\mathbb{R})$.

In our analysis, we use the quasi-interpolatory operator. Let $\{V_j\}$, $j \in \mathbb{Z}$ be a given MRA with underlying refinable function ϕ and $\Psi = \{\psi_1, \ldots, \psi_r\}$ be the set of corresponding tight framelets derived from the UEP. The quasi-interpolatory operator in the affine system $X(\Psi)$ generated by Ψ is defined, for $f \in L_2(\mathbb{R})$,

$$P_j : \ f \mapsto \sum_{k\in\mathbb{Z}} \langle f, \phi_{j,k}\rangle \phi_{j,k}.$$

It is clear that $P_j f \in V_j$. As shown in [20, Lemma 2.4], this quasi-interpolatory operator is the same as *truncated representation*

$$Q_j : \ f \mapsto \sum_{\ell=1}^{r} \sum_{j'<j, k\in\mathbb{Z}} \langle f, \psi_{\ell,j',k}\rangle \psi_{\ell,j',k}.$$

Furthermore, a standard framelet decomposition given in [20] says that

$$P_{j+1} f = P_j f + \sum_{\ell=1}^{r} \sum_{k\in\mathbb{Z}} \langle f, \psi_{\ell,j,k}\rangle \psi_{\ell,j,k} \quad \text{and} \quad P_j f = Q_j f. \tag{1.8}$$

When we consider the MRA based quasi-affine system $X_J^q(\Psi)$ generated by Ψ, the spaces V_j, $j < J$ in the former MRA for the affine system are replaced by $V_j^{q,J}$, $j < J$, for the quasi-affine system. Compared to the space V_j which is spanned by function $\phi_{j,k}$, each space $V_j^{q,J}$ is spanned by functions $\phi_{j,k}^q$, where $\phi_{j,k}^q$ is defined by

$$\phi_{j,k}^q := \begin{cases} D^j T_k \phi, & j \geq J; \\ 2^{\frac{j-J}{2}} T_{2^{-J}k} D^j \phi, & j < J. \end{cases}$$

The spaces $V_j^{q,J}$, $j < J$ are 2^{-J}-shift invariant. We can define the quasi-interpolatory operator $P_j^{q,J}$ and the truncated operator $Q_j^{q,J}$ for the quasi-affine system similarly:

$$P_j^{q,J} : \ f \mapsto \sum_{k\in\mathbb{Z}} \langle f, \phi_{j,k}^q\rangle \phi_{j,k}^q \tag{1.9}$$

and

$$Q_j^{q,J} : \ f \mapsto \sum_{\ell=1}^{r} \sum_{j'<j, k\in\mathbb{Z}} \langle f, \psi_{\ell,j',k}^q\rangle \psi_{\ell,j',k}^q. \tag{1.10}$$

The quasi-interpolatory operator $P_j^{q,J}$ maps $f \in L_2(\mathbb{R})$ to $V_j^{q,J}$. From the definition of $\phi_{j,k}^q$, we can see that $P_j^{q,J} = P_j$ when $j \geq J$ and these two operators are different only when $j < J$. Moreover, since for an arbitrary $f \in L_2(\mathbb{R})$ and $j < J$,

$$\begin{aligned} P_j^{q,J} f &= \sum_{k\in\mathbb{Z}} \langle f, \phi_{j,k}^q \rangle \phi_{j,k}^q \\ &= \sum_{k\in\mathbb{Z}} \langle f, 2^{\frac{j-J}{2}} T_{2^{-J}k} D^j \phi \rangle 2^{\frac{j-J}{2}} T_{2^{-J}k} D^j \phi \\ &= 2^{j-J-0} \sum_{k\in\mathbb{Z}} \langle f, D^J D^{-J} T_{2^{-J}k} D^j \phi \rangle D^J D^{-J} T_{2^{-J}k} D^j \phi \\ &= D^J \sum_{k\in\mathbb{Z}} \langle D^{-J} f, 2^{\frac{j-J-0}{2}} T_k D^{j-J} \phi \rangle 2^{\frac{j-J-0}{2}} T_k D^{j-J} \phi \\ &= D^J P_{j-J}^{q,0} D^{-J} f, \end{aligned}$$

one only needs to understand the case $J = 0$. In this case we simplify our notation by setting

$$P_j^q := P_j^{q,0}, \quad Q_j^q := Q_j^{q,0} \tag{1.11}$$

for the quasi-interpolatory operators and

$$V_j^q := V_j^{q,0}$$

for the nested spaces. From now on, we only give the properties for P_j^q and corresponding spaces V_j^q and the associated quasi-affine system $X^q(\Psi) := X_0^q(\Psi)$. The corresponding results for the over sampling rate of $2^{-J}\mathbb{Z}$ can be obtained similarly. Thus we only consider the case of quasi-affine system $X^q(\Psi)$.

From the following result we can see that for operator P_j^q, a decomposition and reconstruction formula similar to (1.8) holds in quasi-affine tight frame system.

Lemma 1.1. *Let $X(\Psi)$, where the framelets $\Psi = \{\psi_1, \ldots, \psi_r\}$, be the affine tight frame system obtained from $\boldsymbol{h}_0$ and ϕ via the UEP and $X^q(\Psi)$ be the quasi-affine frame derived from $X(\Psi)$. Then we have*

$$P_{j+1}^q f = P_j^q f + \sum_{\ell=1}^{r} \sum_{k\in\mathbb{Z}} \langle f, \psi_{\ell,j,k}^q \rangle \psi_{\ell,j,k}^q, \quad f \in L_2(\mathbb{R}). \tag{1.12}$$

Proof. When $j \geq 0$, we have $\phi^q_{j,k} = D^j T_k \phi = \phi_{j,k}$ and $\psi^q_{j,k} = D^j T_k \psi = \psi_{j,k}$, which imply that

$$P_j f = \sum_{k \in \mathbb{Z}} \langle f, \phi_{j,k} \rangle \phi_{j,k} = \sum_{k \in \mathbb{Z}} \langle f, \phi^q_{j,k} \rangle \phi^q_{j,k} = P^q_j f,$$

and

$$\sum_{\ell=1}^{r} \sum_{k \in \mathbb{Z}} \langle f, \psi^q_{\ell,j,k} \rangle = \sum_{\ell=1}^{r} \sum_{k \in \mathbb{Z}} \langle f, \psi_{\ell,j,k} \rangle.$$

Since in [20, Lemma 2.4], it has already been proved that

$$P_{j+1} f = P_j f + \sum_{\ell=1}^{r} \sum_{k \in \mathbb{Z}} \langle f, \psi_{\ell,j,k} \rangle \psi_{\ell,j,k},$$

we have

$$P^q_{j+1} f = P_{j+1} f = P_j f + \sum_{\ell=1}^{r} \sum_{k \in \mathbb{Z}} \langle f, \psi_{\ell,j,k} \rangle \psi_{\ell,j,k} = P^q_j f + \sum_{\ell=1}^{r} \sum_{k \in \mathbb{Z}} \langle f, \psi^q_{\ell,j,k} \rangle \psi^q_{\ell,j,k},$$

i.e. the identity (1.12) holds when $j \geq 0$. Next we show (1.12) also holds for $j < 0$. We first denote ϕ as ψ_0.

By the definitions of refinable equation (1.1) and framelet (1.4), one obtains that for $\ell = 0, 1, \ldots, r$,

$$\psi_\ell(x) = 2 \sum_{k \in \mathbb{Z}} \boldsymbol{h}_\ell[k] \phi(2x - k).$$

This leads to

$$\begin{aligned}
\psi^q_{\ell,j,k} &= 2^j T_k \psi_\ell(2^j \cdot) \\
&= 2^{j+1} T_k \Big(\sum_{k' \in \mathbb{Z}} \boldsymbol{h}_\ell[k'] \psi_0(2^{j+1} \cdot - k') \Big) \\
&= \sum_{k' \in \mathbb{Z}} \boldsymbol{h}_\ell[k'] 2^{j+1} \psi_0(2^{j+1}(\cdot - k - 2^{-j-1} k')) \\
&= \sum_{k' \in 2^{-j-1}\mathbb{Z}} \boldsymbol{h}_\ell[2^{j+1} k'] 2^{j+1} \psi_0(2^{j+1}(\cdot - k - k')).
\end{aligned}$$

We define the dilated sequence $\boldsymbol{h}_{\ell,j}$ by

$$\boldsymbol{h}_{\ell,j}[k] = \begin{cases} \boldsymbol{h}_\ell[2^{j+1} k], & k \in 2^{-j-1}\mathbb{Z}; \\ 0, & k \notin 2^{-j-1}\mathbb{Z}. \end{cases} \tag{1.13}$$

Such sequence $\boldsymbol{h}_{\ell,j}$ is obtained inductively by inserting 0 between every two entries in $\boldsymbol{h}_{\ell,j+1}$ with $\boldsymbol{h}_{\ell,-1} = \boldsymbol{h}_\ell$. With the dilated sequence, we have

$$\psi^q_{\ell,j,k} = \sum_{k' \in \mathbb{Z}} \boldsymbol{h}_{\ell,j}[k'] \psi^q_{0,j+1,k+k'},$$

and moreover, the right hand side of (1.12) can be written as follows:

$$\begin{aligned}
&\sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \langle f, \psi^q_{\ell,j,k} \rangle \psi^q_{\ell,j,k} \\
&\qquad = \sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \Big(\sum_{k' \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k']} \langle f, \psi^q_{0,j+1,k'+k} \rangle \Big) \Big(\sum_{k'' \in \mathbb{Z}} \boldsymbol{h}_{\ell,j}[k''] \psi^q_{0,j+1,k''+k} \Big) \\
&\qquad = \sum_{k' \in \mathbb{Z}} \sum_{k'' \in \mathbb{Z}} \Big(\sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k + k'' - k'] \Big) \langle f, \psi^q_{0,j+1,k'} \rangle \psi^q_{0,j+1,k''}.
\end{aligned}$$

Next, we check that $\sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k + k'' - k'] = \delta_{0,k'-k''}$. When $k - k'' \in 2^{-j-1}\mathbb{Z}$, there exists $p \in \mathbb{Z}$ such that $k' - k'' = 2^{-j-1}p$ and we have

$$\begin{aligned}
\sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k + k'' - k'] &= \sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k - 2^{-j-1}p] \\
&= \sum_{\ell=0}^{r} \sum_{k \in 2^{-j-1}\mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k - 2^{-j-1}p] \\
&= \sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell}[k]} \boldsymbol{h}_{\ell}[k - p] = \delta_{0,p}.
\end{aligned}$$

The last identity follows by (1.6). The sum is nonzero if and only if $p = 0$, which is exactly $k' = k''$. When $k' - k'' \notin 2^{-j-1}\mathbb{Z}$, there exist $p_1, p_2 \in \mathbb{Z}$ and $p_2 \notin 2^{-j-1}\mathbb{Z}$ such that $k' - k'' = 2^{-j-1}p_1 + p_2$. Then we have

$$\begin{aligned}
\sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k + k'' - k'] &= \sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k - 2^{-j-1}p_1 - p_2] \\
&= \sum_{\ell=0}^{r} \sum_{k \in 2^{-j-1}\mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k - 2^{-j-1}p_1 - p_2].
\end{aligned}$$

Since $k - 2^{-j-1}p_1 - p_2 \notin 2^{-j-1}\mathbb{Z}$ when $k \in 2^{-j-1}\mathbb{Z}$, we have $\boldsymbol{h}_{\ell,j}[k - 2^{-j-1}p_1 - p_2] = 0$ for any $k \in 2^{-j-1}\mathbb{Z}$ and hence the last identity is equal to 0. In conclusion, for the dilated filters $\boldsymbol{h}_{0,j}, \boldsymbol{h}_{1,j}, \ldots, \boldsymbol{h}_{r,j}$, we still have a similar result as (1.6)

$$\sum_{\ell=0}^{r} \sum_{k \in \mathbb{Z}} \overline{\boldsymbol{h}_{\ell,j}[k]} \boldsymbol{h}_{\ell,j}[k - p] = \delta_{0,p}, \quad p \in \mathbb{Z}. \tag{1.14}$$

Thus we have

$$\sum_{\ell=0}^{r}\sum_{k\in\mathbb{Z}}\langle f,\psi^q_{\ell,j,k}\rangle\psi^q_{\ell,j,k}=\sum_{k\in\mathbb{Z}}\langle f,\psi^q_{0,j+1,k}\rangle\psi^q_{0,j+1,k}=P^q_{j+1}f.$$

This is the identity we need to prove when $j<0$. In all, identity (1.12) holds for any $j\in\mathbb{Z}$. □

We note here that in the proof of identity (1.8) for the affine system, one needs both conditions in (1.5); while in the proof of identity (1.12), when the quasi-affine system is used, one only needs (1.6) if $j<0$. More general, it was proven in [20, Lemma 2.4] that the identity $P_jf=Q_jf$ holds for all $f\in L_2(\mathbb{R})$. Next result shows that a similar result also holds for the quasi-affine systems.

Proposition 1.1. *Let $X(\Psi)$ with $\Psi=\{\psi_1,\dots,\psi_r\}$ be the affine tight frame system obtained from $\boldsymbol{h}_0$ and ϕ via the UEP and $X^q(\Psi)$ be the corresponding quasi-affine frame. Then we have $P^q_jf=Q^q_jf$ for all $f\in L_2(\mathbb{R})$.*

Proof. First we consider the case $j\ge 0$. In this case, since $\phi^q_{j,k}=D^jT_k\phi=\phi_{j,k}$, we have

$$P_jf=\sum_{k\in\mathbb{Z}}\langle f,\phi_{j,k}\rangle\phi_{j,k}=\sum_{k\in\mathbb{Z}}\langle f,\phi^q_{j,k}\rangle\phi^q_{j,k}=P^q_jf.$$

Next, we show that $Q_jf=Q^q_jf$ when $j\ge 0$. Since $X(\Psi)$ is a tight frame, $X^q(\Psi)$ is also a tight frame by [34, Theorem 5.5]. On the other hand, $j\ge 0$ implies $\psi^q_{\ell,j,k}=D^jT_k\psi_\ell=\psi_{\ell,j,k}$. Thus we have

$$\begin{aligned}\sum_{\ell=1}^{r}\sum_{j<0}\sum_{k\in\mathbb{Z}}\langle f,\psi_{\ell,j,k}\rangle\psi_{\ell,j,k}&=f-\sum_{\ell=1}^{r}\sum_{j\ge 0}\sum_{k\in\mathbb{Z}}\langle f,\psi_{\ell,j,k}\rangle\psi_{\ell,j,k}\\&=\sum_{\ell=1}^{r}\sum_{j<0}\sum_{k\in\mathbb{Z}}\langle f,\psi^q_{\ell,j,k}\rangle\psi^q_{\ell,j,k}.\end{aligned}$$

Hence, when $j\ge 0$,

$$\begin{aligned}Q^q_jf&=\sum_{\ell=1}^{r}\sum_{j'<0,k\in\mathbb{Z}}\langle f,\psi^q_{\ell,j',k}\rangle\psi^q_{\ell,j',k}+\sum_{\ell=1}^{r}\sum_{j'=0}^{j}\sum_{k\in\mathbb{Z}}\langle f,\psi_{\ell,j',k}\rangle\psi_{\ell,j',k}\\&=\sum_{\ell=1}^{r}\sum_{j'<0,k\in\mathbb{Z}}\langle f,\psi_{\ell,j',k}\rangle\psi_{\ell,j',k}+\sum_{\ell=1}^{r}\sum_{j'=0}^{j}\sum_{k\in\mathbb{Z}}\langle f,\psi_{\ell,j',k}\rangle\psi_{\ell,j',k}\\&=Q_jf.\end{aligned}$$

Since $P_j f = Q_j f$ by [20, Lemma 2.4], we have $P_j^q f = Q_j^q f$ for $j \geq 0$.

Next we show that $P_j^q f = Q_j^q f$ holds when $j < 0$. Applying Lemma 1.1 inductively for any $f \in L_2(\mathbb{R})$ and $j < 0$, we have

$$P_j^q f = P_{j''}^q f + \sum_{\ell=1}^{r} \sum_{j'=j''}^{j} \sum_{k\in\mathbb{Z}} \langle f, \psi_{\ell,j',k}^q \rangle \psi_{\ell,j',k}^q. \tag{1.15}$$

Thus the proof of $P_j^q f = Q_j^q f$ is transferred to the proof of $P_{j''}^q f \to 0$ as $j'' \to -\infty$. The proof below is essentially the same as that of [29, Theorem 2.2].

Since $\boldsymbol{h}_0$ is finitely supported, the refinable function ϕ derived from $\boldsymbol{h}_0$ satisfies (1.2), which implies that the integer shifts of $\phi_{j'',0}^q$ is a Bessel sequence. Because

$$P_{j''}^q f \quad = \quad \sum_{k\in\mathbb{Z}} \langle f, \phi_{j'',k}^q \rangle \phi_{j'',k}^q,$$

the norm of $P_{j''}^q f$ satisfies

$$\|P_{j''}^q f\|_{L_2(\mathbb{R})}^2 \leq C \sum_{k\in\mathbb{Z}} |\langle f, \phi_{j'',k}^q \rangle|^2, \tag{1.16}$$

where the constant C is independent of j''. Based on the result in approximation theory, we only need to check the value of $\|P_{j''}^q f\|_{L_2(\mathbb{R})}$ when f is supported on an interval $[-R, R]$ for some $R > 0$. By the Cauchy-Schwartz inequality we have for $j'' < 0$ and $|j''|$ sufficiently large,

$$\|P_{j''}^q f\|_{L_2(\mathbb{R})}^2 \leq C \|f\|_{L_2}^2 \int_{E_{j''}} |\phi(x)|^2 \,\mathrm{d}x, \tag{1.17}$$

where

$$E_{j''} = \bigcup_{k\in\mathbb{Z}} \Big(k + 2^{j''}[-R, R] \Big).$$

Now $P_{j''}^q f \to 0$ follows by letting $j'' \to -\infty$ in (1.17). Then (1.15) becomes

$$P_j^q f = \sum_{\ell=1}^{r} \sum_{j'<j} \sum_{k\in\mathbb{Z}} \langle f, \psi_{\ell,j',k}^q \rangle \psi_{\ell,j',k}^q = Q_j^q f.$$

Thus we complete our proof of $P_j^q f = Q_j^q f$ for any $j \in \mathbb{Z}$. □

1.2 Discrete Form

The identity (1.12) essentially gives the decomposition and reconstruction of a function in quasi-affine tight frame systems. In the implementation, one needs a complete discrete form of the decomposition and reconstruction and we give such form below.

We introduce the Toeplitz matrix to describe the discrete form of the decomposition and reconstruction procedure. Given a sequence $\boldsymbol{h}_0 = \{\boldsymbol{h}_0[k]\}_{k\in\mathbb{Z}}$, the Toeplitz matrix generated by $\boldsymbol{h}_0$ is a matrix satisfying

$$H_0 = (H_0[l,k]) = (\boldsymbol{h}_0[l-k]),$$

i.e. the (l,k)th entry in H_0 is fully determined by the $(l-k)$th entry in $\boldsymbol{h}_0$. The Toeplitz matrix is also called the convolution matrix since it can be viewed as the matrix representation of linear time invariant filter which can be written as a convolution. Hence the convolution of two sequences can be expressed in terms of matrix vector multiplication, i.e.

$$\boldsymbol{h}_0 * \boldsymbol{v} = H_0\boldsymbol{v}. \tag{1.18}$$

In the following, we will denote the Toeplitz matrix generated from $\boldsymbol{h}_0$ by

$$H_0 = \text{Toeplitz}(\boldsymbol{h}_0).$$

Let the infinite dimensional matrix $H_\ell = \text{Toeplitz}(\boldsymbol{h}_\ell)$ be the Toeplitz matrix generated from the sequence $\boldsymbol{h}_\ell$ for $\ell = 1,\dots,r$. Using the matrix notation, the UEP condition (1.6) can be written as

$$H_0^*H_0 + H_1^*H_1 + \cdots + H_r^*H_r = I, \tag{1.19}$$

where I is the identity operator. To write the decomposition and reconstruction algorithms in convolution form, the filters used in the decomposition below the 0th level need to be dilated. In level $j < 0$, the dilated filter is denoted by $\boldsymbol{h}_{\ell,j}$, which is defined by (also see (1.13))

$$\boldsymbol{h}_{\ell,j}[k] = \begin{cases} \boldsymbol{h}_\ell[2^{j+1}k], & k \in 2^{-j-1}\mathbb{Z}; \\ 0, & k \notin 2^{-j-1}\mathbb{Z}. \end{cases}$$

The corresponding Toeplitz matrix is $H_{\ell,j} = \text{Toeplitz}(\boldsymbol{h}_{\ell,j})$. By the definition of $\boldsymbol{h}_{\ell,j}$, we have $\widehat{\boldsymbol{h}_{\ell,j}} = \widehat{\boldsymbol{h}_\ell}(2^{-j-1}\cdot)$ and hence $|\widehat{\boldsymbol{h}_{\ell,j}}| \leq 1$ a.e. $\omega \in \mathbb{R}$. Moreover, as a byproduct in the proof of Lemma 1.1, we have a condition similar to (1.19) for dilated filters $\boldsymbol{h}_{0,j}, \ldots, \boldsymbol{h}_{r,j}$, $j < 0$:

$$H_{0,j}^* H_{0,j} + H_{1,j}^* H_{1,j} + \cdots + H_{r,j}^* H_{r,j} = I. \tag{1.20}$$

We can see that when $j = -1$, (1.19) and (1.20) are the same.

The discrete forms of decomposition and reconstruction from level j_1 to level j_2, where $j_1, j_2 \geq 0$, are the same as those in the affine system, which are given in [20]. We only consider the discrete form of decomposition and reconstruction from level j_1 to level j_2, where $j_1, j_2 < 0$. For a function $f \in L_2(\mathbb{R})$, we decompose f in $X^q(\Psi)$ and collect the coefficients in each level $j < 0$ to form an infinite column vector

$$\boldsymbol{v}_{\ell,j} := [\ldots, \langle f, \psi_{\ell,j,k}^q \rangle, \ldots]^t,$$

where $\psi_0^q := \phi^q$ and $[\cdots]^t$ means transpose of a row vector to a column form. Set the Toeplitz block matrix

$$\mathcal{H}_j := [H_{0,j}, H_{1,j}, \ldots, H_{r,j}]^t .$$

With this, condition (1.20) implies $\mathcal{H}_j^* \mathcal{H}_j = I$. The decomposition process (1.12) can be written in the matrix form as:

$$\boldsymbol{v}_{\ell,j} = H_{\ell,j} \boldsymbol{v}_{0,j+1}, \quad \ell = 0, \ldots, r,$$

or

$$[\boldsymbol{v}_{0,j}, \ldots, \boldsymbol{v}_{r,j}]^t = \mathcal{H}_j \boldsymbol{v}_{0,j+1}. \tag{1.21}$$

Because of (1.20), the reconstruction process of Lemma 1.1 can be interpreted in the discrete form as

$$\begin{aligned}
\boldsymbol{v}_{0,j+1} &= \mathcal{H}_j^* \mathcal{H}_j \boldsymbol{v}_{0,j+1} \\
&= H_{0,j}^* H_{0,j} \boldsymbol{v}_{0,j+1} + H_{1,j}^* H_{1,j} \boldsymbol{v}_{0,j+1} + \cdots + H_{r,j}^* H_{r,j} \boldsymbol{v}_{0,j+1} \\
&= H_{0,j}^* \boldsymbol{v}_{0,j} + H_{1,j}^* \boldsymbol{v}_{1,j} + \cdots + H_{r,j}^* \boldsymbol{v}_{r,j}.
\end{aligned} \tag{1.22}$$

The identities (1.21) and (1.22) together give the equivalent discrete representation of (1.12).

The above discussion essentially is one level decomposition and reconstruction. Next, we introduce the notation of several to infinite levels decomposition and reconstruction. For any sequence $\boldsymbol{v}$, it is decomposed by $\mathcal{H}_{-1}\boldsymbol{v}$ first, then the low frequency component $H_0\boldsymbol{v}$ is further decomposed by the same procedure. The same process goes inductively. To describe this discrete process, we define the decomposition operator $\mathcal{A}_J$, $J<0$ and $\mathcal{A}$. They are composed of matrix block like $H_{\ell,j}\prod_{j'=j}^{-1} H_{0,j'}$ where $\prod_{j'=j}^{-1} H_{0,j'}$ is the composition of $|j|$ Toeplitz matrices $H_{0,j'}$, $j \le j' \le -1$, satisfying that for any sequence $\boldsymbol{v}\in\ell_2(\mathbb{Z})$,

$$\prod_{j'=j}^{-1} H_{0,j'}\boldsymbol{v} = H_{0,j}H_{0,j+1}\cdots H_{0,-1}\boldsymbol{v}.$$

The decomposition operator $\mathcal{A}_J$ is a (rectangular) block matrix defined as:

$$[(\prod_{j=J}^{-1} H_{0,j}), (H_{1,J}\prod_{j=J+1}^{-1} H_{0,j}), \ldots, (H_{r,J}\prod_{j=J+1}^{-1} H_{0,j}), \ldots, H_{1,-1}, \ldots, H_{r,-1}]^t \tag{1.23}$$

and $\mathcal{A}$ is defined as

$$\begin{aligned}[\ldots, (H_{1,J-1}\prod_{j=J-1}^{-1} H_{0,j}), \ldots, (H_{r,J-1}\prod_{j=J-1}^{-1} H_{0,j}),\ (H_{1,J}\prod_{j=J}^{-1} H_{0,j}), \ldots, (H_{r,J}\prod_{j=J}^{-1} H_{0,j}),\\ (H_{1,J+1}\prod_{j=J+1}^{-1} H_{0,j}), \ldots, (H_{r,J+1}\prod_{j=J+1}^{-1} H_{0,j}), \ldots, H_{1,-1}, \ldots, H_{r,-1}]^t.\end{aligned} \tag{1.24}$$

In (1.23) and (1.24), $H_{\ell,-1}=H_\ell$, $\ell=0,1,\ldots,r$ and thus $\mathcal{A}_{-1}=\mathcal{H}_{-1}$.

As we will see that both $\mathcal{A}_J$ and $\mathcal{A}$ are the operators defined on $\ell_2(\mathbb{Z})$ into the tensor product space

$$\bigotimes_{\ell=0,j=1}^{r,|J|} \ell_2^{\ell,j}(\mathbb{Z}) \quad \text{and} \quad \bigotimes_{\ell=0,j=1}^{r,\infty} \ell_2^{\ell,j}(\mathbb{Z}),$$

respectively, where $\ell_2^{\ell,j}(\mathbb{Z})=\ell_2(\mathbb{Z})$. The reconstruction operators

$$\mathcal{A}_J^* = [(\prod_{j=-1}^{J} H_{0,j}^*), (\prod_{j=-1}^{J+1} H_{0,j}^* H_{1,J}^*), \ldots, (\prod_{j=-1}^{J+1} H_{0,j}^* H_{r,J}^*), \ldots, H_{1,-1}^*, \ldots, H_{r,-1}^*] \tag{1.25}$$

and

$$\mathcal{A}^* = [\ldots, (\prod_{j=-1}^{J-1} H_{0,j}^* H_{1,J-1}^*), \ldots, (\prod_{j=-1}^{J-1} H_{0,j}^* H_{r,J-1}^*), (\prod_{j=-1}^{J} H_{0,j}^* H_{1,J}^*), \ldots,$$
$$(\prod_{j=-1}^{J} H_{0,j}^* H_{r,J}^*), (\prod_{j=-1}^{J+1} H_{0,j}^* H_{1,J+1}^*), \ldots, (\prod_{j=-1}^{J+1} H_{0,j}^* H_{r,J+1}^*), \ldots, H_{1,-1}^*, \ldots, H_{r,-1}^*] \tag{1.26}$$

are the adjoint operators of $\mathcal{A}_J$ and $\mathcal{A}$ respectively.

The operators $\mathcal{A}_J$ and $\mathcal{A}$ are closely related to P_0 and Q_0^q. By Lemma 1.1 we have the identity

$$P_0 f = P_J^q f + \sum_{\ell=1}^{r} \sum_{j=J}^{-1} \sum_{k\in\mathbb{Z}} \langle f, \psi_{\ell,j,k}^q \rangle, \quad J < 0.$$

The corresponding coefficients in the right hand side is $\mathcal{A}_J \boldsymbol{v}_{0,0}$ with $\boldsymbol{v}_{0,0} = \{\langle f, \phi_{0,k}\rangle\}$. Similarly, the coefficients in the right hand side of the identity used in analysis

$$P_0 f = Q_0^q f$$

can be obtained by $\mathcal{A}\boldsymbol{v}_{0,0}$. Furthermore, the next proposition shows that the decomposition and reconstruction process is perfect, i.e. $\mathcal{A}_J^*\mathcal{A}_J = I$ and $\mathcal{A}^*\mathcal{A} = I$.

Proposition 1.2. *The decomposition operators $\mathcal{A}_J$ and $\mathcal{A}$, as defined in* (1.23) *and* (1.24) *respectively, satisfy $\mathcal{A}_J^*\mathcal{A}_J = I$ and $\mathcal{A}^*\mathcal{A} = I$ where I is the identity operator.*

Proof. The result on $\mathcal{A}_J$ can be proved by induction. When $J = -1$, this follows from (1.19). For arbitrary $J < 0$, we start from the definition of $\mathcal{A}_J$. By (1.23), we have

$$\begin{aligned}
\mathcal{A}_J^*\mathcal{A}_J &= \Big(\prod_{j=-1}^{J+1} H_{0,j}^* H_{1,J}^*\Big)\Big(H_{1,J} \prod_{j=J+1}^{-1} H_{0,j}\Big) + \cdots + \Big(\prod_{j=-1}^{J+1} H_{0,j}^* H_{r,J}^*\Big)\Big(H_{r,J} \prod_{j=J+1}^{-1} H_{0,j}\Big) \\
&\quad + \Big(\prod_{j=-1}^{J} H_{0,j}^*\Big)\Big(\prod_{j=J}^{-1} H_{0,j}\Big) + \sum_{j=J+1}^{-1} \sum_{\ell=1}^{r} \Big(\prod_{j'=-1}^{j+1} H_{0,j'}^* H_{\ell,j}^*\Big)\Big(H_{\ell,j} \prod_{j'=j+1}^{-1} H_{0,j'}\Big) \\
&= \Big(\prod_{j=-1}^{J+1} H_{0,j}^*\Big)\Big(\prod_{j=J+1}^{-1} H_{0,j}\Big) + \sum_{j=J+1}^{-1} \sum_{\ell=1}^{r} \Big(\prod_{j'=-1}^{j+1} H_{0,j'}^* H_{\ell,j}^*\Big)\Big(H_{\ell,j} \prod_{j'=j+1}^{-1} H_{0,j'}\Big) \\
&= \mathcal{A}_{J+1}^*\mathcal{A}_{J+1}.
\end{aligned}$$

In the above, $\prod_{j'=j}^{-1} H_{0,j'} = H_{0,j}H_{0,j+1}\cdots H_{0,-1}$ and $\prod_{j'=-1}^{j} H_{0,j'} = H_{0,-1}H_{0,-2}\cdots H_{0,j}$. The last equality can be viewed as the reconstruction process from Jth level to $(J+1)$th level and identity $\mathcal{A}_J^*\mathcal{A}_J = I$ holds for $J<0$ by induction.

For operator $\mathcal{A}$, we note that proving $\mathcal{A}^*\mathcal{A} = I$ is equivalent to proving $\langle \mathcal{A}\boldsymbol{v}, \mathcal{A}\boldsymbol{v}\rangle = \langle \boldsymbol{v}, \boldsymbol{v}\rangle$ holds for any sequence $\boldsymbol{v} \in \ell_2(\mathbb{Z})$. We next note that

$$\begin{aligned}\langle \boldsymbol{v}, \boldsymbol{v}\rangle &= \langle \mathcal{A}_J\boldsymbol{v}, \mathcal{A}_J\boldsymbol{v}\rangle \\ &= \Big(\boldsymbol{v}^* \prod_{j=-1}^{J} H_{0,j}^*\Big)\Big(\prod_{j=J}^{-1} H_{0,j}\boldsymbol{v}\Big) + \sum_{j=J}^{-1}\sum_{\ell=1}^{r}\Big(\boldsymbol{v}^* \prod_{j'=-1}^{j+1} H_{0,j'}^* H_{\ell,j}^*\Big)\Big(H_{\ell,j}\prod_{j'=j+1}^{-1} H_{0,j'}\boldsymbol{v}\Big)\end{aligned} \tag{1.27}$$

and

$$\langle \mathcal{A}\boldsymbol{v}, \mathcal{A}\boldsymbol{v}\rangle = \sum_{j=-\infty}^{-1}\sum_{\ell=1}^{r}\Big(\boldsymbol{v}^* \prod_{j'=-1}^{j+1} H_{0,j'}^* H_{\ell,j}^*\Big)\Big(H_{\ell,j}\prod_{j'=j+1}^{-1} H_{0,j'}\boldsymbol{v}\Big). \tag{1.28}$$

Thus to show $\langle \mathcal{A}\boldsymbol{v}, \mathcal{A}\boldsymbol{v}\rangle = \langle \boldsymbol{v}, \boldsymbol{v}\rangle$ we only need to prove that $\prod_{j=J}^{-1} H_{0,j}\boldsymbol{v} \to \boldsymbol{0}$ as $J \to -\infty$. Since the matrix $H_{0,j}$ are Toeplitz matrices generated by filters $\boldsymbol{h}_{0,j}$, we have

$$\widehat{\prod_{j=J}^{-1} H_{0,j}\boldsymbol{v}} = \prod_{j=J}^{-1} \widehat{\boldsymbol{h}_{0,j}}\widehat{\boldsymbol{v}}.$$

Since $|\widehat{\boldsymbol{h}_{0,j}}| \le 1$,

$$\left|\widehat{\prod_{j=J}^{-1} H_{0,j}\boldsymbol{v}}\right| \le |\widehat{\boldsymbol{v}}|, \quad \text{a.e.} \quad \omega \in \mathbb{R}.$$

Note that the compactly supported refinable function ϕ obtained from the finite length low pass filter $\boldsymbol{h}_0$ can be written as $\widehat{\phi}(\omega) = \prod_{j=0}^{\infty}\widehat{\boldsymbol{h}_0}(2^{-j-1}\omega)$. Since $\phi \in L_2(\mathbb{R})$ is compactly supported, we have $\phi \in L_1(\mathbb{R})$ and $\widehat{\phi} \ne 0$ a.e. $\omega \in \mathbb{R}$ with $\widehat{\phi} \to 0$ as $\omega \to \pm\infty$. Suppose zero set of $\widehat{\phi}$ is $\mathcal{Z}$, which is a zero measure set. Next we consider any $\omega \in \mathbb{R}\backslash\mathcal{Z}$ such that $\widehat{\phi}(\omega) \ne 0$. Because $\widehat{\boldsymbol{h}_{0,j}}(\omega) = \widehat{\boldsymbol{h}_0}(2^{-j-1}\omega)$, we have

$$\widehat{\prod_{j=J}^{-1} H_{0,j}\boldsymbol{v}} = \prod_{j=J}^{-1}\widehat{\boldsymbol{h}_{0,j}}\widehat{\boldsymbol{v}} = \prod_{j=J}^{-1}\widehat{\boldsymbol{h}_0}(2^{-j-1}\cdot)\widehat{\boldsymbol{v}} = \frac{1}{\prod_{j=0}^{\infty}\widehat{\boldsymbol{h}_0}(2^{-j-1}\cdot)}\prod_{j=J}^{\infty}\widehat{\boldsymbol{h}_0}(2^{-j-1}\cdot)\widehat{\boldsymbol{v}} = \frac{1}{\widehat{\phi}}\widehat{\phi}(2^{-J}\cdot)\widehat{\boldsymbol{v}}.$$

Thus

$$\lim_{J\to-\infty}\prod_{j=J}^{-1}\widehat{\boldsymbol{h}_{0,j}}\widehat{\boldsymbol{v}} = \frac{\widehat{\boldsymbol{v}}}{\widehat{\phi}}\lim_{J\to-\infty}\widehat{\phi}(2^{-J}\cdot) = 0.$$

So for any $\omega \in \mathbb{R}$, $|\prod_{j=J}^{-1} \widehat{\boldsymbol{h}_{0,j}}||\widehat{\boldsymbol{v}}| \to 0$ a.e. as $J \to -\infty$. Applying the Dominated Convergence Theorem, we obtain

$$\|\prod_{j=J}^{-1} H_{0,j}\boldsymbol{v}\|_{\ell_2(\mathbb{Z})} = \frac{1}{\sqrt{2\pi}}\|\prod_{j=J}^{-1} \widehat{\boldsymbol{h}_{0,j}}\widehat{\boldsymbol{v}}\|_{L_2[-\pi,\pi]} \to 0, \quad J \to -\infty.$$

Let $J \to -\infty$ in (1.27), we have $\langle \boldsymbol{v}, \boldsymbol{v}\rangle = \lim_{J\to-\infty}\langle \mathcal{A}_J\boldsymbol{v}, \mathcal{A}_J\boldsymbol{v}\rangle = \langle \mathcal{A}\boldsymbol{v}, \mathcal{A}\boldsymbol{v}\rangle$, which completes our proof. □

Chapter 2

Formulation and Algorithms

This chapter is to formulate the deconvolution problem via the multiresolution analysis and the framelet analysis. It converts the deconvolution problem to the problem of filling the missing framelet coefficients. Consider the convolution equation

$$\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{b} + \boldsymbol{\epsilon} = \boldsymbol{c}, \tag{2.1}$$

where $\boldsymbol{h}_0$ is a low pass filter with finite support and $\boldsymbol{b}$, $\boldsymbol{c}$ are the sequences in $\ell_2(\mathbb{Z})$. The error term $\boldsymbol{\epsilon} \in \ell_2(\mathbb{Z})$ satisfies $\|\boldsymbol{\epsilon}\|_{\ell_2(\mathbb{Z})} \leq \varepsilon$. To simplify our notation, we use $\|\cdot\| := \|\cdot\|_{\ell_2(\mathbb{Z})}$.

Our approach starts with the refinable function generated by the low pass filter $\boldsymbol{h}_0$. There are many sufficient conditions on the low pass filter $\boldsymbol{h}_0$ with $\widehat{\boldsymbol{h}_0}(0) = 1$, under which ϕ is in $L_2(\mathbb{R})$. Here we assume that $\boldsymbol{h}_0$ satisfies the following condition

$$|\widehat{\boldsymbol{h}_0}(\omega)|^2 + |\widehat{\boldsymbol{h}_0}(\omega + \pi)|^2 \leq 1, \quad \text{a.e.} \quad \omega \in \mathbb{R}. \tag{2.2}$$

The following proposition shows that the corresponding refinable function ϕ generated from $\boldsymbol{h}_0$ by assuming (2.2) is in $L_2(\mathbb{R})$.

Proposition 2.1. *Suppose $\boldsymbol{h}_0$ is finitely supported and satisfies the following condition:*

$$\begin{cases} |\widehat{\boldsymbol{h}_0}(\omega)|^2 + |\widehat{\boldsymbol{h}_0}(\omega + \pi)|^2 \leq 1, & a.e. \quad \omega \in \mathbb{R}; \\ \widehat{\boldsymbol{h}_0}(0) = 1. \end{cases} \tag{2.3}$$

The solution ϕ of the refinement equation

$$\phi(x) = 2\sum_{k\in\mathbb{Z}} \boldsymbol{h}_0[k]\phi(2x-k)$$

is in $L_2(\mathbb{R})$.

Proof. Since $\boldsymbol{h}_0$ is finitely supported and $\widehat{\boldsymbol{h}_0}(0) = 1$, the compactly supported refinable function ϕ exists in the sense of distribution with the Fourier transform of ϕ given by

$$\widehat{\phi}(\omega) = \prod_{j=1}^{\infty} \widehat{\boldsymbol{h}_0}(\frac{\omega}{2^j}), \tag{2.4}$$

satisfying $\widehat{\phi}(0) = 1$. Further, the distribution solution ϕ is unique. In the following, we will prove $\phi \in L_2(\mathbb{R})$ whenever $\boldsymbol{h}_0$ satisfies (2.3).

Our proof uses the cascade algorithm defined by

$$\widehat{\phi_n}(\omega) = \widehat{\boldsymbol{h}_0}(\frac{\omega}{2})\widehat{\phi_{n-1}}(\frac{\omega}{2}) = \prod_{j=1}^{n} \widehat{\boldsymbol{h}_0}(\frac{\omega}{2^j})\widehat{\phi_0}(\frac{\omega}{2^n}), \quad n>0, \tag{2.5}$$

with initial function ϕ_0 satisfying $\widehat{\phi_0}(\omega) = \chi_{[-\pi,\pi)}(\omega)$. It is known that the cascade algorithm always converges to ϕ as a distribution. Because $\widehat{\phi_0}(\omega)$ satisfies

$$\sum_{k\in\mathbb{Z}} |\widehat{\phi_0}(\omega+2k\pi)|^2 = 1, \quad \text{a.e.} \quad \omega\in\mathbb{R},$$

it can be proven inductively that for any ϕ_n, $n>0$,

$$\sum_{k\in\mathbb{Z}} |\widehat{\phi_n}(\omega+2k\pi)|^2 \le 1, \quad \text{a.e.} \quad \omega\in\mathbb{R}.$$

Thus we have

$$\|\widehat{\phi_n}\|^2_{L_2(\mathbb{R})} = \int_{\mathbb{R}} |\widehat{\phi_n}(\omega)|^2\,\mathrm{d}\omega = \int_{-\pi}^{\pi}\sum_{k\in\mathbb{Z}} |\widehat{\phi_n}(\omega+2k\pi)|^2\,\mathrm{d}\omega \le 2\pi.$$

Since the sequence $\{\|\widehat{\phi_n}\|_{L_2(\mathbb{R})}\}$ is bounded for each n, there exists a subsequence $\{\widehat{\phi_{n_j}}\}$ which converges weakly to some function $\widehat{g} \in L_2(\mathbb{R})$. As shown in [18], when $\boldsymbol{h}_0$ is finitely supported, $\widehat{\phi_n}$ in (2.5) converges absolutely and uniformly on compact sets. Thus the function $\widehat{\phi}$ is uniformly continuous on compact sets. Since $\widehat{\phi}(0) = 1$, in a neighborhood of

0, we have $\widehat{\phi} \neq 0$. Thus each $\widehat{\phi_{n_j}} \neq 0$ in such a neighborhood. It leads to the weak limit $\widehat{g} \neq 0$ in this neighborhood. On the other hand, because the sequence $\{\phi_n\}$ converges to the function ϕ in the sense of distribution, which is stronger than the weak convergence, we have $\phi = g \in L_2(\mathbb{R})$. □

Remark 2.1. It was shown in [14] that if $\boldsymbol{h}_0$ satisfies (2.3) and if the corresponding refinable function ϕ is in $L_2(\mathbb{R})$, then there is constructive way to derive a set of tight framelets. Further, if ϕ is symmetric, the framelets are symmetric or antisymmetric. Constructions of tight frames when the refinement mask $\boldsymbol{h}_0$ satisfies (2.3) are also given in [20] in their construction of tight frames from pseudo-splines (also available in [22]). The above proposition shows that condition (2.3) on $\boldsymbol{h}_0$ implies the corresponding refinable function $\phi \in L_2(\mathbb{R})$.

We further remark that (2.2) is not a strong assumption. For example, all refinement masks of B-splines, the refinable functions whose shifts form an orthonormal system derived in [18], the base functions of interpolatory functions, and more general, pseudo-splines introduced by [20] and [22] satisfy this assumption. In fact, many low pass filters used in practical problems satisfy (2.2). For example, the low pass filters used in high resolution image reconstructions satisfy (2.2). Furthermore, with this assumption, we can construct a corresponding tight frame system via unitary extension principle of [34] which is used in our algorithm.

To make our ideas work here, the crucial step is to construct a tight frame system via a multiresolution analysis with underlying refinement mask being the given low pass filter. The assumption (2.2) is a necessary and sufficient condition to have a tight frame system associated with the given low pass filter. When the refinable function ϕ is in $L_2(\mathbb{R})$, whose refinement mask is the given low pass filter in (2.1), together with some additional minor conditions, we can always obtain a bi-frame system via the mixed unitary extension principle of [35] and more generally the mixed oblique extension principle of [15] and [20]. For example, let

$$h_0(z) := \sum_{k\in\mathbb{Z}} \boldsymbol{h}_0[k] z^{-k}.$$

Then (2.2) can be replaced by the condition that $h_0(z)$ and $h_0(-z)$ have no common zeros in complex domain. With this, one can construct a bi-frame system by using the mixed unitary extension principle. This is essentially the approach taken by [8]. Our analysis can be carried out with some efforts. To simplify our discussion here, we only use the tight frame system, hence assuming (2.2).

Finally, since our approach based on denoising schemes that threshold of framelet coefficients, we implicitly assume that the underlying function of the data set has a sparse representation by the tight frame system used and the errors are small and spread in the frame transform domain.

2.1 Formulation in MRA

This section is to formulate the problem of solving

$$\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{b} + \boldsymbol{\epsilon} = \boldsymbol{c} \tag{2.6}$$

via the multiresolution analysis framework. As we will see, the approach here reduces solving equation (2.6) to the problem of filling the missing framelet coefficients. This approach was first taken by [8], however, we give a complete analysis and formulation here.

As we mentioned before, by using

$$P_J f = D^J P_0 D^{-J} f \quad \text{and} \quad P_{J-1}^{q,J} f = D^J P_{-1}^q D^{-J} f,$$

we may assume that data set is given on level $J = 0$ without loss of generality. In fact, when the data set is given in $2^{-J}\mathbb{Z}$, we consider function $f(2^{-J}\cdot)$ instead of f. The approximation power of a function f in space V_J is the same as that of the function $f(2^{-J}\cdot)$ in space V_0.

Let $\phi \in L_2(\mathbb{R})$ be the refinable function with refinement mask $\boldsymbol{h}_0$ and $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be high pass filters obtained via the UEP which are the framelet masks of $\psi_1, \ldots, \psi_r$. First we suppose that the given data set contains no error, i.e. $\boldsymbol{\epsilon} = \mathbf{0}$. The convolution equation

$\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{b}$ implies that $\boldsymbol{b}$ is obtained by passing the original sequence $\boldsymbol{v}$ through a low pass filter $\boldsymbol{h}_0$. Assume that $\boldsymbol{b} = \{\langle S, \phi^q_{-1,k}\rangle\}$, where $S \in L_2(\mathbb{R})$ is the underlying function from that the data set $\boldsymbol{b}$ is obtained. Then we are given

$$P^q_{-1}S = \sum_{k\in\mathbb{Z}} \langle S, \phi^q_{-1,k}\rangle \phi^q_{-1,k} = \sum_{k\in\mathbb{Z}} \boldsymbol{b}[k]\phi^q_{-1,k}. \tag{2.7}$$

Let $\boldsymbol{v}^S = \{\langle S, \phi_{0,k}\rangle\}$, then

$$P_0 S = \sum_{k\in\mathbb{Z}} \langle S, \phi_{0,k}\rangle \phi_{0,k} = \sum_{k\in\mathbb{Z}} \boldsymbol{v}^S[k]\phi_{0,k}. \tag{2.8}$$

Applying the framelet decomposition algorithm (1.12), one obtains that $\boldsymbol{h}_0 * \boldsymbol{v}^S = \boldsymbol{b}$. This implies that solving equation (2.6) is equivalent to reconstructing the quasi-interpolation $P_0S \in V_0$ from the quasi-interpolation $P^q_{-1}S \in V^q_{-1}$. Since

$$P_0 S = P^q_{-1}S + \sum_{\ell=1}^{r}\sum_{k\in\mathbb{Z}} \langle S, \psi^q_{\ell,-1,k}\rangle \psi^q_{\ell,-1,k},$$

to recover $\boldsymbol{v}^S = \{\langle S, \phi_{0,k}\rangle\}$ from given $\boldsymbol{b}$, we need the framelet coefficients $\{\langle S, \psi^q_{\ell,-1,k}\rangle\}$. This leads to an iterative algorithm that restores $\boldsymbol{v}^S$ from data $\boldsymbol{b}$ iteratively by updating the framelet coefficients $\{\langle S, \psi^q_{\ell,-1,k}\rangle\}$ in each iteration. All these have been given in [8] and consequent papers [10, 11] in their reconstructions of high resolution images. In fact, it motivates the algorithms developed in [8, 10, 11].

By this approach, we not only give a solution of (2.6), but also give an interpretation in terms of the underlying function S where we view the data $\boldsymbol{b} = \{\langle S, \phi^q_{-1,k}\rangle\}$ as the given sample of S. Under this setting, we are given $P^q_{-1}S \in V^q_{-1}$, and the solution of (2.6) leads to $P_0S \in V_0$, which is a higher resolution subspace in the multiresolution analysis. Although there are more than one function whose quasi-interpolations is $P^q_{-1}S$ and P_0S given as (2.7) and (2.8), we never get the underlying function S. One can only expect to obtain a better approximation P_0S of S from the given $P^q_{-1}S$ approximation. The approximation power of P_0S and $P^q_{-1}S$ and their difference can be established for smooth functions by applying the corresponding results in [20] which depends on the properties of the underlying refinable function; more general for piecewise smooth functions, it can be studied by applying results and ideas from [2] and [3] which depends on the properties of the framelets. We omit the detailed discussion here.

Roughly speaking, the idea of solving equation (2.6) here can be understood as for a given coarse level approximation $P^q_{-1}S$ to find a finer level approximation P_0S which is reduced to find the coefficients $\boldsymbol{v}^S = \{\langle S, \phi_{0,k}\rangle\}$. The derivation of $\boldsymbol{v}^S$ is an iterative process which recovers P_0S from $P^q_{-1}S$ as discussed before and detailed in the algorithms given in the next section. Then $\boldsymbol{h}_0 * \boldsymbol{v}^S = \boldsymbol{b}$ by the decomposition algorithm (1.12) and we conclude that $\boldsymbol{v}^S$ is a solution of (2.6).

However, the data given may contain errors, i.e. instead of $\boldsymbol{b}$, the data is given by $\boldsymbol{c} = \boldsymbol{b} + \boldsymbol{\epsilon}$. Furthermore, the given data set $\boldsymbol{b}$ may not be necessary of the form of $\{\langle S, \phi^q_{-1,k}\rangle\}$, for some $S \in L_2(\mathbb{R})$. In both cases, the exact $\ell_2(\mathbb{Z})$ solution of $\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{c}$ may not exist or it may not be desirable or not be possible to get the exact solution.

Nevertheless, there is a need to have

$$\tilde{s} = \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \tilde{s}_{\ell,j,k} \psi^q_{\ell,j,k} \in V_0$$

to approximate the underlying function where the sample data set $\boldsymbol{c}$ comes from. Let

$$\tilde{\boldsymbol{s}} = \{\tilde{s}_{\ell,j,k}\}, \quad \text{and} \quad \boldsymbol{s} = \mathcal{A}^* \tilde{\boldsymbol{s}}, \tag{2.9}$$

where $\mathcal{A}^*$ is the reconstruction operator given in (1.26). For the vector $\boldsymbol{s}$ being a candidate of the solution of (2.6), it requires $\boldsymbol{h}_0 * \boldsymbol{s}$ within the $\boldsymbol{\epsilon}$ ball of $\boldsymbol{c}$ and the function $\tilde{s}$ has some smoothness. The smoothness of the function is reflected by the decay of the framelet coefficients which is measured by the ℓ_p norm of $\tilde{\boldsymbol{s}}$. Given any sequence $\boldsymbol{v}$ determined by three indices (ℓ, j, k) with $\ell = 1, \ldots, r$, $j < 0$ and $k \in \mathbb{Z}$, we say $\boldsymbol{v}$ is in space ℓ_p, for a given p, if $\sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} |v_{\ell,j,k}|^p < \infty$.

Assuming that there exists function S such that $\tilde{s}_{\ell,j,k} = \langle S, \psi^q_{\ell,j,k}\rangle$, then function $\tilde{s} = Q^q_0 S$. For given $1 \le p \le 2$, we say that the pair $(\boldsymbol{s}, \tilde{\boldsymbol{s}})$ defined in (2.9) is the solution of (2.6) (and $\tilde{s}$ is an approximation of the underlying function of the data set) if for all $g \in L_2(\mathbb{R})$, let $\tilde{g} = \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \langle g, \psi^q_{\ell,j,k}\rangle \psi^q_{\ell,j,k} = Q^q_0 g$ with $\tilde{\boldsymbol{g}} = \{\langle g, \psi^q_{\ell,j,k}\rangle\}$ in ℓ_p and let $\boldsymbol{g} = \mathcal{A}^* \tilde{\boldsymbol{g}}$, the following inequality

$$\|\boldsymbol{h}_0 * \boldsymbol{g} - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\langle g, \psi^q_{\ell,j,k}\rangle|^p \geq \|\boldsymbol{h}_0 * \boldsymbol{s} - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\langle S, \psi^q_{\ell,j,k}\rangle|^p \tag{2.10}$$

holds. Here $\gamma \leq \lambda_j \leq \gamma'$, $j \in \mathbb{Z}$, where $0 < \gamma \leq \gamma' \leq \infty$, are parameters which will be determined by the error level.

The function $\tilde{s}$ is considered as an approximation of the underlying function whose sample is given by $\boldsymbol{c}$. The first term measures the residue of the solution $\boldsymbol{s}$ and the given data set $\boldsymbol{c}$. The second term is a penalization term using a weighted (with weights λ_j) ℓ_p-norm of the coefficients of framelets. Since the framelet coefficients are closely related to the smoothness of the underlying function (see [2, 3]), minimization problem (2.10) balances the fitness of the solution and the smoothness of the solution function $\tilde{s}$.

The minimization condition (2.10) can be stated as following: for a fixed $1 \leq p \leq 2$, the pair $(\boldsymbol{s}, \tilde{\boldsymbol{s}})$ defined in (2.9) is a solution of (2.6) (the function $\tilde{s}$ is an approximation of the underlying function of the data) if for all $\eta \in L_2(\mathbb{R})$ with $\tilde{\boldsymbol{\eta}} = \{\tilde{\eta}_{\ell,j,k}\} = \{\langle \eta, \psi^q_{\ell,j,k}\rangle\} \in \ell_p$, the pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$, where $\boldsymbol{\eta} = \mathcal{A}^*\tilde{\boldsymbol{\eta}}$, satisfies the following inequality

$$\|\boldsymbol{h}_0 * (\boldsymbol{s} + \boldsymbol{\eta}) - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \geq \|\boldsymbol{h}_0 * \boldsymbol{s} - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}_{\ell,j,k}|^p. \tag{2.11}$$

However, as we will see that the sequence $\tilde{\boldsymbol{s}}$ is uniquely determined by algorithm, it may not be of the form $\{\langle S, \psi^q_{\ell,j,k}\rangle\}$ for any $S \in L_2(\mathbb{R})$, since $\{\psi^q_{\ell,j,k}\}_{j<0}$ is redundant which implies that the representation $\tilde{s}$ is not unique. Nevertheless, the pair $(\boldsymbol{s}, \tilde{\boldsymbol{s}})$ can still be considered as a solution of equation (2.6) if (2.11) holds with the pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ satisfying $\tilde{\boldsymbol{\eta}} = \{\tilde{\eta}_{\ell,j,k}\} = \{\langle \eta, \psi^q_{\ell,j,k}\rangle\} \in \ell_p$ and $\boldsymbol{\eta} = \mathcal{A}^*\tilde{\boldsymbol{\eta}}$, for all $\eta \in L_2(\mathbb{R})$. Here, we remark that since $\tilde{\boldsymbol{\eta}} = \{\langle \eta, \psi^q_{\ell,j,k}\rangle\}$, $\boldsymbol{\eta} = \mathcal{A}^*\tilde{\boldsymbol{\eta}}$ implies that $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta}$ by the decomposition algorithm.

The function $\tilde{s}$ enters the discussion to give an analysis in the function form of the underlying solution. The underlying function and $\tilde{s}$ play a role of analysis, but does not enter the algorithm. Next, we link the formulation to a direct discrete form of minimization problem (2.11). The minimization problem (2.11) can be stated as follows: for a given $1 \leq p \leq 2$, a pair of sequences $(\boldsymbol{s}, \tilde{\boldsymbol{s}})$, satisfying $\tilde{\boldsymbol{s}} \in \ell_p$ and $\boldsymbol{s} = \mathcal{A}^*\tilde{\boldsymbol{s}}$, is the solution of (2.6) if for arbitrary pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ satisfying $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta} \in \ell_p$, the following inequality holds:

$$\|\boldsymbol{h}_0 * (\boldsymbol{s}+\boldsymbol{\eta}) - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \geq \|\boldsymbol{h}_0 * \boldsymbol{s} - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}_{\ell,j,k}|^p. \tag{2.12}$$

We should remark here the condition $\boldsymbol{s} = \mathcal{A}^*\tilde{\boldsymbol{s}}$ on the pair $(\boldsymbol{s}, \tilde{\boldsymbol{s}})$ is different from the condition $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta}$ on the pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$. The condition $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta}$ implies $\boldsymbol{\eta} = \mathcal{A}^*\tilde{\boldsymbol{\eta}}$, since $\mathcal{A}^*\tilde{\boldsymbol{\eta}} = \mathcal{A}^*\mathcal{A}\boldsymbol{\eta} = \boldsymbol{\eta}$ by $\mathcal{A}^*\mathcal{A} = \mathsf{I}$. However,the condition $\boldsymbol{s} = \mathcal{A}^*\tilde{\boldsymbol{s}}$, in general, does not implies $\tilde{\boldsymbol{s}} = \mathcal{A}\boldsymbol{s}$, unless $\mathcal{A}\mathcal{A}^* = \mathsf{I}$ or $\tilde{\boldsymbol{s}}$ happens to be $\mathcal{A}\boldsymbol{s}$. Note that the identity $\mathcal{A}\mathcal{A}^* = \mathsf{I}$ does not hold for any redundant system. The reasons for imposing the different conditions are due to that $(\boldsymbol{s}, \tilde{\boldsymbol{s}})$ is obtained by the algorithm which only satisfies $\boldsymbol{s} = \mathcal{A}^*\tilde{\boldsymbol{s}}$, while for given $\boldsymbol{\eta}$, there is more than one $\tilde{\boldsymbol{\eta}}$ such that $\mathcal{A}^*\tilde{\boldsymbol{\eta}} = \boldsymbol{\eta}$. We choose the canonical pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ with $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta}$.

2.2 Algorithms

We give the algorithms to solve (2.6) with the formulation in MRA. In our approach, the algorithm iteratively improves the framelet coefficients using the previous iterative result in each iteration. Let $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the sequences derived from $\boldsymbol{h}_0$ via the UEP and $\mathsf{H}_0, \mathsf{H}_1, \ldots, \mathsf{H}_r$ be the corresponding Toeplitz matrices. Our algorithms based on the UEP condition

$$\mathsf{H}_0^*\mathsf{H}_0 + \sum_{\ell=1}^{r} \mathsf{H}_\ell^*\mathsf{H}_\ell = \mathsf{I}. \tag{2.13}$$

Let $\boldsymbol{v}_n$ be the solution for the nth iteration, then

$$\mathsf{H}_0^*\mathsf{H}_0\boldsymbol{v}_n + \sum_{\ell=1}^{r} \mathsf{H}_\ell^*\mathsf{H}_\ell\boldsymbol{v}_n = \boldsymbol{v}_n. \tag{2.14}$$

First, we consider the case that $\boldsymbol{b} = \{\langle S, \phi_{-1,k}^q \rangle\}$, where S is the underlying function and $\boldsymbol{b}$ is the given data as a set of the samples of S, and $\boldsymbol{\epsilon} = \mathbf{0}$. Then by $\boldsymbol{h}_0 * \boldsymbol{v}^S = \boldsymbol{b}$ with $\boldsymbol{v}^S = \{\langle S, \phi_{0,k} \rangle\}$, we have $\boldsymbol{v}^S$ is a solution to equation (2.6). In each iteration, we can replace $\mathsf{H}_0\boldsymbol{v}_n$ by the known data $\boldsymbol{b}$ to improve the approximation. This can also be viewed as that we use the framelet coefficients of the nth iteration to approximate the framelet coefficients of the underlying function S. We summarize the algorithm as follows:

Algorithm 2.1.

(i) *Choose an initial approximation* $\boldsymbol{v}_0$ *(e.g.* $\boldsymbol{v}_0 = \boldsymbol{b}$*);*

(ii) *Iterate on* n *until convergence:*

$$\boldsymbol{v}_{n+1} = H_0^*\boldsymbol{b} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{v}_n. \tag{2.15}$$

As will see in the next section, Algorithm 2.1 converges, but it converges very slowly. We need to adjust the iteration in Algorithm 2.1 to quicken the convergence. This motivates us to introduce the acceleration factor $0 < \beta < 1$ into the above algorithm and the new iteration with β is given below:

$$\boldsymbol{v}_{n+1} = \beta(H_0^*\boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{v}_n) = H_0^*\beta\boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{v}_n. \tag{2.16}$$

This scheme can be viewed as the traditional regularization method used in noise removal, the solution of which satisfies the matrix equation

$$\Big(H_0^* H_0 + (1-\beta)\sum_{\ell=1}^{r} H_\ell^* H_\ell\Big)\boldsymbol{v} = H_0^*\beta\boldsymbol{c}.$$

Here β is a regularization parameter. The solution of the original convolution equation (2.6) is $\boldsymbol{v} = \boldsymbol{v}^\beta/\beta$ with $\boldsymbol{v}^\beta$ the solution to the above matrix equation. The solution $\boldsymbol{v}$ minimizes the following functional:

$$\|H_0\boldsymbol{v} - \boldsymbol{c}\|^2 + \frac{(1-\beta)^2}{\beta^2}\|\boldsymbol{v}\|^2.$$

This is the standard regularization form with a special regularization operator, which was more or less the [8, Algorithm 2] given to us. The parameter β has to be carefully chosen to balance the error and smoothness of the solution. It plays a role in both convergence acceleration and error removal. However, when a different penalty functional instead of ℓ_2 norm of the solution (e.g. the one given in the formulation), which is desirable in many applications, is used, we need a different approach. In our new algorithms, the acceleration factor β is mainly used to accelerate the convergence and leave the "regularization" part to a threshold process. Finally, we remark that, as will see in §4, in the numerical implementation, when proper boundary conditions (e.g. periodic boundary condition) are used, the matrix H_0 becomes a nonsingular finite order matrix. The iteration in Algorithm 2.1 converges with a rate $1-\lambda$, where λ is the minimum eigenvalue of $H_0^* H_0$. Hence, we do not need to introduce the acceleration factor β.

Next, we introduce the following denoising operators to the iteration (2.16).

Denoising Operator. When data are contaminated with errors, we need to remove the error term from each iteration before putting it into the next iteration. The denoising scheme is needed to prevent the limit of iteration (2.16) from following the noise residing in $\boldsymbol{c}$. For any vector $\boldsymbol{v}$, let threshold operator be

$$\mathcal{D}_\lambda^p(\boldsymbol{v}) := \left(t_\lambda^p(\boldsymbol{v}[0]), t_\lambda^p(\boldsymbol{v}[1]), \ldots\right), \quad 1 \le p \le 2, \tag{2.17}$$

where $t_\lambda^p(x)$ is the threshold function. When $p=1$, $t_\lambda(x) := t_\lambda^1(x)$ is the soft-threshold function $\operatorname{sgn}(x)\max(|x|-\lambda/2, 0)$; when $1 < p \le 2$, the threshold function is defined by the inverse of function

$$\mathrm{F}_\lambda^p(x) := x + \frac{p\lambda}{2}\operatorname{sgn}(x)|x|^{p-1}. \tag{2.18}$$

Function $\mathrm{F}_\lambda^p(x)$ is a one-to-one differentiable function with unique inverse. For $1 < p \le 2$, the explicit formula of the inverse of function F_λ^p is not always available. Numerical method may be needed to calculate the value of $t_\lambda^p(x) := (\mathrm{F}_\lambda^p)^{-1}(x)$. Further, the threshold function is nonexpansive, i.e. for any $x \in \mathbb{R}$, we have $|t_\lambda^p(x)| \le |x|$. As we will see, the difference of the threshold operators $\mathcal{D}_\lambda^p$ according to different values of p is that the limit of the algorithm has different optimal properties.

When a sequence $\boldsymbol{v}$ is given, normal procedure is first transforming $\boldsymbol{v}$ to the framelet domain via the decomposition operator $\mathcal{A}$ to decorrelate the signal, and then applying the threshold operator $\mathcal{D}_{\lambda_j}^p$ with the threshold parameter λ_j depending on the decomposition level j. For a given sequence $\boldsymbol{v} \in \ell_2(\mathbb{Z})$, the denoising operator $\mathcal{T}^p$ which applies the threshold operator $\mathcal{D}^p_{\lambda_j}$ on $\mathcal{A}\boldsymbol{v}$ with the threshold parameters $\{\lambda_j\}$ is defined as:

$$\mathcal{T}^p\mathcal{A}(\boldsymbol{v}) = [\mathcal{D}^p_{\lambda_j}(H_{\ell,j}\prod_{j'=j+1}^{-1} H_{0,j'}\boldsymbol{v})]^t_{\ell,j}, \quad 1 \le p \le 2, \quad \ell = 1, 2, \ldots, r, \quad j < 0. \tag{2.19}$$

This noise removal scheme will then be applied at each iteration before applying the next iteration in Algorithm 2.1.

Algorithm 2.2 is motivated by [19]. At the nth step, the threshold operator is applied to the framelet decomposition of $H_0^*\beta\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* H_\ell \beta \boldsymbol{v}_n$. The parameters λ_j are fixed during the iteration.

Algorithm 2.2.

(i) *Choose an initial approximation* $\boldsymbol{v}_0$ *(e.g.* $\boldsymbol{v}_0 = \boldsymbol{c}$*);*

(ii) *Iterate on* n *until convergence:*

$$\boldsymbol{v}_{n+1} = \mathcal{A}^*\mathcal{T}^p\mathcal{A}(H_0^*\beta\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* H_\ell \beta \boldsymbol{v}_n); \tag{2.20}$$

(iii) *Suppose the limit of step (ii) is* $\boldsymbol{v}^\beta$*. Then the final solution is*

$$\boldsymbol{s}^\beta = \frac{1}{\beta}\boldsymbol{v}^\beta.$$

We will prove that the pair $(\boldsymbol{s}^\beta, \tilde{\boldsymbol{s}}^\beta)$ where $\tilde{\boldsymbol{s}}^\beta = \frac{1}{\beta}\mathcal{T}^p\mathcal{A}(H_0^*\beta\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* H_\ell \beta \boldsymbol{v}^\beta)$, obtained from step (iii) of Algorithm 2.2 satisfies the inequality (2.12) (up to arbitrary small ε). Next algorithm has a different denoising scheme from Algorithm 2.2. Instead of applying the denoising operator to each iteration before it is put into the next iteration, the denoising operator only acts on the approximation of the missing framelets coefficients. This is the process suggested by [8, 10, 11].

Algorithm 2.3.

(i) *Choose an initial approximation* $\boldsymbol{v}_0$ *(e.g.* $\boldsymbol{v}_0 = \boldsymbol{c}$*);*

(ii) *Iterate on* n *until convergence:*

$$\boldsymbol{v}_{n+1} = H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* \left(\mathcal{A}^* \mathcal{T}^p \mathcal{A}\right) \left(\beta H_\ell \boldsymbol{v}_n\right); \tag{2.21}$$

(iii) *Let* $\boldsymbol{v}^\beta$ *be the final iterative solution from (ii). Then the solution to the algorithm is*

$$\boldsymbol{s}^\beta = \boldsymbol{v}^\beta / \beta.$$

For better denoising effect, we may apply the denoising scheme to the final result $\boldsymbol{s}^\beta$, i.e. we take an additional step

(iv) $\boldsymbol{v} = \mathcal{A}^* \mathcal{T}^p \mathcal{A}(\boldsymbol{s}^\beta)$

to further remove the error effect arose by $\boldsymbol{c}$, which is used in [8, 10, 11].

Chapter 3

Analysis of Algorithms

This chapter focuses on the analysis of the algorithms given in §2.2. We first show that all algorithms converge. Secondly, we prove that the solutions of Algorithm 2.2 and 2.3 satisfy some minimization property.

3.1 Convergence

In this section, we will show the convergence of Algorithm 2.1, 2.2 and 2.3. The proof of the convergence of Algorithm 2.1 was given in [8] and [11]. We include the proof here for the sake of the self completeness of the paper. However, the proofs of the convergence of Algorithm 2.2 and 2.3 are new. This is important, since both algorithms are the ones used in practice.

Proposition 3.1. *Let $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the high pass filters of a tight frame system derived by the UEP with finitely supported $\boldsymbol{h}_0$ being the given low pass filter which satisfies* (2.2)*. Suppose there exists a function S such that $\boldsymbol{c} = \{\langle S, \phi^q_{-1,k}\rangle\}$. Then for arbitrary $\boldsymbol{v}_0 \in \ell_2(\mathbb{Z})$, the sequence $\boldsymbol{v}_n$ defined by* (2.15) *converges to $\boldsymbol{v} = \{\langle S, \phi^q_{0,k}\rangle\}$ with $\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{c}$.*

Proof. The proof was given in [8]. Writing (2.15) in frequency domain, one obtains

$$\widehat{\boldsymbol{v}}_{n+1} = \overline{\widehat{\boldsymbol{h}_0}}\widehat{\boldsymbol{c}} + \sum_{\ell=1}^{r} \overline{\widehat{\boldsymbol{h}_\ell}}\widehat{\boldsymbol{h}_\ell}\widehat{\boldsymbol{v}}_n.$$

Let $\boldsymbol{v} = \{\langle S, \phi_{0,k}\rangle\}$. Since $\boldsymbol{c} = \{\langle S, \phi^q_{-1,k}\rangle\}$, $\boldsymbol{v}$ is the solution to (2.6). Using the UEP condition, we have

$$\widehat{\boldsymbol{v}} = \overline{\widehat{\boldsymbol{h}_0}}\widehat{\boldsymbol{c}} + \sum_{\ell=1}^{r} \overline{\widehat{\boldsymbol{h}_\ell}}\widehat{\boldsymbol{h}_\ell}\widehat{\boldsymbol{v}}.$$

For arbitrary $\boldsymbol{v}_0 \in \ell_2(\mathbb{Z})$, applying the iteration n times, we have

$$\widehat{\boldsymbol{v}}_n - \widehat{\boldsymbol{v}} = \left(\sum_{\ell=1}^{r} \overline{\widehat{\boldsymbol{h}_\ell}}\widehat{\boldsymbol{h}_\ell}\right)^n (\widehat{\boldsymbol{v}}_0 - \widehat{\boldsymbol{v}}).$$

From (2.2), we have $0 \le |\widehat{\boldsymbol{h}_0}(\omega)| \le 1$ a.e. $\omega \in \mathbb{R}$ and $|\widehat{\boldsymbol{h}_0}(\omega)| = 0$ only holds on a zero measure set since $\widehat{\boldsymbol{h}_0}(\omega)$ is a polynomial the zero points of which are finite. Because $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ satisfy (1.5), it follows that

$$\sum_{\ell=1}^{r} |\widehat{\boldsymbol{h}_\ell}(\omega)|^2 \le 1, \quad \text{a.e.} \quad \omega \in \mathbb{R}$$

and the equality only holds on a zero measure set. Thus we have $|\widehat{\boldsymbol{v}}_n - \widehat{\boldsymbol{v}}| \le |\widehat{\boldsymbol{v}}_0 - \widehat{\boldsymbol{v}}|$ and $\widehat{\boldsymbol{v}}_n - \widehat{\boldsymbol{v}} \to 0$ a.e. $\omega \in \mathbb{R}$ as $n \to \infty$. Then by Dominated Convergence Theorem, $\|\boldsymbol{v}_n - \boldsymbol{v}\|_{\ell_2(\mathbb{Z})} = \frac{1}{\sqrt{2\pi}}\|\widehat{\boldsymbol{v}}_n - \widehat{\boldsymbol{v}}\|_{L_2[-\pi,\pi]} \to 0$, i.e. $\boldsymbol{v}_n$ converges to $\boldsymbol{v}$ as $n \to \infty$. □

Since $|\sum_{\ell=1}^{r} \overline{\widehat{\boldsymbol{h}_\ell}}\widehat{\boldsymbol{h}_\ell}| = 1$ at π, the convergence of the algorithm is slow. That is the reason we introduce the acceleration factor β into iteration. The convergence of iteration (2.16) can be proved similarly. Next we show the convergence of the iterations in Algorithm 2.2 and Algorithm 2.3. The following lemma is needed, the proof of which is given in [19, Lemma 2.2].

Proposition 3.2. *The denoising operator $\mathcal{D}^p_\lambda$ is non-expansive, i.e. for any two sequences $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ in $\ell_2(\mathbb{Z})$,*

$$\|\mathcal{D}^p_\lambda(\boldsymbol{v}_1) - \mathcal{D}^p_\lambda(\boldsymbol{v}_2)\| \le \|\boldsymbol{v}_1 - \boldsymbol{v}_2\|.$$

Furthermore, since $\mathcal{T}^p$ is defined via $\mathcal{D}^p_\lambda$, it also satisfies that

$$\|\mathcal{T}^p\mathcal{A}(\boldsymbol{v}_1) - \mathcal{T}^p\mathcal{A}(\boldsymbol{v}_2)\| \le \|\boldsymbol{v}_1 - \boldsymbol{v}_2\|.$$

In particular, we have

$$\|\mathcal{T}^p\mathcal{A}\boldsymbol{v}_1\| \le \|\boldsymbol{v}_1\|.$$

Now we are ready to show the convergence of Algorithm 2.2.

Theorem 3.1. *Let $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the high pass filters of a tight frame system derived by the UEP with $\boldsymbol{h}_0$ being the given low pass filter which satisfies* (2.2)*. Then the sequence $\boldsymbol{v}_n$ defined by* (2.20) *in Algorithm 2.2 converges for arbitrary initial seed $\boldsymbol{v}_0 \in \ell_2(\mathbb{Z})$ to $\boldsymbol{v}^\beta$ which satisfies*

$$\boldsymbol{v}^\beta = \mathcal{A}^* \mathcal{T}^p \mathcal{A}(H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{v}^\beta). \tag{3.1}$$

Proof. The idea of the proof is to show that the sequence $\{\boldsymbol{v}_n\}$ is a Cauchy sequence. We first note that $\|\mathcal{A}^*\| \leq 1$. Let

$$\boldsymbol{v}_n = \mathcal{A}^* \mathcal{T}^p \mathcal{A}(H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{v}_{n-1})$$

and for $m > 0$

$$\boldsymbol{v}_{n+m} = \mathcal{A}^* \mathcal{T}^p \mathcal{A}(H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{v}_{n+m-1}).$$

For convenience, denote

$$\boldsymbol{u} = H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{v}_{n-1}$$

and

$$\boldsymbol{u}' = H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{v}_{n+m-1}.$$

Then using Proposition 3.2 we have:

$$\|\boldsymbol{v}_{n+m} - \boldsymbol{v}_n\| = \|\mathcal{A}^*(\mathcal{T}^p \mathcal{A}\boldsymbol{u}' - \mathcal{T}^p \mathcal{A}\boldsymbol{u})\| \leq \|\mathcal{T}^p \mathcal{A}\boldsymbol{u}' - \mathcal{T}^p \mathcal{A}\boldsymbol{u}\| \leq \|\boldsymbol{u}' - \boldsymbol{u}\| \leq \beta \|\boldsymbol{v}_{n+m-1} - \boldsymbol{v}_{n-1}\|.$$

Inductively, we finally obtain that

$$\|\boldsymbol{v}_{n+m} - \boldsymbol{v}_n\| \leq \beta^n \|\boldsymbol{v}_m - \boldsymbol{v}_0\|. \tag{3.2}$$

Then sequence $\{\boldsymbol{v}_n\}$ is a Cauchy sequence if $\{\boldsymbol{v}_n\}$ is bounded. Since $0 < \beta < 1$, indeed due to Proposition 3.2 we have

$$\|\boldsymbol{v}_n\| = \|\mathcal{A}^* \mathcal{T}^p \mathcal{A}\boldsymbol{u}\| \leq \|\mathcal{T}^p \mathcal{A}\boldsymbol{u}\| \leq \|\boldsymbol{u}\| \leq \beta\|\boldsymbol{c}\| + \beta\|\boldsymbol{v}_{n-1}\| \leq \frac{\beta}{1-\beta}\|\boldsymbol{c}\| + \|\boldsymbol{v}_0\|. \tag{3.3}$$

Hence the limit of the iteration (2.20) exists. The limit $\boldsymbol{v}^\beta$ satisfying $\boldsymbol{v}^\beta = \mathcal{A}^* \mathcal{T}^p \mathcal{A}(H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{v}^\beta)$ follows the continuity of denoising operator $\mathcal{T}^p$ at $\mathbf{0}$ and $\mathcal{T}^p \mathcal{A}(\mathbf{0}) = \mathbf{0}$. □

Here we note that the limit $\boldsymbol{v}^\beta$ of iteration (2.20) satisfies (3.1). Let $\tilde{\boldsymbol{v}}^\beta$ be the sequence $\mathcal{T}^p\mathcal{A}(H_0^*\beta\boldsymbol{c}+\sum_{\ell=1}^r H_\ell^* H_\ell\beta\boldsymbol{v}^\beta)$, then the pair $(\boldsymbol{v}^\beta,\tilde{\boldsymbol{v}}^\beta)$ satisfies $\boldsymbol{v}^\beta=\mathcal{A}^*\tilde{\boldsymbol{v}}^\beta$. As a consequence, the pair $(\boldsymbol{s}^\beta,\tilde{\boldsymbol{s}}^\beta)$ where $\boldsymbol{s}^\beta=\frac{1}{\beta}\boldsymbol{v}^\beta$ and

$$\tilde{\boldsymbol{s}}^\beta=\frac{1}{\beta}\mathcal{T}^p\mathcal{A}(H_0^*\beta\boldsymbol{c}+\sum_{\ell=1}^r H_\ell^* H_\ell\beta\boldsymbol{v}^\beta)$$

also satisfies $\boldsymbol{s}^\beta=\mathcal{A}^*\tilde{\boldsymbol{s}}^\beta$. We will prove in the next subsection that the pair $(\boldsymbol{s}^\beta,\tilde{\boldsymbol{s}}^\beta)$ satisfies the inequality (2.12) up to a small $\varepsilon>0$ when β is close to 1. Hence, $\boldsymbol{s}^\beta$ is the solution of equation (2.6).

A similar proof shows the convergence of Algorithm 2.3, i.e. iteration (2.21) converges, as stated in the following proposition.

Theorem 3.2. *Let $\boldsymbol{h}_1,\ldots,\boldsymbol{h}_r$ be the high pass filters of a tight frame system derived by the UEP with $\boldsymbol{h}_0$ being the given low pass filter which satisfies* (2.2)*. Then the sequence $\boldsymbol{v}_n$ defined by* (2.21) *in Algorithm 2.3 converges for arbitrary initial seed $\boldsymbol{v}_0\in\ell_2(\mathbb{Z})$ to $\boldsymbol{v}^\beta$ which satisfies*

$$\boldsymbol{v}^\beta=H_0^*\beta\boldsymbol{c}+\sum_{\ell=1}^r H_\ell^*\mathcal{A}^*\mathcal{T}^p\mathcal{A}(H_\ell\beta\boldsymbol{v}^\beta). \tag{3.4}$$

3.2 Minimization Property of Algorithm 2.2

In this section, we discuss to what extend that the solution $\boldsymbol{s}^\beta$ obtained from Algorithm 2.2 satisfies (2.12). Without further clarification, $p\in[1,2]$ in the following discussion.

By Algorithm 2.2,

$$\boldsymbol{s}^\beta=\frac{1}{\beta}\boldsymbol{v}^\beta \quad\text{and}\quad \tilde{\boldsymbol{s}}^\beta=\frac{1}{\beta}\tilde{\boldsymbol{v}}^\beta,$$

where $\tilde{\boldsymbol{v}}^\beta=\mathcal{T}^p\mathcal{A}(H_0^*\beta\boldsymbol{c}+\sum_{\ell=1}^r H_\ell^* H_\ell\beta\boldsymbol{v}^\beta)$ and $\boldsymbol{v}^\beta=\mathcal{A}^*\tilde{\boldsymbol{v}}^\beta$ are obtained by the limit of iteration (2.20). First, if $\tilde{\boldsymbol{s}}^\beta\notin\ell_p$, then for any pair $(\boldsymbol{\eta},\tilde{\boldsymbol{\eta}})$ with $\tilde{\boldsymbol{\eta}}=\mathcal{A}\boldsymbol{\eta}\in\ell_p$, the values of both sides in (2.12) are infinite and the inequality holds. For the case $\tilde{\boldsymbol{s}}^\beta\in\ell_p$, what we will prove is a slightly weaker result than (2.12) for the pair $(\boldsymbol{s}^\beta,\tilde{\boldsymbol{s}}^\beta)$ as stated below.

For given constants $C > 0$ and $\varepsilon > 0$, the pair $(\boldsymbol{s}^\beta, \tilde{\boldsymbol{s}}^\beta)$ satisfies the following equality

$$\|\boldsymbol{h}_0 * (\boldsymbol{s}^\beta + \boldsymbol{\eta}) - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0, k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \geq \|\boldsymbol{h}_0 * \boldsymbol{s}^\beta - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0, k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k}|^p - \varepsilon, \tag{3.5}$$

for any pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ satisfying $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta} \in \ell_p$ with $\|\boldsymbol{\eta}\| \leq C$, as long as the acceleration factor β is close enough to 1.

As we will see in next section, when certain boundary condition is imposed in numerical implementations, the solution will satisfy (2.12).

We first prove the following statement: for given constants $C > 0$ and $\varepsilon > 0$, the pair $(\boldsymbol{v}^\beta, \tilde{\boldsymbol{v}}^\beta)$ satisfies the following inequality

$$\begin{aligned} &\|H_0(\boldsymbol{v}^\beta + \boldsymbol{\eta}) - \beta \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0, k\in\mathbb{Z}} \beta^{2-p} \lambda_j |(\tilde{v}^\beta)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p + (1-\beta)^2 \sum_{\ell=1}^{r} \|H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta})\|^2 \\ &\quad \geq \|H_0 \boldsymbol{v}^\beta - \beta \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0, k\in\mathbb{Z}} \beta^{2-p} \lambda_j |(\tilde{v}^\beta)_{\ell,j,k}|^p + (1-\beta)^2 \sum_{\ell=1}^{r} \|H_\ell \boldsymbol{v}^\beta\|^2 - \varepsilon, \end{aligned} \tag{3.6}$$

whenever the pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ satisfying $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta} \in \ell_p$ with $\|\boldsymbol{\eta}\| \leq C$ and the acceleration factor β is close enough to 1. Note that the threshold parameters $\beta^{2-p}\lambda_j$ are less than those in (3.5). It is reasonable because the use of acceleration factor β helps to damp out the noise residing in $\boldsymbol{c}$.

To show (3.6), we introduce the following functionals. For a given pair of sequences $(\boldsymbol{v}, \tilde{\boldsymbol{v}})$ satisfying $\boldsymbol{v} = \mathcal{A}^* \tilde{\boldsymbol{v}}$ and a sequence $\boldsymbol{a}$, define

$$\Phi(\boldsymbol{v}) := \|H_0 \boldsymbol{v} - \beta \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0, k\in\mathbb{Z}} \beta^{2-p} \lambda_j |\tilde{v}_{\ell,j,k}|^p + (1-\beta)^2 \sum_{\ell=1}^{r} \|H_\ell \boldsymbol{v}\|^2. \tag{3.7}$$

and

$$\tilde{\Phi}(\boldsymbol{v}; \boldsymbol{a}) := \|H_0 \boldsymbol{v} - \beta \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0, k\in\mathbb{Z}} \beta^{2-p} \lambda_j |\tilde{v}_{\ell,j,k}|^p + \sum_{\ell=1}^{r} \|H_\ell(\boldsymbol{v} - \beta \boldsymbol{a})\|^2. \tag{3.8}$$

It is clear that when $\boldsymbol{a} = \boldsymbol{v}$, we have $\tilde{\Phi}(\boldsymbol{v}, \boldsymbol{v}) = \Phi(\boldsymbol{v})$. Furthermore, the following result on $\tilde{\Phi}(\boldsymbol{v}, \boldsymbol{a})$ holds.

Proposition 3.3. *Suppose* $\tilde{\boldsymbol{a}} = \mathcal{A}\boldsymbol{a} \in \ell_p$ *and* $\tilde{\boldsymbol{c}} = \mathcal{A}\boldsymbol{c} \in \ell_p$. *Let* $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ *be the high pass filters obtained from* $\boldsymbol{h}_0$ *by the UEP and* $H_0, H_1, \ldots, H_r$ *be the corresponding matrix counterparts of these filters as defined in* (1.18). *Let*

$$\tilde{\boldsymbol{v}}_\beta^\star = \mathcal{T}^p \mathcal{A}(H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{a}) = \mathcal{T}^p \mathcal{A}(\beta \boldsymbol{a} + \beta(H_0^* \boldsymbol{c} - H_0^* H_0 \boldsymbol{a})) \tag{3.9}$$

and $\boldsymbol{v}_\beta^\star = \mathcal{A}^* \tilde{\boldsymbol{v}}_\beta^\star$. *Then the pair* $(\boldsymbol{v}_\beta^\star, \tilde{\boldsymbol{v}}_\beta^\star)$ *satisfies that for any pair* $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ *satisfying* $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta} \in \ell_p$,

$$\tilde{\Phi}(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}; \boldsymbol{a}) \geq \tilde{\Phi}(\boldsymbol{v}_\beta^\star; \boldsymbol{a}) + \|\boldsymbol{\eta}\|^2. \tag{3.10}$$

Proof. The (ℓ, j, k)th entries of sequences $\tilde{\boldsymbol{v}}_\beta^\star$ and $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta}$ are denoted by $(\tilde{v}_\beta^\star)_{\ell,j,k}$ and $\tilde{\eta}_{\ell,j,k}$ respectively.

From the definition of $\tilde{\Phi}(\boldsymbol{v}; \boldsymbol{a})$ by (3.8), we have

$$\begin{aligned}
\tilde{\Phi}(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}; \boldsymbol{a}) &= \|H_0(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}) - \beta \boldsymbol{c}\|^2 + \|\boldsymbol{v}_\beta^\star + \boldsymbol{\eta} - \beta \boldsymbol{a}\|^2 - \|H_0(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}) - \beta H_0 \boldsymbol{a}\|^2 \\
&\quad + \sum_{\ell=1}^{r} \sum_{j<0, k \in \mathbb{Z}} \beta^{2-p} \lambda_j |(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \\
&= \|H_0 \boldsymbol{v}_\beta^\star - \beta \boldsymbol{c}\|^2 + 2\langle H_0 \boldsymbol{\eta}, H_0 \boldsymbol{v}_\beta^\star - \beta \boldsymbol{c} \rangle + \|H_0 \boldsymbol{\eta}\|^2 + \|\boldsymbol{v}_\beta^\star - \beta \boldsymbol{a}\|^2 + \|\boldsymbol{\eta}\|^2 \\
&\quad + 2\langle \boldsymbol{\eta}, \boldsymbol{v}_\beta^\star - \beta \boldsymbol{a} \rangle - \|H_0 \boldsymbol{v}_\beta^\star - \beta H_0 \boldsymbol{a}\|^2 - 2\langle H_0 \boldsymbol{\eta}, H_0 \boldsymbol{v}_\beta^\star - \beta H_0 \boldsymbol{a} \rangle - \|H_0 \boldsymbol{\eta}\|^2 \\
&\quad + \sum_{\ell=1}^{r} \sum_{j<0, k \in \mathbb{Z}} \beta^{2-p} \lambda_j |(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \\
&= \|H_0 \boldsymbol{v}_\beta^\star - \beta \boldsymbol{c}\|^2 + \|\boldsymbol{v}_\beta^\star - \beta \boldsymbol{a}\|^2 - \|H_0 \boldsymbol{v}_\beta^\star - \beta H_0 \boldsymbol{a}\|^2 \\
&\quad + \sum_{\ell=1}^{r} \sum_{j<0, k \in \mathbb{Z}} \beta^{2-p} \lambda_j |(\tilde{v}_\beta^\star)_{\ell,j,k}|^p + \|\boldsymbol{\eta}\|^2 \\
&\quad + 2\langle \boldsymbol{\eta}, H_0^*(H_0 \boldsymbol{v}_\beta^\star - \beta \boldsymbol{c}) \rangle + 2\langle \boldsymbol{\eta}, \boldsymbol{v}_\beta^\star - \beta \boldsymbol{a} \rangle - 2\langle \boldsymbol{\eta}, H_0^*(H_0 \boldsymbol{v}_\beta^\star - \beta H_0 \boldsymbol{a}) \rangle \\
&\quad + \sum_{\ell=1}^{r} \sum_{j<0, k \in \mathbb{Z}} \beta^{2-p} \lambda_j (|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - |(\tilde{v}_\beta^\star)_{\ell,j,k}|^p) \\
&= \tilde{\Phi}(\boldsymbol{v}_\beta^\star; \boldsymbol{a}) + \|\boldsymbol{\eta}\|^2 + 2\langle \boldsymbol{\eta}, \boldsymbol{v}_\beta^\star - \beta \boldsymbol{a} - H_0^* \beta \boldsymbol{c} + H_0^* H_0 \beta \boldsymbol{a} \rangle \\
&\quad + \sum_{\ell=1}^{r} \sum_{j<0, k \in \mathbb{Z}} \beta^{2-p} \lambda_j (|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - |(\tilde{v}_\beta^\star)_{\ell,j,k}|^p).
\end{aligned} \tag{3.11}$$

Since $\mathcal{A}^*\mathcal{A} = I$ by Lemma 1.2, the inner product in (3.11) can be further extended as

$$\begin{aligned}\langle \boldsymbol{\eta}, \boldsymbol{v}_\beta^\star - \beta\boldsymbol{a} - H_0^*\beta\boldsymbol{c} + H_0^*H_0\beta\boldsymbol{a}\rangle &= \langle \boldsymbol{\eta}, \mathcal{A}^*\tilde{\boldsymbol{v}}_\beta^\star - \mathcal{A}^*\mathcal{A}\beta\boldsymbol{a} - \mathcal{A}^*\mathcal{A}H_0^*\beta\boldsymbol{c} + \mathcal{A}^*\mathcal{A}H_0^*H_0\beta\boldsymbol{a}\rangle \\ &= \langle \mathcal{A}\boldsymbol{\eta}, \tilde{\boldsymbol{v}}_\beta^\star - \mathcal{A}\beta\boldsymbol{a} - \mathcal{A}(H_0^*(\beta\boldsymbol{c} - H_0\beta\boldsymbol{a}))\rangle.\end{aligned} \tag{3.12}$$

By this, together with the simplified notation $\sum_{\ell,j,k} := \sum_{\ell=1}^{r}\sum_{j<0}\sum_{k\in\mathbb{Z}}$ and $\lambda_j^\beta := \beta^{2-p}\lambda_j$, $\tilde{\Phi}(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}; \boldsymbol{a})$ becomes:

$$\begin{aligned}\tilde{\Phi}(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}; \boldsymbol{a}) = \tilde{\Phi}(\boldsymbol{v}_\beta^\star; \boldsymbol{a}) &+ \|\boldsymbol{\eta}\|^2 \\ &+ \sum_{\ell,j,k}\lambda_j^\beta(|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - |(\tilde{v}_\beta^\star)_{\ell,j,k}|^p) \\ &+ \sum_{\ell,j,k} 2\tilde{\eta}_{\ell,j,k}((\tilde{v}_\beta^\star)_{\ell,j,k} - \beta a_{\ell,j,k} - (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k}).\end{aligned} \tag{3.13}$$

Next we prove the inequality

$$\Phi(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}; \boldsymbol{a}) \geq \Phi(\boldsymbol{v}_\beta^\star; \boldsymbol{a}) + \|\boldsymbol{\eta}\|^2$$

for $1 \leq p \leq 2$. For this we only need to show the nonnegativity of

$$\sum_{\ell,j,k}\lambda_j^\beta(|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - |(\tilde{v}_\beta^\star)_{\ell,j,k}|^p) + \sum_{\ell,j,k} 2\tilde{\eta}_{\ell,j,k}((\tilde{v}_\beta^\star)_{\ell,j,k} - \beta a_{\ell,j,k} - (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k}). \tag{3.14}$$

Since $\tilde{\boldsymbol{a}}$ and $\tilde{\boldsymbol{c}}$ are in ℓ_p, when $\tilde{\boldsymbol{v}}_\beta^\star \in \ell_p$, by applying the Minkowski's and Young's inequalities as well as the nonexpansive property of the threshold function $t_\lambda^p(x)$, we have $\{(\tilde{v}_\beta^\star)_{\ell,j,k} - \beta a_{\ell,j,k} - (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k}\} \in \ell_p(\mathbb{Z})$. Because $\tilde{\boldsymbol{\eta}} \in \ell_p$ and $q = \frac{p}{p-1} \geq 1$, we have $\tilde{\boldsymbol{\eta}} \in \ell_q(\mathbb{Z})$ and by Hölder inequality,

$$\{\tilde{\eta}_{\ell,j,k}((\tilde{v}_\beta^\star)_{\ell,j,k} - \beta a_{\ell,j,k} - (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k})\} \in \ell_1(\mathbb{Z}).$$

Thus the sequences in (3.14) are absolutely convergent and hence we can prove (3.14) term by term, i.e. we prove

$$\lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - \lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k}|^p + 2\tilde{\eta}_{\ell,j,k}((\tilde{v}_\beta^\star)_{\ell,j,k} - \beta a_{\ell,j,k} - (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k}) \geq 0. \tag{3.15}$$

First we consider the case $p = 1$. The threshold function is the soft-threshold function and the (ℓ, j, k)th entry of $\tilde{\boldsymbol{v}}_\beta^\star$ satisfies that $(\tilde{v}_\beta^\star)_{\ell,j,k} = t_{\lambda_j^\beta}(\beta a_{\ell,j,k} + (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k})$. We show (3.15) case by case.

1. $(\tilde{v}_\beta^\star)_{\ell,j,k} = 0$, then $|\beta a_{\ell,j,k} + (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k}| \leq \lambda_j^\beta/2$.

$$\lambda_j^\beta|\tilde{\eta}_{\ell,j,k}| + 2\tilde{\eta}_{\ell,j,k}(-\beta a_{\ell,j,k} - (\beta H_0^*\boldsymbol{c} - \beta H_0^* H_0\boldsymbol{a})_{\ell,j,k}) \geq \lambda_j^\beta(|\tilde{\eta}_{\ell,j,k}| - \tilde{\eta}_{\ell,j,k}) \geq 0;$$

2. $(\tilde{v}_\beta^\star)_{\ell,j,k} > 0$, then $(\tilde{v}_\beta^\star)_{\ell,j,k} = \beta a_{\ell,j,k} + (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k} - \lambda_j^\beta/2$.

$$\begin{aligned}
&\lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}| - \lambda_j^\beta(\tilde{v}_\beta^\star)_{\ell,j,k} \\
&\qquad\qquad + 2\tilde{\eta}_{\ell,j,k}((\tilde{v}_\beta^\star)_{\ell,j,k} - \beta a_{\ell,j,k} - (\beta H_0^*\boldsymbol{c} - \beta H_0^* H_0\beta\boldsymbol{a})_{\ell,j,k}) \\
&\qquad = \lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}| - \lambda_j^\beta(\tilde{v}_\beta^\star)_{\ell,j,k} + 2\tilde{\eta}_{\ell,j,k}(-\lambda_j^\beta/2) \\
&\qquad = \lambda_j^\beta\left(|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}| - ((\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k})\right) \geq 0;
\end{aligned}$$

3. $(\tilde{\boldsymbol{v}}_\beta^\star)_{\ell,j,k} < 0$, then $(\tilde{\boldsymbol{v}}_\beta^\star)_{\ell,j,k} = \beta\boldsymbol{a}_{\ell,j,k} + (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k} + \lambda_j^\beta/2$.

$$\begin{aligned}
&\lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}| + \lambda_j^\beta(\tilde{v}_\beta^\star)_{\ell,j,k} \\
&\qquad\qquad + 2\tilde{\eta}_{\ell,j,k}((\tilde{v}_\beta^\star)_{\ell,j,k} - \beta a_{\ell,j,k} - (\beta H_0^*\boldsymbol{c} - \beta H_0^* H_0\boldsymbol{a})_{\ell,j,k}) \\
&\qquad = \lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}| + \lambda_j^\beta(\tilde{v}_\beta^\star)_{\ell,j,k} + 2\tilde{\eta}_{\ell,j,k}(\lambda_j^\beta/2) \\
&\qquad = \lambda_j^\beta\left(|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}| + ((\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k})\right) \geq 0.
\end{aligned}$$

Thus when $p = 1$ the sum in (3.13) is nonnegative and hence the inequality holds.

Next we consider the case $1 < p \leq 2$. When $1 < p \leq 2$, the value of $\boldsymbol{v}_\beta^\star$ is given by $(\tilde{v}_\beta^\star)_{\ell,j,k} = (\mathrm{F}^p_{\lambda_j^\beta})^{-1}(\beta a_{\ell,j,k} + (\beta H_0^*(\boldsymbol{c} - H_0\boldsymbol{a}))_{\ell,j,k})$ and we have

$$\begin{aligned}
&\lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - \lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k}|^p + 2\tilde{\eta}_{\ell,j,k}((\tilde{v}_\beta^\star)_{\ell,j,k} - \mathrm{F}^p_{\lambda_j^\beta}((\tilde{v}_\beta^\star)_{\ell,j,k})) \\
&\qquad = \lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - \lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k}|^p - \tilde{\eta}_{\ell,j,k} p\lambda_j \operatorname{sgn}((\tilde{v}_\beta^\star)_{\ell,j,k})|(\tilde{v}_\beta^\star)_{\ell,j,k}|^{p-1}.
\end{aligned}$$

If $(\tilde{v}_\beta^\star)_{\ell,j,k} = 0$, then (3.15) holds clearly. If $(\tilde{v}_\beta^\star)_{\ell,j,k} \neq 0$, we check it using function $\theta(t) = |t|^p$ where $p > 1$. The second order derivative is $\theta''(t) = p(p-1)|t|^{p-2}$, which is nonnegative for any value of t except 0. By Taylor expansion,

$$\lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k}+\tilde{\eta}_{\ell,j,k}|^p-\lambda_j^\beta|(\tilde{v}_\beta^\star)_{\ell,j,k}|^p-2\tilde{\eta}_{\ell,j,k}((\tilde{v}_\beta^\star)_{\ell,j,k}-\mathrm{F}^p_{\lambda_j^\beta}(\tilde{v}^\star_{\ell,j,k})) = \frac{1}{2}\lambda_j^\beta p(p-1)|\xi|^{p-2}\tilde{\eta}^2_{\ell,j,k},$$

where ξ is between $(\tilde{v}^\star_\beta)_{\ell,j,k}$ and $(\tilde{v}^\star_\beta)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}$. Thus (3.15) still holds when $1 < p \le 2$. In conclusion, when $1 \le p \le 2$, we always have (3.15) and hence (3.14). Therefore, the inequality $\tilde{\Phi}(\boldsymbol{v}^\star_\beta + \boldsymbol{\eta}; \boldsymbol{a}) \ge \tilde{\Phi}(\boldsymbol{v}^\star_\beta; \boldsymbol{a}) + \|\boldsymbol{\eta}\|^2$ holds for $1 \le p \le 2$. □

A similar proposition is proved in [19], where the underlying system used in denoising is orthonormal basis and their proof depends on the fact $\mathcal{A}^*\mathcal{A} = \mathcal{A}\mathcal{A}^* = I$. However, for the tight frame system, one only has $\mathcal{A}^*\mathcal{A} = I$ ($\mathcal{A}\mathcal{A}^*$ is not I). This adds the difficulties of the proof and it also leads the conditions on the pairs $(\boldsymbol{v}^\star_\beta, \tilde{\boldsymbol{v}}^\star_\beta)$ and $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$. As we pointed out before, the condition on the pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ is stronger than that on $(\boldsymbol{v}^\star_\beta, \tilde{\boldsymbol{v}}^\star_\beta)$.

To give the minimization property of $(\boldsymbol{v}^\beta, \tilde{\boldsymbol{v}}^\beta)$, we need that $\boldsymbol{v}^\beta$ is uniformly bounded regardless of β. This will be true if we assume the threshold parameters λ_j satisfy that $\inf_\beta \inf_j \lambda_j \ge \gamma > 0$, $j < 0$ and $0 < \beta < 1$. This condition is natural in applications. Indeed, this assumption requires to discard the framelet coefficients when $|j|$ is sufficiently large. It is reasonable because for a given signal, when $|j|$ is large enough, the coefficients of the low frequency subband are very small and can be discarded anyway. We first prove the following lemma:

Lemma 3.1. *Let $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the high pass filters of a tight frame system derived by the UEP with $\boldsymbol{h}_0$ being the given low pass filter. Suppose the threshold parameters $\lambda > 0$, then there exists a constant $0 < \rho < 1$ such that for any sequence $\boldsymbol{v} \in \ell_2(\mathbb{Z})$*

$$\|\mathcal{D}^p_\lambda(\boldsymbol{v})\| \le \rho \|\boldsymbol{v}\|,$$

where $\mathcal{D}^p_\lambda$ is the threshold operator defined in (2.17). Further, let $\mathcal{T}^p\mathcal{A}$ be the denoising operator. Assuming that $\inf_j \lambda_j \ge \gamma > 0$, we have

$$\|\mathcal{T}^p\mathcal{A}(\boldsymbol{v})\| \le \rho \|\boldsymbol{v}\|, \quad 0 < \rho < 1.$$

Proof. By (2.17), we have

$$\|\mathcal{D}^p_\lambda(\boldsymbol{v})\|^2 = \sum_{k\in\mathbb{Z}} |t^p_\lambda(\boldsymbol{v}[k])|^2.$$

When $p = 1$, it is the soft-threshold function $t_\lambda(x) := t^1_\lambda(x) = \operatorname{sgn}(x)\max(|x| - \lambda/2, 0)$. If $\lambda \ge 2\sup_{k\in\mathbb{Z}} |\boldsymbol{v}[k]|$, then $\mathcal{D}^p_\lambda(\boldsymbol{v}) = \boldsymbol{0}$ and hence the inequality $|t_\lambda(\boldsymbol{v}[k])| \le \rho|\boldsymbol{v}[k]|$ holds for any $0 < \rho < 1$. If $\lambda < 2\sup_{k\in\mathbb{Z}} |\boldsymbol{v}[k]|$, then for a given $k \in \mathbb{Z}$, we have

$$\left|\frac{t_\lambda(\boldsymbol{v}[k])}{\boldsymbol{v}[k]}\right| \leq 1 - \frac{\lambda}{2|\boldsymbol{v}[k]|} \leq 1 - \frac{\lambda}{2\|\boldsymbol{v}\|}.$$

Since $\boldsymbol{v} \in \ell_2(\mathbb{Z})$, we have $\rho = \sup_{k\in\mathbb{Z}} \left|\frac{t_\lambda(\boldsymbol{v}[k])}{\boldsymbol{v}[k]}\right| \leq 1 - \frac{\lambda}{2\|\boldsymbol{v}\|} < 1$.

Next, when $1 < p \leq 2$, by (2.18), we have $t_\lambda^p(x) = (F_\lambda^p)^{-1}(x)$ where $F_\lambda^p(x) = x + \frac{p\lambda}{2}\operatorname{sgn}(x)|x|^{p-1}$. For given $\boldsymbol{v}[k]$, $k \in \mathbb{Z}$, assume $(F_\lambda^p)^{-1}(\boldsymbol{v}[k]) \neq 0$. Let $y = (F_\lambda^p)^{-1}(\boldsymbol{v}[k])$. Since $1 < p \leq 2$, we have

$$\left|\frac{(F_\lambda^p)^{-1}(\boldsymbol{v}[k])}{\boldsymbol{v}[k]}\right| = \left|\frac{y}{y + \frac{p\lambda}{2}\operatorname{sgn}(y)|y|^{p-1}}\right| = \frac{1}{1 + \frac{p\lambda}{2}|y|^{p-2}} \leq \frac{1}{1 + \frac{p\lambda}{2}\|\boldsymbol{v}\|^{p-2}} < 1.$$

When $(F_\lambda^p)^{-1}(\boldsymbol{v}[k]) = 0$, it is clear that

$$|(F_\lambda^p)^{-1}(\boldsymbol{v}[k])| \leq \frac{1}{1 + \frac{p\lambda}{2}\|\boldsymbol{v}\|^{p-2}}|\boldsymbol{v}[k]|.$$

Thus when $1 < p \leq 2$, we choose

$$\rho = \sup_{k\in\mathbb{Z}}\left|\frac{(F_{\lambda_j}^p)^{-1}(\boldsymbol{v}[k])}{\boldsymbol{v}[k]}\right| \leq \frac{1}{1 + \frac{p\lambda}{2}\|\boldsymbol{v}\|^{p-2}} < 1.$$

Thus threshold operator $\mathcal{D}_\lambda^p$ satisfies

$$\|\mathcal{D}_\lambda^p(\boldsymbol{v})\| \leq \rho\|\boldsymbol{v}\|, \quad 1 \leq p \leq 2.$$

For the denoising operator, by (2.19),

$$\|\mathcal{T}^p\mathcal{A}(\boldsymbol{v})\|^2 = \sum_{\ell=1}^{r}\sum_{j=-\infty}^{-1}\|\mathcal{D}_{\lambda_j}^p(H_{\ell,j}\prod_{j'=j+1}^{-1}H_{0,j'}\boldsymbol{v})\|^2,$$

then for each sequence $H_{\ell,j}\prod_{j'=j+1}^{-1}H_{0,j'}\boldsymbol{v}$, there exists $\rho_{\ell,j}$ such that

$$\|\mathcal{D}_{\lambda_j}^p(H_{\ell,j}\prod_{j'=j+1}^{-1}H_{0,j'}\boldsymbol{v})\| \leq \rho_{\ell,j}\|H_{\ell,j}\prod_{j'=j+1}^{-1}H_{0,j'}\boldsymbol{v}\|.$$

Since $\|H_{\ell,j}\prod_{j'=j+1}^{-1}H_{0,j'}\boldsymbol{v}\| \leq \|\boldsymbol{v}\|$ and $\inf_j \lambda_j \geq \gamma > 0$, we can take

$$\rho = \sup_{\ell,j}\rho_{\ell,j} \leq \begin{cases} 1 - \frac{\gamma}{2\|\boldsymbol{v}\|} < 1, & \text{when } p = 1; \\ \frac{1}{1+\frac{p\gamma}{2}\|\boldsymbol{v}\|^{p-2}} < 1, & \text{when } 1 < p \leq 2. \end{cases}$$

Thus we have $\|\mathcal{T}^p\mathcal{A}(\boldsymbol{v})\|^2 \leq \sum_{\ell=1}^{r}\sum_{j=-\infty}^{-1}\rho_{\ell,j}^2\|H_{\ell,j}\prod_{j'=j+1}^{-1}H_{0,j'}\boldsymbol{v}\|^2 \leq \rho^2\|\boldsymbol{v}\|^2$, which completes the proof. □

Note that since $\mathcal{D}_\lambda^p$ is not linear, we do not have

$$\|\mathcal{T}^p\mathcal{A}(\boldsymbol{v}_1) - \mathcal{T}^p\mathcal{A}(\boldsymbol{v}_2)\| \le \rho\|\boldsymbol{v}_1 - \boldsymbol{v}_2\|,$$

although we have

$$\|\mathcal{T}^p\mathcal{A}(\boldsymbol{v}_1) - \mathcal{T}^p\mathcal{A}(\boldsymbol{v}_2)\| \le \|\boldsymbol{v}_1 - \boldsymbol{v}_2\|$$

by Proposition 3.2.

Based on Lemma 3.1, we can derive that the iterative sequence is uniformly bounded. More precisely, we have the following proposition.

Proposition 3.4. *Let $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the high pass filters of a tight frame system derived by the UEP with $\boldsymbol{h}_0$ being the given low pass filter and $\boldsymbol{v}^\beta$ be the limit of iteration* (2.20) *for $0 < \beta < 1$. Assume that the the threshold parameters λ_j, $j < 0$ are independent of iteration with $\inf_\beta \inf_j \lambda_j \ge \gamma > 0$. Then there exists $C > 0$, such that $\|\boldsymbol{v}^\beta\| \le C$, for all $0 < \beta < 1$.*

Proof. For any given initial value $\boldsymbol{v}_0 \in \ell_2(\mathbb{Z})$ and a fixed $\beta \in (0,1)$, let $\{\boldsymbol{v}_n^\beta\}$ be the sequence obtained by iteration (2.20) in Algorithm 2.2. Applying Lemma 3.1 and the argument used in (3.3) lead to

$$\|\boldsymbol{v}_n^\beta\| \le \rho(\|\boldsymbol{c}\| + \|\boldsymbol{v}_{n-1}^\beta\|) \le \frac{\rho}{1-\rho}\|\boldsymbol{c}\| + \|\boldsymbol{v}_0\|.$$

Let $C = \frac{\rho}{1-\rho}\|\boldsymbol{c}\| + \|\boldsymbol{v}_0\|$, then $\|\boldsymbol{v}_n^\beta\| \le C$. Hence, the limit $\boldsymbol{v}^\beta$ to $\boldsymbol{v}_n^\beta$ also satisfies that $\|\boldsymbol{v}^\beta\| \le C$. ☐

A consequence of Proposition 3.3 is the following result which states that the minimization property of $\boldsymbol{v}^\beta$.

Proposition 3.5. *Suppose $\tilde{\boldsymbol{c}} = \mathcal{A}\boldsymbol{c} \in \ell_p$. For given $\varepsilon > 0$ and $C > \sup_\beta \|\boldsymbol{v}^\beta\|$, there exists $\delta > 0$, which only depends on ε and C, such that for all $\beta \in (1-\delta, 1)$, the corresponding limit $(\boldsymbol{v}^\beta, \tilde{\boldsymbol{v}}^\beta)$ of iteration* (2.20) *in Algorithm 2.2 satisfies the inequality* (3.6) *for an arbitrary pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ satisfying $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta} \in \ell_p$ and $\|\boldsymbol{\eta}\| \le C$.*

Proof. Note inequality (3.6) for an arbitrary β is equivalent to

$$\Phi(\boldsymbol{v}^\beta + \boldsymbol{\eta}) \geq \Phi(\boldsymbol{v}^\beta) - \varepsilon$$

for all $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$, satisfying $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta}$ and $\|\boldsymbol{\eta}\| \leq C$.

Applying Proposition 3.3 by letting $\boldsymbol{a} = \boldsymbol{v}^\beta$, we have inequality

$$\tilde{\Phi}(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}; \boldsymbol{v}^\beta) \geq \tilde{\Phi}(\boldsymbol{v}_\beta^\star; \boldsymbol{v}^\beta) + \|\boldsymbol{\eta}\|^2 \tag{3.16}$$

for any pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ satisfying $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta}$. Since the limit $\boldsymbol{v}^\beta$ satisfies (3.1) by Theorem 3.1 and $\boldsymbol{v}_\beta^\star$ is given by (3.9) in Proposition 3.3, we have

$$\boldsymbol{v}_\beta^\star = \mathcal{A}^*\mathcal{T}^p\mathcal{A}(H_0^*\beta\boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \beta \boldsymbol{v}^\beta) = \boldsymbol{v}^\beta.$$

Hence $\tilde{\Phi}(\boldsymbol{v}_\beta^\star; \boldsymbol{v}^\beta) = \tilde{\Phi}(\boldsymbol{v}^\beta; \boldsymbol{v}^\beta) = \Phi(\boldsymbol{v}^\beta)$. By the definition of $\tilde{\Phi}(\boldsymbol{v}; \boldsymbol{a})$, one obtains that

$$\begin{aligned}\tilde{\Phi}(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}; \boldsymbol{v}^\beta) &= \|H_0(\boldsymbol{v}^\beta + \boldsymbol{\eta}) - \beta\boldsymbol{c}\|^2 + \sum_{\ell=1}^{r}\sum_{j<0,k\in\mathbb{Z}} \beta^{2-p}\lambda_j |(\tilde{v}^\beta)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \\ &\quad + \sum_{\ell=1}^{r} \|(1-\beta)H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta}) + \beta H_\ell \boldsymbol{\eta}\|^2.\end{aligned}$$

Since

$$\begin{aligned}\|(1-\beta)H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta}) + \beta H_\ell\boldsymbol{\eta}\|^2 &\leq (1-\beta)^2\|H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta})\|^2 + \beta^2\|H_\ell\boldsymbol{\eta}\|^2 \\ &\quad + 2\beta(1-\beta)\|H_\ell\boldsymbol{\eta}\|\|H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta})\|,\end{aligned}$$

this leads to

$$\begin{aligned}\tilde{\Phi}(\boldsymbol{v}_\beta^\star + \boldsymbol{\eta}; \boldsymbol{v}^\beta) \leq{}& \|H_0(\boldsymbol{v}^\beta + \boldsymbol{\eta}) - \beta\boldsymbol{c}\|^2 + \sum_{\ell=1}^{r}\sum_{j<0,k\in\mathbb{Z}} \beta^{2-p}\lambda_j |(\tilde{v}^\beta)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \\ &+ (1-\beta)^2\sum_{\ell=1}^{r}\|H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta})\|^2 + \beta^2\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|^2 \\ &+ 2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\|H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta})\|.\end{aligned}$$

Note that

$$\begin{aligned}\Phi(\boldsymbol{v}^\beta + \boldsymbol{\eta}) ={}& \|H_0(\boldsymbol{v}^\beta + \boldsymbol{\eta}) - \beta\boldsymbol{c}\|^2 + \sum_{\ell=1}^{r}\sum_{j<0,k\in\mathbb{Z}} \beta^{2-p}\lambda_j |(\tilde{v}^\beta)_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \\ &+ (1-\beta)^2\sum_{\ell=1}^{r}\|H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta})\|^2.\end{aligned}$$

So we have

$$\Phi(\boldsymbol{v}^\beta+\boldsymbol{\eta})+\beta^2\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|^2+2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\|H_\ell(\boldsymbol{v}^\beta+\boldsymbol{\eta})\|\geq\Phi(\boldsymbol{v}^\beta)+\|\boldsymbol{\eta}\|^2.$$

This leads to the following equality

$$\Phi(\boldsymbol{v}^\beta+\boldsymbol{\eta})\geq\Phi(\boldsymbol{v}^\beta)+\|\boldsymbol{\eta}\|^2-\beta^2\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|^2-2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\|H_\ell(\boldsymbol{v}^\beta+\boldsymbol{\eta})\|.$$

Using

$$\|\boldsymbol{\eta}\|^2=\|H_0\boldsymbol{\eta}\|^2+\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|^2,$$

one obtains

$$\begin{aligned}
&\|\boldsymbol{\eta}\|^2-\beta^2\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|^2-2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\|H_\ell(\boldsymbol{v}^\beta+\boldsymbol{\eta})\|\\
&\quad=\|H_0\boldsymbol{\eta}\|^2+(1-\beta^2)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|^2-2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\|H_\ell(\boldsymbol{v}^\beta+\boldsymbol{\eta})\|\\
&\quad=\|H_0\boldsymbol{\eta}\|^2+\sum_{\ell=1}^{r}(1-\beta)\|H_\ell\boldsymbol{\eta}\|\Big((1+\beta)\|H_\ell\boldsymbol{\eta}\|-2\beta\|H_\ell(\boldsymbol{v}^\beta+\boldsymbol{\eta})\|\Big)\\
&\quad\geq\|H_0\boldsymbol{\eta}\|^2+2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\Big(\|H_\ell\boldsymbol{\eta}\|-\|H_\ell(\boldsymbol{v}^\beta+\boldsymbol{\eta})\|\Big)\\
&\quad\geq\|H_0\boldsymbol{\eta}\|^2+2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\Big(\|H_\ell\boldsymbol{\eta}\|-\|H_\ell\boldsymbol{v}^\beta\|-\|H_\ell\boldsymbol{\eta}\|\Big)\\
&\quad=\|H_0\boldsymbol{\eta}\|^2-2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\|H_\ell\boldsymbol{v}^\beta\|.
\end{aligned}$$

This leads to

$$\Phi(\boldsymbol{v}^\beta+\boldsymbol{\eta})\geq\Phi(\boldsymbol{v}^\beta)+\|H_0\boldsymbol{\eta}\|^2-2\beta(1-\beta)\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\|H_\ell\boldsymbol{v}^\beta\|. \tag{3.17}$$

Because $\boldsymbol{v}^\beta$ is bounded by Lemma 3.1 and $\boldsymbol{\eta}$ is also bounded, the term $\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{\eta}\|\|H_\ell\boldsymbol{v}^\beta\|$ is bounded by rC^2. So given arbitrary $\varepsilon>0$, we can take $\delta\leq\frac{\varepsilon}{2rC^2}$ and then for any $\beta\in(1-\delta,1)$,

$$\Phi(\boldsymbol{v}^\beta+\boldsymbol{\eta})\geq\Phi(\boldsymbol{v}^\beta)+\|H_0\boldsymbol{\eta}\|^2-\varepsilon\geq\Phi(\boldsymbol{v}^\beta)-\varepsilon, \tag{3.18}$$

which completes the proof. □

Based on the minimization of $\boldsymbol{v}^\beta$, the minimization property of $\boldsymbol{s}^\beta$ is straightforward. It is given by the following theorem.

Theorem 3.3. *Suppose* $\tilde{\boldsymbol{c}} \in \ell_p$. *For given* $\varepsilon > 0$ *and* $C > \sup_\beta \|\boldsymbol{v}^\beta\|$, *there exists* $\delta > 0$, *which only depends on* ε *and* C, *such that for all* $\beta \in (1-\delta, 1)$, *the solution* $(\boldsymbol{s}^\beta, \tilde{\boldsymbol{s}}^\beta)$ *of Algorithm 2.2 satisfies the inequality* (3.5) *for any pair* $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ *satisfying* $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta} \in \ell_p$ *and* $\|\boldsymbol{\eta}\| \leq C$.

Proof. For given $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$, set $\boldsymbol{\eta}_1 = \beta\boldsymbol{\eta}$ and $\tilde{\boldsymbol{\eta}}_1 = \beta\tilde{\boldsymbol{\eta}}$ which satisfy that $\|\boldsymbol{\eta}_1\| \leq C$ and $\tilde{\boldsymbol{\eta}}_1 \in \ell_p$. Then, for arbitrary $\varepsilon > 0$, applying Proposition 3.5, there exists $\delta_1 > 0$ such that for any $\beta \in (1-\delta_1, 1)$ the pair $(\boldsymbol{v}^\beta, \tilde{\boldsymbol{v}}^\beta)$ satisfies

$$\Phi(\boldsymbol{v}^\beta + \boldsymbol{\eta}_1) \geq \Phi(\boldsymbol{v}^\beta) - \frac{\varepsilon}{8}, \tag{3.19}$$

as long as $(\boldsymbol{\eta}_1, \tilde{\boldsymbol{\eta}}_1)$ satisfies $\tilde{\boldsymbol{\eta}}_1 = \mathcal{A}\boldsymbol{\eta}_1 \in \ell_p$ and $\|\boldsymbol{\eta}_1\| \leq C$. From Algorithm 2.2, we have $\boldsymbol{s}^\beta = \frac{\boldsymbol{v}^\beta}{\beta}$ and $\tilde{\boldsymbol{s}}^\beta = \frac{\tilde{\boldsymbol{v}}^\beta}{\beta}$. Dividing β^2 on both sides of (3.19), we have

$$\begin{aligned}
\|H_0(\boldsymbol{s}^\beta + \boldsymbol{\eta}) - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k} + \frac{(\tilde{\eta}_1)_{\ell,j,k}}{\beta}|^p + \frac{(1-\beta)^2}{\beta^2} \sum_{\ell=1}^{r} \|H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta}_1)\|^2 \\
\geq \|H_0\boldsymbol{s}^\beta - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k}|^p + \frac{(1-\beta)^2}{\beta^2} \sum_{\ell=1}^{r} \|H_\ell\boldsymbol{v}^\beta\|^2 - \frac{\varepsilon}{8\beta^2}.
\end{aligned} \tag{3.20}$$

Because $\boldsymbol{v}^\beta$ and $\boldsymbol{\eta}_1$ are bounded, for any given $\varepsilon > 0$, we can take $\delta_2 \leq \frac{1}{4C}\sqrt{\frac{\varepsilon}{r}}$ and then any $\beta \in (1-\delta_2, 1)$ satisfies $(1-\beta)^2 \sum_{\ell=1}^r (\|H_\ell(\boldsymbol{v}^\beta + \boldsymbol{\eta}_1)\|^2 - \|H_\ell \boldsymbol{v}^\beta\|^2) < \frac{\varepsilon}{8}$. Taking $\delta = \min(\delta_1, \delta_2, \frac{1}{2})$ and combining with (3.20), the pair $(\boldsymbol{s}^\beta, \tilde{\boldsymbol{s}}^\beta)$ satisfies for any $\beta \in (1-\delta, 1)$

$$\begin{aligned}
\|H_0(\boldsymbol{s}^\beta + \boldsymbol{\eta}) - \boldsymbol{c}\|^2 &+ \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \\
&\geq \|H_0\boldsymbol{s}^\beta - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k}|^p - \frac{\varepsilon}{4\beta^2} \\
&\geq \|H_0\boldsymbol{s}^\beta - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k}|^p - \varepsilon,
\end{aligned}$$

as long as the pair $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}})$ satisfies $\tilde{\boldsymbol{\eta}} = \mathcal{A}\boldsymbol{\eta} \in \ell_p$ and $\|\boldsymbol{\eta}\| \leq C$. □

Remark 3.1. We note that since each iteration solution pair $(\boldsymbol{v}_n^\beta, \tilde{\boldsymbol{v}}_n^\beta)$ satisfy $\boldsymbol{v}_n^\beta \in \ell_2(\mathbb{Z})$ and $\tilde{\boldsymbol{v}}_n^\beta \in \ell_2$, and $(\boldsymbol{v}^\beta, \tilde{\boldsymbol{v}}^\beta)$ is the limit to the iteration pair, it leads to $\boldsymbol{v}^\beta \in \ell_2(\mathbb{Z})$ and $\tilde{\boldsymbol{v}}^\beta \in \ell_2$, and furthermore $\boldsymbol{s}^\beta \in \ell_2(\mathbb{Z})$ and $\tilde{\boldsymbol{s}}^\beta \in \ell_2$. The minimization property (3.5) holds with finite value on both sides whenever $p = 2$. For $1 \le p < 2$, as we have proved, when $\tilde{\boldsymbol{s}}^\beta$ is an ℓ_p, $1 \le p < 2$ sequence, the solution satisfies the minimization inequality (3.5). In fact, the values on the both sides of inequality (3.5) are finite.

In the proof of Theorem 3.3 (See (3.20)), when β is chosen to be small (say smaller than 1/2) instead of closing to 1, we have

$$\|H_0(\boldsymbol{s}^\beta + \boldsymbol{\eta}) - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p + \lambda \sum_{\ell=1}^{r} \|H_\ell(\boldsymbol{s}^\beta + \boldsymbol{\eta})\|^2$$
$$\ge \|H_0\boldsymbol{s}^\beta - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r} \sum_{j<0,k\in\mathbb{Z}} \lambda_j |\tilde{s}^\beta_{\ell,j,k}|^p + \lambda \sum_{\ell=1}^{r} \|H_\ell \boldsymbol{s}^\beta\|^2 - \varepsilon.$$

In this case, in addition to penalize the functional in (2.12) we also penalize

$$\sum_{\ell=1}^{r} \|H_\ell \boldsymbol{s}\|^2, \tag{3.21}$$

the high frequency information of the solution. However, as we discussed in the formulation, since the deconvolution processing is essentially to recover the term $\sum_{\ell=1}^r H_\ell \boldsymbol{s}$, we do not want to penalize (3.21). This motivates us to suggest that β to be chosen close to 1, although smaller β will give a fast convergence rate. Our numerical simulation also shows that when smaller β is chosen, the corresponding solution is over smoothed. This leads to inefficient deconvolution. We summarize the numerical results in Table 3.1 where the filters in Example 1.2 are used and the original signal is given in Figure 4.1 (a).

	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.4$	$\beta = 0.5$	$\beta = 0.6$	$\beta = 0.7$	$\beta = 0.8$	$\beta = 0.9$
RE	0.0643	0.0633	0.0622	0.0611	0.0599	0.0585	0.0571	0.0555	0.0537
PSNR	31.1299	31.2889	31.4560	31.6336	31.8439	32.0849	32.3537	32.6583	32.9811

Table 3.1. Numerical results of Algorithm 2.2 when β changes from 0.1 to 0.9

As can be seen, when β is small, the algorithm only removes the noise from the data but does not deconvolve the signal significantly. When β becomes close to 1, relative error becomes smaller and peak signal-to-noise ratio is much better. These numerical data coincide with our analysis because smaller β penalizes more the high frequency components which are needed to be recovered from the algorithms.

In practice, we are given the data of finite dimension. As will see in §4, we can make the finite dimensional matrix $H_0^* H_0$ nonsingular and hence the iteration (2.20) will converges without the acceleration factor β. In such a case, we can directly prove inequality (2.12).

3.3 Minimization Property of Algorithm 2.3

In this section, we discuss the minimization property of the solution $\boldsymbol{s}^\beta$ obtained in Algorithm 2.3. We use the similar approach to that used in the last section.

We characterize the minimization property of solution $\boldsymbol{s}^\beta$ paralleled to that of Algorithm 2.2. From the iteration (2.21), we obtain the limit $\boldsymbol{v}^\beta$ which satisfies

$$\boldsymbol{v}^\beta = H_0^* \beta \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* \mathcal{A}^* \mathcal{T}^p \mathcal{A}(H_\ell \beta \boldsymbol{v}^\beta). \tag{3.22}$$

Define

$$\tilde{\boldsymbol{v}}_\ell^\beta = \mathcal{T}^p \mathcal{A}(H_\ell \beta \boldsymbol{v}^\beta) \quad \text{and} \quad \boldsymbol{v}_\ell^\beta = \mathcal{A}^* \tilde{\boldsymbol{v}}_\ell^\beta, \quad \ell = 1, \ldots, r. \tag{3.23}$$

If we further denote $\beta \boldsymbol{c}$ by $\boldsymbol{v}_0^\beta$, the limit of iteration (2.21) satisfies $\boldsymbol{v}^\beta = \mathcal{A}_{-1}^* \{\boldsymbol{v}_\ell^\beta\}_{\ell=0}^r$ where $\mathcal{A}_{-1}^*$ is given by (1.25). We denote the quantities that determine the limit $\boldsymbol{v}^\beta$ by the $(r+1)$-tuple be $(\boldsymbol{v}^\beta, \tilde{\boldsymbol{v}}_1^\beta, \ldots, \tilde{\boldsymbol{v}}_r^\beta)$.

The solution of Algorithm 2.3 is given by another $(r+1)$-tuple $(\boldsymbol{s}^\beta, \tilde{\boldsymbol{s}}_1^\beta, \ldots, \tilde{\boldsymbol{s}}_r^\beta)$ with

$$\tilde{\boldsymbol{s}}_\ell^\beta = \frac{\tilde{\boldsymbol{v}}_\ell^\beta}{\beta} \quad \text{and} \quad \boldsymbol{s}_\ell^\beta = \mathcal{A}^* \tilde{\boldsymbol{s}}_\ell^\beta, \quad \ell = 1, \ldots, r. \tag{3.24}$$

Since $\boldsymbol{v}^\beta$ satisfies (3.22), we have

$$\boldsymbol{s}^\beta = \mathcal{A}_{-1}^* \{\boldsymbol{s}_\ell^\beta\}_{\ell=0}^r = H_0^* \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* \mathcal{A}^* \tilde{\boldsymbol{s}}_\ell^\beta, \tag{3.25}$$

where $\boldsymbol{s}_0^\beta := \boldsymbol{c}$ and $\boldsymbol{v}^\beta$, $\tilde{\boldsymbol{v}}_\ell^\beta$ are given in (3.22) and (3.23). In the following, we denote the (ℓ', j, k)th entries in $\tilde{\boldsymbol{s}}_\ell^\beta$, $\ell = 1, \ldots, r$, by $(\tilde{s}_\ell^\beta)_{\ell', j, k}$ where $\ell' = 1, \ldots, r$, $j < 0$ and $k \in \mathbb{Z}$.

The solution of Algorithm 2.3 has different minimization property from the solution of Algorithm 2.2. Given any $\varepsilon > 0$ and $C > 0$, the $(r+1)$-tuple $(\boldsymbol{s}^\beta, \tilde{\boldsymbol{s}}_1^\beta, \ldots, \tilde{\boldsymbol{s}}_r^\beta)$ satisfies the following inequality

$$\begin{aligned}\|\boldsymbol{h}_0 * (\boldsymbol{s}^\beta + \boldsymbol{\eta}) - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j<0,k\in\mathbb{Z}} \lambda_j |(\tilde{s}_\ell^\beta)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p \\ \geq \|\boldsymbol{h}_0 * \boldsymbol{s}^\beta - \boldsymbol{c}\|^2 + \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j<0,k\in\mathbb{Z}} \lambda_j |(\tilde{s}_\ell^\beta)_{\ell',j,k}|^p - \varepsilon.\end{aligned} \tag{3.26}$$

for any $(r+1)$-tuple $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}_1, \ldots, \tilde{\boldsymbol{\eta}}_r)$ satisfying $\tilde{\boldsymbol{\eta}}_\ell = \mathcal{A}(H_\ell \boldsymbol{\eta})$, $\tilde{\boldsymbol{\eta}}_1, \ldots, \tilde{\boldsymbol{\eta}}_r \in \ell_p$ and $\|\boldsymbol{\eta}\| \leq C$. Note that $\tilde{\boldsymbol{\eta}}_\ell = \mathcal{A}(H_\ell \boldsymbol{\eta})$ implies that $H_\ell \boldsymbol{\eta} = \mathcal{A}^*\mathcal{A}(H_\ell \boldsymbol{\eta}) = \mathcal{A}^*\tilde{\boldsymbol{\eta}}_\ell$ for $\ell = 1, \ldots, r$. The high frequency components $H_\ell \boldsymbol{\eta}$, $\ell = 1, \ldots, r$, are further decomposed by decomposition operator $\mathcal{A}$. More precisely, $\mathcal{A}(H_\ell \boldsymbol{\eta})$ is the coefficients of framelet packet in canonical form (see [5] and [32]). From the penalty terms in (3.26), we can also see that the terms $\tilde{\boldsymbol{s}}_\ell^\beta$, $\ell = 1, \ldots, r$ are no longer framelet coefficients in (3.5) but coefficients of framelet packet decomposition of the high frequency component $\boldsymbol{s}_\ell^\beta = \mathcal{A}^*\tilde{\boldsymbol{s}}_\ell^\beta$, $\ell = 1, \ldots r$, which also reflect certain smoothness of the underlying functions. It is nature to penalize the ℓ_p-norm of framelet packet coefficients of each high frequency component $H_\ell \boldsymbol{s}^\beta$, $\ell = 1, \ldots r$, since as pointed out in the formulation that the deconvolution is essentially to put back the missing components $H_\ell \boldsymbol{s}^\beta$, $\ell = 1, \ldots r$ and we do not want them too rough. In fact, we can put it into a similar formulation as Algorithm 2.2 in terms of the framelet packets. However, we omit the details.

As we did for Algorithm 2.2, we can derive the following result on the minimization property of $(r+1)$-tuple $(\boldsymbol{s}^\beta, \tilde{\boldsymbol{s}}_1^\beta, \ldots, \tilde{\boldsymbol{s}}_r^\beta)$. Since the proof is similar to that of Theorem 3.3, and since we will give a full proof of this result for the finite data set, we omit it here.

Theorem 3.4. *For given $\varepsilon > 0$ and $C > \sup_\beta \|\boldsymbol{v}^\beta\|$, there exists $\delta > 0$, which only depends on ε and C, such that for all $\beta \in (1-\delta, 1)$, the corresponding $(r+1)$-tuple $(\boldsymbol{s}^\beta, \tilde{\boldsymbol{s}}_1^\beta, \ldots, \tilde{\boldsymbol{s}}_r^\beta)$ of iteration (2.21) in Algorithm 2.3 satisfies inequality (3.26) for any $(r+1)$-tuple $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}_1, \ldots, \tilde{\boldsymbol{\eta}}_r)$ satisfying $\tilde{\boldsymbol{\eta}}_\ell = \mathcal{A}(H_\ell \boldsymbol{\eta})$, $\tilde{\boldsymbol{\eta}}_1, \ldots, \tilde{\boldsymbol{\eta}}_r \in \ell_p$ and $\|\boldsymbol{\eta}\| \leq C$.*

Chapter 4

Deconvolution of Finite Data Set

In the previous sections, our algorithms and analysis are given for the infinite data set which is of theoretic interests and connects to multiresolution analysis. However, in application, given data sets are always finite, a vector in e.g. $\mathbb{R}^{N_0}$. Thus it is necessary to adjust our approach for these cases. This is achieved by extrapolating the data out of the boundary. The numerical simulation shows that the algorithms work well under different boundary conditions as shown in [8, 10, 11].

4.1 Algorithms for Finite Data

In this section, we convert the algorithms given in previous chapters to the ones which deal with the finite data. The convolution equation becomes

$$\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{b} + \boldsymbol{\epsilon} = \boldsymbol{c}$$

with the finite given data set $\boldsymbol{c}$ and $\|\boldsymbol{\epsilon}\|_2 = \varepsilon < \infty$ where $\|\cdot\|_2$ is the spectral norm of vector or matrix. Since our data are no longer infinite, the boundary conditions are needed to extend the data beyond their original domain. Basically, there are three types of boundary conditions: zero-padding, periodic and symmetric. Since the zero-padding boundary condition simply adds zeros out of the original domain, it is more or less reduced to the case discussed in the previous section and it normally gives boundary artifacts, we omit discussions on this case. We focus on more detailed discussions on periodic boundary condition and the discussion of symmetric boundary condition can be carried out similarly.

When the given data $\boldsymbol{\alpha}$ is extended using the periodic boundary condition, i.e.

$$\boldsymbol{\alpha}[n] = \boldsymbol{\alpha}[n \bmod N_0], \quad n \in \mathbb{Z},$$

where N_0 is the length of data $\boldsymbol{\alpha}$, the convolution of data $\boldsymbol{\alpha}$ with given filter $\boldsymbol{h}_0$ then becomes a special kind of convolution, *circular convolution.* We denote such circular convolution by

$$\boldsymbol{h}_0 * \boldsymbol{\alpha} := \boldsymbol{h}_0 \circledast \boldsymbol{\alpha}.$$

The circular convolution can also be written as a matrix-vector multiplication where the matrix is a circulant matrix, a special kind of Toeplitz matrix, i.e. the entries of matrix H_0 generated from $\boldsymbol{h}_0$ are

$$H_0[l,k] = \boldsymbol{h}_0[(l-k) \bmod N_0], \quad 0 \le l,k < N_0. \tag{4.1}$$

Using periodic boundary condition to extend data implies that matrices H_0, $H_1, \ldots, H_r$ used in convolution are now circulant matrices of finite order generated from the filters $\boldsymbol{h}_0, \boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$. Further, we have dilated filters $\boldsymbol{h}_{0,j}, \ldots, \boldsymbol{h}_{r,j}$ for the jth level decomposition, where $\boldsymbol{h}_{\ell,j}$ is obtained by inserting $2^{-j-1}-1$ zeros between every two entries in $\boldsymbol{h}_\ell$. With these, we define the discrete decomposition and reconstruction operators A_J and A_J^* analog to (1.23) and (1.25) for a finite J by

$$A_J = [\Big(\prod_{j=J}^{-1} H_{0,j}\Big),\ \Big(H_{1,J}\prod_{j=J+1}^{-1} H_{0,j}\Big), \ldots, \Big(H_{r,J}\prod_{j=J+1}^{-1} H_{0,j}\Big), \ldots, H_1, \ldots, H_r]^t \tag{4.2}$$

and

$$A_J^* = [\Big(\prod_{j=-1}^{J} H_{0,j}^*\Big),\ \Big(\prod_{j=-1}^{J+1} H_{0,j}^* H_{1,J}^*\Big), \ldots, \Big(\prod_{j=-1}^{J+1} H_{0,j}^* H_{r,J}^*\Big), \ldots, H_1^*, \ldots, H_r^*]. \tag{4.3}$$

They are essentially block matrices with circulant blocks. Each entry in A_J and A_J^* is the product of a series of circulant matrices $H_{\ell,j}$ generated from filter $\boldsymbol{h}_{\ell,j}$. Similar to Proposition 1.2, it can be proved that A_J^* is the adjoint operator of A_J and $A_J^* A_J = I$, where I is the identity matrix.

The finiteness of data makes it possible to remove the acceleration factor from the iterations in both Algorithm 2.2 and 2.3. From the proof of the convergence of Algorithm 2.2 and 2.3, to remove the acceleration factor, we need that the largest eigenvalues of the matrix

$$\sum_{\ell=1}^{r} H_\ell^* H_\ell$$

are less than 1. Since

$$\sum_{\ell=1}^{r} H_\ell^* H_\ell = I - H_0^* H_0,$$

the convergence of the iteration depends on the nonsingularity of matrix $H_0^* H_0$. We note that in the case when the data are infinite dimension as discussed in §3, the spectrum of corresponding operator H_0 contains zero. Hence, we have to use the acceleration factor in this case.

To implement our algorithms on the data of finite dimension, we decompose to a finite level to denoise. Hence in the iteration, operators A_J and A_J^* are used instead of $\mathcal{A}$ and $\mathcal{A}^*$. Moreover, we need in the algorithms the following denoising operator for data of finite dimension: given a finite sequence $\boldsymbol{v}$, define

$$\mathcal{T}^p A_J(\boldsymbol{v}) = [(\prod_{j=J}^{-1} H_{0,j}\boldsymbol{v}),\ \mathcal{D}^p_{\lambda_J}(H_{1,J} \prod_{j=J+1}^{-1} H_{0,j}\boldsymbol{v}), \ldots, \mathcal{D}^p_{\lambda_J}(H_{r,J} \prod_{j=J+1}^{-1} H_{0,j}\boldsymbol{v}),$$
$$\ldots, \mathcal{D}^p_{\lambda_{-1}}(H_1\boldsymbol{v}), \ldots, \mathcal{D}^p_{\lambda_{-1}}(H_r\boldsymbol{v})]^t,$$

where the threshold operator $\mathcal{D}^p_\lambda$ is given in (2.17). With these notations, we can give out the algorithms for finite data set.

Algorithm 4.1 (Algorithm 2.2 for finite data)**.**

(i) *Choose an initial approximation* $\boldsymbol{v}_0$ *(e.g.* $\boldsymbol{v}_0 = \boldsymbol{c}$*);*

(ii) *Iterate on* n *until convergence:*

$$\boldsymbol{v}_{n+1} = A_J^* \mathcal{T}^p A_J(H_0^* \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{v}_n). \tag{4.4}$$

Algorithm 4.2 (Algorithm 2.3 for finite data)**.**

(i) *Choose an initial approximation* $\boldsymbol{v}_0$ *(e.g.* $\boldsymbol{v}_0 = \boldsymbol{c}$*);*

(ii) *Iterate on n until convergence:*

$$\boldsymbol{v}_{n+1} = H_0^* \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* (A_J^* \mathcal{T}^p A_J)(H_\ell \boldsymbol{v}_n). \tag{4.5}$$

(iii) *Let* $\boldsymbol{v}_{n_0}$ *be the final iterative solution from (ii). Then the solution to the algorithm is*

$$\boldsymbol{v} = A_J^* \mathcal{T}^p A_J (\boldsymbol{v}_{n_0}).$$

As we can see the difference between Algorithm 4.1 and 4.2 is the different denoising schemes used in the iteration. The above algorithms can be understood as Algorithm 2.2 and 2.3 being applied to finite data. The underlying framelet analysis can also be carried out by using the framelets on intervals, e.g. periodic framelets when the periodic boundary conditions are imposed. We omit the discussion here. On the other hand, the above algorithms can be also viewed as algorithms to solve the equation:

$$H_0 \boldsymbol{v} = \boldsymbol{b} + \boldsymbol{\epsilon} = \boldsymbol{c}, \tag{4.6}$$

where H_0 is the matrix depends on the boundary condition imposed, e.g. H_0 is a circulant matrix generated by $\boldsymbol{h}_0$ when the periodic boundary conditions are imposed. We note that we can always make H_0 to be nonsingular by Proposition 4.1 given in the next section and hence the linear system always has a unique solution.

4.2 Convergence and Minimization Properties

In this section, we analyze the convergence of the algorithms given in §4.1 and discuss the minimization properties of the corresponding solutions.

The analysis is based on the nonsingularity of the matrix H_0. We consider the finite dimension data with periodic boundary condition which leads H_0 to be circulant. The eigenvalues of the circulant matrix H_0 generated from $\boldsymbol{h}_0$ can be given out explicitly as follows:

$$\lambda_p[H_0] = \sum_{k=0}^{K-1} \boldsymbol{h}_0[k] \exp\left(-\frac{i2kp\pi}{N_0}\right), \quad p = 0, 1, \cdots, N_0 - 1, \tag{4.7}$$

where N_0 is the length of given data and K is the length of filter $\boldsymbol{h}_0$. Here we assume, without loss of generality, $K < N_0$. To see why the eigenvalues of matrix H_0 are in form of (4.7), we need to notice that the data set is extended periodically and the finite filter $\boldsymbol{h}_0$ is extended to be an infinite one $\widetilde{\boldsymbol{h}_0}$ by adding zeros beyond the finite length, i.e.

$$\widetilde{\boldsymbol{h}_0} = [\cdots, 0, \cdots, 0, \boldsymbol{h}_0[0], \cdots, \boldsymbol{h}_0[K-1], 0, \cdots, 0, \cdots].$$

Since the eigenvalues of circulant matrix H_0 generated by $\boldsymbol{h}_0$ are

$$\lambda_p[H_0] = \sum_{k=0}^{N_0-1} \widetilde{\boldsymbol{h}_0}[k] \exp\left(-\frac{i2kp\pi}{N_0}\right), \quad p = 0, 1, \cdots, N_0 - 1,$$

removing the zero entries from the above expression we get the eigenvalues of H_0 which are the same as (4.7).

Furthermore, we can see that the eigenvalues of the matrix H_0 are the values of polynomial $\widehat{\boldsymbol{h}_0}(\omega)$ at $\omega = \frac{2p\pi}{N_0}$, $p = 0, \ldots, N_0 - 1$. The matrix H_0 is nonsingular, whenever $\widehat{\boldsymbol{h}_0}(\frac{2p\pi}{N_0})$ is not equal to zero for each $p = 0, \ldots, N_0 - 1$. Since $\widehat{\boldsymbol{h}_0}$ only has finitely many zeros, we can extend the data set to increase the length of the data from N_0 to N_1 (before making a periodical extension of the data) to avoid the zero eigenvalue of H_0. This observation is summarized in the following result.

Proposition 4.1. *Let $\boldsymbol{h}_0$ be the given low pass filter with length K and the given data having length $N_0 > K$. Then the data set can always be extended to have the length $N_1 > N_0$ such that the corresponding circulant matrix H_0 generated from $\boldsymbol{h}_0$ with the data set of length N_1 is nonsingular. Consequently, the matrices H_0^* and $H_0^* H_0$ are nonsingular.*

Proof. We start the proof from the explicit form of the eigenvalues of the circulant matrix H_0 generated from filter $\boldsymbol{h}_0$ with the data of length N. The eigenvalues of the N-by-N

circulant matrix H_0 are:

$$\lambda_p[H_0] = \sum_{k=0}^{K-1} \boldsymbol{h}_0[k] \exp\left(-\frac{i2\pi kp}{N}\right) = \widehat{\boldsymbol{h}_0}(\frac{2p\pi}{N}), \quad p = 0, 1, \cdots, N-1.$$

If $\widehat{\boldsymbol{h}_0}(2\pi\omega) \neq 0$, for $\omega \in \mathbb{Q}$ with $\mathbb{Q}$ the set of rational numbers, then $\lambda_p[H_0] \neq 0$ for $p = 0, 1, \ldots, N-1$. Since $\boldsymbol{h}_0$ is finitely supported, the polynomial $\widehat{\boldsymbol{h}_0}(\omega)$ has finitely many zeros. Suppose those zeros of $\widehat{\boldsymbol{h}_0}(\omega)$ in terms of rational multiples of 2π are

$$\left\{2\pi\frac{q_i}{p_i} : i = 1, 2, \cdots, n\right\}, \tag{4.8}$$

where for each i, $\gcd(q_i, p_i) = 1$. Because $\widehat{\boldsymbol{h}_0}(\omega)$ is 2π-periodic, we can take the rationales being proper fractions, i.e. $q_i < p_i$. It is not necessary to consider the case $p_i = q_i$ since $\widehat{\boldsymbol{h}_0}(2\pi) = \widehat{\boldsymbol{h}_0}(0) = 1$. To make the matrix H_0 nonsingular, the value of N should satisfy

$$\frac{p}{N} \notin \left\{\frac{q_i}{p_i} : i = 1, 2, \cdots, n\right\}, \; p = 0, 1, \cdots, N-1. \tag{4.9}$$

One sufficient condition on N such that (4.9) holds is

$$p_i \nmid N, \quad i = 1, 2, \cdots, n. \tag{4.10}$$

This is because, if (4.9) is not true, i.e. there exist p_{i_0} and q_{i_0} in the set given in (4.9) such that $\frac{p}{N} = \frac{q_{i_0}}{p_{i_0}}$, then $pp_{i_0} = Nq_{i_0}$. Since $\gcd(p_{i_0}, q_{i_0}) = 1$, it leads to $p_{i_0}|N$, which is a contradiction of $p_{i_0} \nmid N$. Hence, for a given filter $\boldsymbol{h}_0$, there are infinitely many N such that as long as the data length N satisfies (4.10), the corresponding circulant matrix H_0 is nonsingular. For a given data with length N_0, if N_0 dose not satisfies (4.9), we just simply extend the data to the the length N_1 satisfying (4.9). For example, one can take N_1 prime to each p_i. Then the circulant matrix H_0 generated from $\boldsymbol{h}_0$ with respect to the extended data of length N_1 is nonsingular. Since $\det(H_0^*) = \det(H_0)$ and $\det(H_0^* H_0) = \det(H_0)^2$, the matrices H_0^* and $H_0^* H_0$ are nonsingular once H_0 is. □

Remark 4.1. In fact, the processing is constructive once all the zeros in terms of rational multiples of 2π as those in (4.8) are available. Based on the sufficient condition (4.10), we first factorize p_i, $i = 1, \cdots, n$, in (4.8) as:

$$\begin{cases} p_1 & = & m_1^{e_{11}} m_2^{e_{12}} \cdots m_l^{e_{1l}}; \\ p_2 & = & m_1^{e_{21}} m_2^{e_{22}} \cdots m_l^{e_{2l}}; \\ \vdots & & \vdots \\ p_n & = & m_1^{e_{n1}} m_2^{e_{n2}} \cdots m_l^{e_{nl}}, \end{cases}$$

where m_i, $i = 1, \ldots, l$ are prime numbers. Then $N_1 \geq N_0$ satisfying (4.10) means that

$$m_i \nmid N_1, \; i = 1, \cdots, l.$$

Starting from above criterion, we can find the minimum N_1 by directly computation, e.g. using sieve of Eratosthenes. The value of N_1 generated in such a way is smaller than taking N_1 prime to each p_i.

After we calculate the value of N_1, we need to extend the data by $N_1 - N_0$ entries. To make the extension meaningful, a possible way is to repeat the entries in the original data set. For instance, we can append the first $N_1 - N_0$ entries to the end of the data set. If $\boldsymbol{h}_0$ is a refinement mask of a spline, pseudo-spline or one of those used in high resolution image reconstructions, then $N_1 - N_0 \leq 1$, since $\widehat{\boldsymbol{h}_0}(2\omega\pi) = 0$ with $\omega \in \mathbb{Q}$ only when $\omega = \frac{1}{2}$. Thus, as long as N_0 is odd, the corresponding circulant matrix H_0 is nonsingular. This implies that we can simply append at most the first entry in the data set to guarantee the nonsingularity of H_0.

As shown in the proof and remark above, the extension results in a small difference between the number N_0 and N_1. In fact, for many cases, whenever the length of the data is odd, the corresponding circulant matrix H_0 is nonsingular. In the following, we assume that the length of data is N_1 such that the corresponding circulant matrix is nonsingular. As we will see, the nonsingularity of H_0 ensures the convergence of iterations without using the acceleration factor β. Furthermore, the threshold parameters λ_j no longer need to satisfy the additional condition $\inf_j \lambda_j > 0$ imposed in last section.

The convergence of iteration (4.4) in Algorithm 4.1 and iteration (4.5) in Algorithm 4.2 can be proved based on the nonsingularity of the circulant matrix H_0.

Theorem 4.1. *Let $\boldsymbol{h}_0$ be the low pass filter in the convolution equation and $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the high pass filters generated from $\boldsymbol{h}_0$ via the UEP. The corresponding circulant matrices are $H_0, \ldots, H_r$ with $H_0^* H_0$ being nonsingular. Then iteration* (4.4) *in Algorithm 4.1 converges for any initial seed $\boldsymbol{v}_0$ and the limit satisfies*

$$\boldsymbol{s} = A_J^* \mathcal{T}^p A_J (H_0^* \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{s}). \tag{4.11}$$

Similarly, iteration (4.5) *in Algorithm 4.2 converges for any initial seed $\boldsymbol{v}_0$ and the limit satisfies*

$$\boldsymbol{s} = H_0^* \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* A_J^* \mathcal{T}^p A_J (H_\ell \boldsymbol{s}). \tag{4.12}$$

Proof. Because $H_0^* H_0$ is nonsingular and $I = H_0^* H_0 + \sum_{\ell=1}^{r} H_\ell^* H_\ell$, the eigenvalues of $I - H_0^* H_0$ are strictly less than 1, i.e. there exists a constant $\mu < 1$ such that $\| \sum_{\ell=1}^{r} H_\ell^* H_\ell \|_2 = \|I - H_0^* H_0\|_2 \leq \mu$.

Following the proof of Theorem 3.1, given any $\boldsymbol{v}_0$, for any positive integers m and n,

$$\|\boldsymbol{v}_{n+m} - \boldsymbol{v}_n\|_2 \leq \| \sum_{\ell=1}^{r} H_\ell^* H_\ell \|_2 \|\boldsymbol{v}_{n+m-1} - \boldsymbol{v}_{n-1}\|_2 \leq \mu \|\boldsymbol{v}_{n+m-1} - \boldsymbol{v}_{n-1}\|_2 \leq \mu^n \|\boldsymbol{v}_m - \boldsymbol{v}_0\|_2.$$

and

$$\|\boldsymbol{v}_n\|_2 \leq \|\boldsymbol{c}\|_2 + \mu \|\boldsymbol{v}_{n-1}\|_2 \leq \frac{1}{1-\mu} \|\boldsymbol{c}\|_2 + \|\boldsymbol{v}_0\|_2.$$

Thus the iteration sequence $\{\boldsymbol{v}_n\}$ is a Cauchy sequence and the limit exists and satisfies (4.11).

Next we prove the convergence of iteration (4.5). Let $H = [H_1, \ldots, H_r]^t$, then we have

$$\|H\|_2^2 = \max_{\|\boldsymbol{u}\|_2 = 1} \|H \boldsymbol{u}\|_2^2 = \max_{\|\boldsymbol{u}\|_2 = 1} \boldsymbol{u}^* \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{u} = \| \sum_{\ell=1}^{r} H_\ell^* H_\ell \|_2 \leq \mu.$$

Denote $\boldsymbol{g}_{(v,v')} = [(\mathcal{T}^p A_J H_1 \boldsymbol{v} - \mathcal{T}^p A_J H_1 \boldsymbol{v}'), \ldots, (\mathcal{T}^p A_J H_r \boldsymbol{v} - \mathcal{T}^p A_J H_r \boldsymbol{v}')]^t$ for any two vectors $\boldsymbol{v}$ and $\boldsymbol{v}'$. Following the proof of Theorem 3.1, given any $\boldsymbol{v}_0$, for any positive integers m and n,

$$\begin{aligned}
\|\boldsymbol{v}_{n+m}-\boldsymbol{v}_n\|_2 &= \|\sum_{\ell=1}^{r} H_\ell^* A_J^* (\mathcal{T}^p A_J H_\ell \boldsymbol{v}_{n+m-1} - \mathcal{T}^p A_J H_\ell \boldsymbol{v}_{n-1})\|_2 \\
&= \|H^* A_J^* \boldsymbol{g}_{(\boldsymbol{v}_{n+m-1},\boldsymbol{v}_{n-1})}\|_2 \\
&\le \|H^*\|_2 \|\boldsymbol{g}_{(\boldsymbol{v}_{n+m-1},\boldsymbol{v}_{n-1})}\|_2 \\
&\le \|H^*\|_2 \|H\|_2 \|\boldsymbol{v}_{n+m-1}-\boldsymbol{v}_{n-1}\|_2 \\
&\le \mu \|\boldsymbol{v}_{n+m-1}-\boldsymbol{v}_{n-1}\|_2.
\end{aligned}$$

Similarly, one can prove by using (4.5)

$$\|\boldsymbol{v}_n\|_2 \le \|\boldsymbol{c}\|_2 + \mu\|\boldsymbol{v}_{n-1}\|_2 \le \frac{1}{1-\mu}\|\boldsymbol{c}\|_2 + \|\boldsymbol{v}_0\|_2. \tag{4.13}$$

Thus the iteration sequence $\{\boldsymbol{v}_n\}$ is a Cauchy sequence and the limit exists and satisfies (4.12). □

Paralleled to the minimization properties given in §3.2 and §3.3, we have the following results about the minimization properties of the limits to iterations (4.4) and (4.5).

Theorem 4.2. *Let $\boldsymbol{h}_1,\dots,\boldsymbol{h}_r$ be the high pass filters obtained from $\boldsymbol{h}_0$ by the UEP and $H_0, H_1, \dots, H_r$ be the corresponding circulant matrices of these filters. Let $\tilde{\boldsymbol{s}} = \mathcal{T}^p A_J(H_0^*\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* H_\ell \boldsymbol{s})$ where $\boldsymbol{s}$ is the limit of iteration (4.4) satisfying $\boldsymbol{s} = A_J^*\tilde{\boldsymbol{s}}$. Then given fixed $1 \le p \le 2$ the pair $(\boldsymbol{s},\tilde{\boldsymbol{s}})$ satisfies for any pair $(\boldsymbol{\eta},\tilde{\boldsymbol{\eta}})$ with $\tilde{\boldsymbol{\eta}} = A_J\boldsymbol{\eta}$,*

$$\begin{aligned}
\|H_0(\boldsymbol{s}+\boldsymbol{\eta})-\boldsymbol{c}\|_2^2 &+ \sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1}\lambda_j|\tilde{s}_{\ell,j,k}+\tilde{\eta}_{\ell,j,k}|^p \\
&\ge \|H_0\boldsymbol{s}-\boldsymbol{c}\|_2^2 + \sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1}\lambda_j|\tilde{s}_{\ell,j,k}|^p.
\end{aligned}$$

Proof. We prove this theorem by the method used in the proof of Proposition 3.3. First we define the following two functionals of finite sequences. For arbitrary sequence $\boldsymbol{a}$ and any given $(\boldsymbol{v},\tilde{\boldsymbol{v}})$ satisfying $\boldsymbol{v} = A_J^*\tilde{\boldsymbol{v}}$, define

$$\Phi^{\text{finite}}(\boldsymbol{v}) := \|H_0\boldsymbol{v}-\boldsymbol{c}\|_2^2 + \sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1}\lambda_j|\tilde{v}_{\ell,j,k}|^p$$

and

$$\tilde{\Phi}^{\text{finite}}(\boldsymbol{v};\boldsymbol{a}) := \|H_0\boldsymbol{v}-\boldsymbol{c}\|_2^2 + \sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1}\lambda_j|\tilde{v}_{\ell,j,k}|^p + \sum_{\ell=1}^{r}\|H_\ell\boldsymbol{v}-H_\ell\boldsymbol{a}\|_2^2.$$

In Φ^{finite} and $\tilde{\Phi}^{\text{finite}}$, the coefficients of framelet decomposition are used. Moreover, if we take $\boldsymbol{a}=\boldsymbol{v}$, then $\tilde{\Phi}^{\text{finite}}(\boldsymbol{v};\boldsymbol{v}) = \Phi_1^{\text{finite}}(\boldsymbol{v})$.

Note that the statement we need to prove in Theorem 4.2 is equivalent to inequality of $\Phi^{\text{finite}}(\boldsymbol{s}+\boldsymbol{\eta}) \geq \Phi^{\text{finite}}(\boldsymbol{s})$ for any $(\boldsymbol{\eta},\tilde{\boldsymbol{\eta}})$ satisfying $\tilde{\boldsymbol{\eta}} = A_J\boldsymbol{\eta}$. We prove this inequality of Φ^{finite} via functional $\tilde{\Phi}^{\text{finite}}$. Given arbitrary sequence $\boldsymbol{a}$, let $\tilde{\boldsymbol{v}}^\star = \mathcal{T}^p A_J(H_0^*\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* H_\ell \boldsymbol{a})$ and $\boldsymbol{v}^\star = A_J^*\tilde{\boldsymbol{v}}^\star$. We will show that for any $(\boldsymbol{\eta},\tilde{\boldsymbol{\eta}})$ where $\tilde{\boldsymbol{\eta}} = A_J\boldsymbol{\eta}$, $(\boldsymbol{v}^\star,\tilde{\boldsymbol{v}}^\star)$ satisfies

$$\tilde{\Phi}^{\text{finite}}(\boldsymbol{v}^\star+\boldsymbol{\eta};\boldsymbol{a}) \geq \tilde{\Phi}^{\text{finite}}(\boldsymbol{v}^\star;\boldsymbol{a}) + \|\boldsymbol{\eta}\|^2. \tag{4.14}$$

Taking $\boldsymbol{a} = \boldsymbol{s}$ with $\boldsymbol{s}$ the limit of iteration (4.5), we have $\boldsymbol{v}^\star = A_J^*\mathcal{T}^p A_J(H_0^*\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* H_\ell \boldsymbol{s}) = \boldsymbol{s}$ and (4.14) implies that

$$\Phi^{\text{finite}}(\boldsymbol{s}+\boldsymbol{\eta}) + \|\boldsymbol{\eta}\|_2^2 - \|H_0\boldsymbol{\eta}\|_2^2 = \tilde{\Phi}^{\text{finite}}(\boldsymbol{s}+\boldsymbol{\eta};\boldsymbol{s}) \geq \tilde{\Phi}^{\text{finite}}(\boldsymbol{s};\boldsymbol{s}) + \|\boldsymbol{\eta}\|_2^2 = \Phi^{\text{finite}}(\boldsymbol{s}) + \|\boldsymbol{\eta}\|_2^2,$$

which leads to the minimization property of $\boldsymbol{s}$. In the following we show that (4.14) holds for arbitrary $\boldsymbol{a}$.

Given $(\boldsymbol{\eta},\tilde{\boldsymbol{\eta}})$, we have:

$$\begin{aligned}\tilde{\Phi}^{\text{finite}}(\boldsymbol{v}^\star+\boldsymbol{\eta};\boldsymbol{a}) &= \tilde{\Phi}^{\text{finite}}(\boldsymbol{v}^\star;\boldsymbol{a}) + \|\boldsymbol{\eta}\|_2^2 \\ &\quad + \sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1}\lambda_j\left(|\tilde{v}^\star_{\ell,j,k}+\tilde{\eta}_{\ell,j,k}|^p - |\tilde{v}^\star_{\ell,j,k}|^p\right) \\ &\quad + 2\langle\boldsymbol{\eta}, \boldsymbol{v}^\star - H_0^*\boldsymbol{c} - \sum_{\ell=1}^{r} H_\ell^* H_\ell\boldsymbol{a}\rangle,\end{aligned} \tag{4.15}$$

where $\tilde{\boldsymbol{\eta}} = A_J\boldsymbol{\eta}$. From the definition of $\boldsymbol{v}^\star$ and $A_J^*A_J = I$, the inner product can be written as

$$\begin{aligned}\langle\boldsymbol{\eta}, \boldsymbol{v}^\star - H_0^*\boldsymbol{c} - \sum_{\ell=1}^{r} H_\ell^* H_\ell\boldsymbol{a}\rangle &= \langle\boldsymbol{\eta}, A_J^*\tilde{\boldsymbol{v}}^\star - A_J^*A_J\boldsymbol{a} - A_J^*A_J(H_0^*(\boldsymbol{c}-H_0^*H_0\boldsymbol{a}))\rangle \\ &= \langle A_J\boldsymbol{\eta}, \tilde{\boldsymbol{v}}^\star - A_J\boldsymbol{a} - A_J(H_0^*(\boldsymbol{c}-H_0^*H_0\boldsymbol{a}))\rangle.\end{aligned} \tag{4.16}$$

Then replacing the inner product in (4.15) by (4.16), we have

$$\begin{aligned}\tilde{\Phi}^{\text{finite}}(\boldsymbol{v}^\star + \boldsymbol{\eta}; \boldsymbol{a}) &= \tilde{\Phi}^{\text{finite}}(\boldsymbol{v}^\star; \boldsymbol{a}) + \|\boldsymbol{\eta}\|_2^2 \\ &+ \sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j \left(|\tilde{v}^\star_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - |\tilde{v}^\star_{\ell,j,k}|^p\right) \\ &+ 2\langle A_J\boldsymbol{\eta}, \tilde{\boldsymbol{v}}^\star - A_J\boldsymbol{a} - A_J(H_0^*(\boldsymbol{c} - H_0^* H_0 \boldsymbol{a}))\rangle.\end{aligned} \tag{4.17}$$

Comparing (4.17) and (4.14), we only need to show

$$\sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j \left(|\tilde{v}^\star_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - |\tilde{v}^\star_{\ell,j,k}|^p\right) + 2\langle A_J\boldsymbol{\eta}, \tilde{\boldsymbol{v}}^\star - A_J\boldsymbol{a} - A_J(H_0^*(\boldsymbol{c} - H_0^* H_0 \boldsymbol{a}))\rangle$$

is nonnegative to prove inequality (4.14). We further set the simplified notation $\sum_{\ell,j,k} := \sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1}$ and expand the inner product (4.21) using the definition of A_J and the denoising operator $\mathcal{T}^p$ for finite case. Then we finally obtain that

$$\sum_{\ell,j,k} \lambda_j \left(|\tilde{v}^\star_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - |\tilde{v}^\star_{\ell,j,k}|^p\right) + 2\sum_{\ell,j,k} \tilde{\eta}_{\ell,j,k} \left(\tilde{v}^\star_{\ell,j,k} - a_{\ell,j,k} - (H_0^*(\boldsymbol{c} - H_0^* H_0 \boldsymbol{a}))_{\ell,j,k}\right). \tag{4.18}$$

The remaining part is to check the nonnegativity of

$$\lambda_j \left(|\tilde{v}^\star_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p - |\tilde{v}^\star_{\ell,j,k}|^p\right) + 2\tilde{\eta}_{\ell,j,k} \left(\tilde{v}^\star_{\ell,j,k} - a_{\ell,j,k} - (H_0^*(\boldsymbol{c} - H_0^* H_0 \boldsymbol{a}))_{\ell,j,k}\right),$$

which follows the same way as that in the proof of Proposition 3.3. □

Theorem 4.3. *Let $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the high pass filters obtained from $\boldsymbol{h}_0$ by the UEP and $H_0, H_1, \ldots, H_r$ be the corresponding circulant matrices of these filters. Let $\tilde{\boldsymbol{s}}_\ell = \mathcal{T}^p A_J(H_\ell \boldsymbol{s})$ for $\ell = 1, \ldots, r$, where $\boldsymbol{s}$ is the limit of iteration (4.5) satisfying $\boldsymbol{s} = A_{-1}^*\{\boldsymbol{s}_\ell\}_{\ell=0}^r$ with $\boldsymbol{s}_\ell = A_J^* \tilde{\boldsymbol{s}}_\ell$ for $\ell = 1, \ldots, r$ and $\boldsymbol{s}_0 = \boldsymbol{c}$. Then given fixed $1 \le p \le 2$, the $(r+1)$-tuple $(\boldsymbol{s}, \tilde{\boldsymbol{s}}_1, \ldots, \tilde{\boldsymbol{s}}_r)$ satisfies the following inequality*

$$\begin{aligned}\|H_0(\boldsymbol{s} + \boldsymbol{\eta}) - \boldsymbol{c}\|_2^2 &+ \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j |(\tilde{s}_\ell)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p \\ &\ge \|H_0\boldsymbol{s} - \boldsymbol{c}\|_2^2 + \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j |(\tilde{s}_\ell)_{\ell',j,k}|^p,\end{aligned}$$

for any $(r+1)$-tuple $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}_1, \ldots, \tilde{\boldsymbol{\eta}}_r)$ with $\tilde{\boldsymbol{\eta}}_\ell = A_J(H_\ell \boldsymbol{\eta})$ for $\ell = 1, \ldots, r$.

Proof. We prove this theorem by proving a more general inequality. First for given sequence $\boldsymbol{a}$, we define $\tilde{\boldsymbol{v}}_\ell^\star = \mathcal{T}^p A_J(H_\ell \boldsymbol{a})$ and $\boldsymbol{v}_\ell^\star = A_J^*(\tilde{\boldsymbol{v}}_\ell^\star)$ for $\ell = 1, \ldots, r$. Denote $\boldsymbol{c}$ by $\boldsymbol{v}_0^\star$ for the simplicity, then the $(r+1)$-tuple $(\boldsymbol{v}^\star, \tilde{\boldsymbol{v}}_1^\star, \ldots, \tilde{\boldsymbol{v}}_r^\star)$ satisfies

$$\boldsymbol{v}^\star = A_{-1}^*\{\boldsymbol{v}_\ell^\star\}_{\ell=0}^r = H_0^*\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* A_J^* \mathcal{T}^p A_J(H_\ell \boldsymbol{a}).$$

We will then show below that the inequality

$$\begin{aligned} \|H_0(\boldsymbol{v}^\star + \boldsymbol{\eta}) - \boldsymbol{c}\|_2^2 + \sum_{\ell=1}^r \sum_{\ell'=1}^r \sum_{j=J}^{-1} \sum_{k=0}^{N_1-1} \lambda_j |(\tilde{v}_\ell^\star)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p \\ + \sum_{\ell=1}^r \|H_\ell(\boldsymbol{v}^\star + \boldsymbol{\eta}) - H_\ell \boldsymbol{a}\|_2^2 \\ \geq \|H_0 \boldsymbol{v}^\star - \boldsymbol{c}\|_2^2 + \sum_{\ell=1}^r \sum_{\ell'=1}^r \sum_{j=J}^{-1} \sum_{k=0}^{N_1-1} \lambda_j |(\tilde{v}_\ell^\star)_{\ell',j,k}|^p \\ + \sum_{\ell=1}^r \|H_\ell \boldsymbol{v}^\star - H_\ell \boldsymbol{a}\|_2^2 + \|\boldsymbol{\eta}\|_2^2, \end{aligned} \tag{4.19}$$

holds for any $(r+1)$-tuple $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}_1, \ldots, \tilde{\boldsymbol{\eta}}_r)$ with $\tilde{\boldsymbol{\eta}}_\ell = A_J(H_\ell \boldsymbol{\eta})$ for $\ell = 1, \ldots, r$. Note that if we take $\boldsymbol{a} = \boldsymbol{s}$, where $\boldsymbol{s}$ is the limit to iteration (4.5) satisfying (4.12), then

$$\boldsymbol{v}^\star = H_0^*\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* A_J^* \mathcal{T}^p A_J(H_\ell \boldsymbol{s}) = \boldsymbol{s},$$

and (4.19) becomes the inequality we need to prove in the theorem. In the following we give the proof of (4.19), which is similar to that of Proposition 3.3.

Given the $(r+1)$-tuple $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}_1, \ldots, \tilde{\boldsymbol{\eta}}_r)$, we expand the left hand side of (4.19) as follows:

$$\begin{aligned}
\|H_0(\boldsymbol{v}^\star + \boldsymbol{\eta}) - \boldsymbol{c}\|_2^2 &+ \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j |(\tilde{v}_\ell^\star)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p \\
&+ \sum_{\ell=1}^{r} \|H_\ell(\boldsymbol{v}^\star + \boldsymbol{\eta}) - H_\ell \boldsymbol{a}\|_2^2 \\
&= \|H_0 \boldsymbol{v}^\star - \boldsymbol{c}\|_2^2 + \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j |(\tilde{v}_\ell^\star)_{\ell',j,k}|^p \\
&+ \sum_{\ell=1}^{r} \|H_\ell \boldsymbol{v}^\star - H_\ell \boldsymbol{a}\|_2^2 + \|\boldsymbol{\eta}\|_2^2 \\
&+ \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j \left(|(\tilde{v}_\ell^\star)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p - |(\tilde{v}_\ell^\star)_{\ell',j,k}|^p\right) \\
&+ 2\langle \boldsymbol{\eta}, \boldsymbol{v}^\star - H_0^* \boldsymbol{c} - \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{a}\rangle.
\end{aligned} \tag{4.20}$$

Compared with (4.19), we only need to show

$$\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j \left(|(\tilde{v}_\ell^\star)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p - |(\tilde{v}_\ell^\star)_{\ell',j,k}|^p\right) + 2\langle \boldsymbol{\eta}, \boldsymbol{v}^\star - H_0^* \boldsymbol{c} - \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{a}\rangle \geq 0.$$

From the definition of $\boldsymbol{v}^\star$ and $A_J^* A_J = I$, the inner product in (4.20) can be written as

$$\begin{aligned}
\langle \boldsymbol{\eta}, \boldsymbol{v}^\star - H_0^* \boldsymbol{c} - \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{a}\rangle &= \langle \boldsymbol{\eta}, \sum_{\ell=1}^{r} H_\ell^* (A_J^* \mathcal{T}^p A_J(H_\ell \boldsymbol{a}) - H_\ell \boldsymbol{a})\rangle \\
&= \sum_{\ell=1}^{r} \langle A_J(H_\ell \boldsymbol{\eta}), \mathcal{T}^p A_J(H_\ell \boldsymbol{a}) - A_J(H_\ell \boldsymbol{a})\rangle \\
&= \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} (\tilde{\eta}_\ell)_{\ell',j,k}((\tilde{v}_\ell^\star)_{\ell',j,k} - (a_\ell)_{\ell',j,k}).
\end{aligned} \tag{4.21}$$

With this, we only need to show

$$\begin{aligned}
&\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} \lambda_j \left(|(\tilde{v}_\ell^\star)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p - |(\tilde{\boldsymbol{v}}_\ell^\star)_{\ell',j,k}|^p\right) \\
&\quad + 2\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} (\tilde{\eta}_\ell)_{\ell',j,k}((\tilde{v}_\ell^\star)_{\ell',j,k} - (a_\ell)_{\ell',j,k}) \geq 0.
\end{aligned} \tag{4.22}$$

This is proved by showing term by term the nonnegativity of the following sum:

$$\lambda_j \left(|(\tilde{v}_\ell^\star)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p - |(\tilde{v}_\ell^\star)_{\ell',j,k}|^p\right) + 2(\tilde{\eta}_\ell)_{\ell',j,k}((\tilde{v}_\ell^\star)_{\ell',j,k} - (a_\ell)_{\ell',j,k}).$$

We do not need to consider the convergence of the above series since it is only a finite sum. The proof is same as that in the proof of Proposition 3.3 by using the definition of threshold function $t_\lambda^p(x)$ with respect to different value of p, which we omit here. □

When the symmetric boundary condition is used, the filter $\boldsymbol{h}_0$ also needs to be symmetric. The matrix form of the convolution equation generated from $\boldsymbol{h}_0$ according to symmetric boundary condition is of Toeplitz plus pseudo-Hankel type or Toeplitz plus Hankel type, where the (l,k)th entry in a Hankel matrix depends only on the $(l+k)$th entry in the generation sequence. More precisely, when the low pass filter $\boldsymbol{h}_0$ is whole point symmetric, i.e.

$$\boldsymbol{h}_0 = [\boldsymbol{h}_0[n], \ldots, \boldsymbol{h}_0[1], \boldsymbol{h}_0[0], \boldsymbol{h}_0[1], \ldots, \boldsymbol{h}_0[n]], \quad K = 2n+1,$$

the corresponding convolution matrix is of Toeplitz plus pseudo-Hankel type:

$$H_0 = \begin{pmatrix} \boldsymbol{h}_0[0] & \boldsymbol{h}_0[1] & \ldots & \boldsymbol{h}_0[n] & 0 & \ldots & 0 & 0 \\ \boldsymbol{h}_0[1] & \boldsymbol{h}_0[0] & \ldots & \boldsymbol{h}_0[n-1] & \boldsymbol{h}_0[n] & \ldots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \ldots & 0 & 0 & \ldots & \boldsymbol{h}_0[0] & \boldsymbol{h}_0[1] \\ 0 & 0 & \ldots & 0 & 0 & \ldots & \boldsymbol{h}_0[1] & \boldsymbol{h}_0[0] \end{pmatrix}$$

$$+ \begin{pmatrix} 0 & \boldsymbol{h}_0[1] & \ldots & \boldsymbol{h}_0[n-1] & \boldsymbol{h}_0[n] & 0 & \ldots & 0 & 0 & 0 \\ 0 & \boldsymbol{h}_0[2] & \ldots & \boldsymbol{h}_0[n] & 0 & 0 & \ldots & 0 & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & 0 & 0 & 0 & \ldots & \boldsymbol{h}_0[3] & \boldsymbol{h}_0[2] & 0 \\ 0 & 0 & \ldots & 0 & 0 & 0 & \ldots & \boldsymbol{h}_0[2] & \boldsymbol{h}_0[1] & 0 \end{pmatrix};$$

when the low pass filter $\boldsymbol{h}_0$ is half point symmetric, i.e.

$$\boldsymbol{h}_0 = [\boldsymbol{h}_0[n-1], \ldots, \boldsymbol{h}_0[1], \boldsymbol{h}_0[0], \boldsymbol{h}_0[0], \boldsymbol{h}_0[1], \ldots, \boldsymbol{h}_0[n-1]], \quad K = 2n,$$

the corresponding matrix is of Toeplitz plus Hankel type:

$$H_0 = \begin{pmatrix} \boldsymbol{h}_0[0] & \boldsymbol{h}_0[1] & \dots & \boldsymbol{h}_0[n] & 0 & \dots & 0 & 0 \\ \boldsymbol{h}_0[1] & \boldsymbol{h}_0[0] & \dots & \boldsymbol{h}_0[n-1] & \boldsymbol{h}_0[n] & \dots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & \boldsymbol{h}_0[0] & \boldsymbol{h}_0[1] \\ 0 & 0 & \dots & 0 & 0 & \dots & \boldsymbol{h}_0[1] & \boldsymbol{h}_0[0] \end{pmatrix}$$
$$+ \begin{pmatrix} \boldsymbol{h}_0[1] & \dots & \boldsymbol{h}_0[n-1] & 0 & \dots & 0 & 0 \\ \boldsymbol{h}_0[2] & \dots & 0 & 0 & \dots & 0 & 0 \\ \vdots & & \vdots & \vdots & & \vdots & \vdots \\ 0 & \dots & 0 & 0 & \dots & \boldsymbol{h}_0[3] & \boldsymbol{h}_0[2] \\ 0 & \dots & 0 & 0 & \dots & \boldsymbol{h}_0[2] & \boldsymbol{h}_0[1] \end{pmatrix}.$$

We can still prove that for many filters $\boldsymbol{h}_0$, e.g. refinement masks of splines, the corresponding matrix $H_0^* H_0$ is nonsingular with H_0 being the matrix generated from $\boldsymbol{h}_0$. Hence the algorithms converge without the acceleration factor β and the limits to the iterations have the same minimization properties. If the nonsingularity of the matrix $H_0^* H_0$ does not hold, we then return to the method of embedding the acceleration factor into the iteration to ensure the convergence. We need to choose β close to 1 in order for better data recovering as discussed in §3.2. We omit the detailed discussion here.

4.3 Stability Analysis

In this section, we discuss the stability of the algorithms given in §4.1. An algorithm of solving $H_0 \boldsymbol{v} = \boldsymbol{b} + \boldsymbol{\epsilon} = \boldsymbol{c}$ is stable if the result of the algorithm approaches to the exact solution of the equation $H_0 \boldsymbol{v} = \boldsymbol{b}$, as $\|\boldsymbol{\epsilon}\|_2 = \varepsilon \to 0$. We give the stability analysis of Algorithm 4.1, and the analysis of Algorithms 4.2 is similar.

Let the threshold $\lambda_j = C_j \varepsilon$ for some constant C_j, $J \le j < 0$. Let $C = \max_{J \le j < 0} C_j$ and without loss of generality, we take $C = 1$ below. For a given pair $(\boldsymbol{v}, \tilde{\boldsymbol{v}})$ with $\boldsymbol{v} = A_J^* \tilde{\boldsymbol{v}}$, we denote

$$|\tilde{\boldsymbol{v}}|_p := \sum_{\ell=1}^{r}\sum_{j=J}^{-1}\sum_{k=0}^{N_1-1} C_j|\tilde{v}_{\ell,j,k}|^p.$$

Let the pair $(\boldsymbol{s}^\varepsilon, \tilde{\boldsymbol{s}}^\varepsilon)$ be the limit of iteration (4.4) associated with the error bound ε and let $\boldsymbol{\nu}$ be the exact solution to linear system $H_0\boldsymbol{v} = \boldsymbol{b}$ which satisfies $\|H_0\boldsymbol{\nu} - \boldsymbol{c}\|_2 = \|\boldsymbol{\epsilon}\|_2 = \varepsilon$. Here, we note that the existence of $\boldsymbol{\nu}$ follows from the nonsingularity of H_0.

Proposition 4.2. *Let $\boldsymbol{s}^\varepsilon$ be the limit of iteration* (4.4) *associated with error bound $\|\boldsymbol{\epsilon}\|_2 = \varepsilon$ and $\boldsymbol{\nu}$ be the exact solution to $H_0\boldsymbol{v} = \boldsymbol{b}$. Then we have*

$$\lim_{\varepsilon\to 0}\|\boldsymbol{s}^\varepsilon - \boldsymbol{\nu}\|_2 = 0.$$

Proof. We only need to consider the case when $\varepsilon \le 1$. Based on the proof of Theorem 4.1 (see (4.13)), $\boldsymbol{s}^\varepsilon$ is bounded by a constant independent of ε once the initial seed $\boldsymbol{v}_0$ is fixed for all $\varepsilon \le 1$. Since $\tilde{\boldsymbol{s}}^\varepsilon = \mathcal{T}^p A_J(H_0^*\boldsymbol{c} + \sum_{\ell=1}^r H_\ell H_\ell^*\boldsymbol{s}^\varepsilon)$, we have $\|\tilde{\boldsymbol{s}}^\varepsilon\|_{\ell_2} \le \|\boldsymbol{c}\|_2 + \|\boldsymbol{s}^\varepsilon\|_2$, i.e. its ℓ_2 norm is bounded independent of ε by Proposition 3.2. This leads to $|\tilde{\boldsymbol{s}}^\varepsilon|_p \le B$ with B independent of ε and of p, $1 \le p \le 2$, because $\tilde{\boldsymbol{s}}^\varepsilon$ is a finite dimensional vector. From the boundedness of $\boldsymbol{s}^\varepsilon$ and $\tilde{\boldsymbol{s}}^\varepsilon$ and vector $\boldsymbol{\nu}$, for any ε there is a pair $(\boldsymbol{\eta}^\varepsilon, \tilde{\boldsymbol{\eta}}^\varepsilon)$ with $\tilde{\boldsymbol{\eta}}^\varepsilon = A_J\boldsymbol{\eta}^\varepsilon$ such that $\|\boldsymbol{s}^\varepsilon + \boldsymbol{\eta}^\varepsilon - \boldsymbol{\nu}\|_2 < \varepsilon$ and $|\tilde{\boldsymbol{s}}^\varepsilon + \tilde{\boldsymbol{\eta}}^\varepsilon|_p \le B'$. The pair $(\boldsymbol{\eta}^\varepsilon, \tilde{\boldsymbol{\eta}}^\varepsilon)$ depends on ε; however, the constant B' can be chosen to be independent of ε. By minimization property of $\boldsymbol{s}^\varepsilon$ given in Theorem 4.2, we have

$$\begin{aligned}
\|H_0\boldsymbol{s}^\varepsilon - \boldsymbol{c}\|_2^2 &\le \|H_0\boldsymbol{s}^\varepsilon - \boldsymbol{c}\|_2^2 + \varepsilon|\tilde{\boldsymbol{s}}^\varepsilon|_p \\
&\le \|H_0(\boldsymbol{s}^\varepsilon + \boldsymbol{\eta}^\varepsilon) - \boldsymbol{c}\|_2^2 + \varepsilon|\tilde{\boldsymbol{s}}^\varepsilon + \tilde{\boldsymbol{\eta}}^\varepsilon|_p \\
&\le 2\|H_0(\boldsymbol{s}^\varepsilon + \boldsymbol{\eta}^\varepsilon) - H_0\boldsymbol{\nu}\|_2^2 + 2\|H_0\boldsymbol{\nu} - \boldsymbol{c}\|_2^2 + \varepsilon|\tilde{\boldsymbol{s}}^\varepsilon + \tilde{\boldsymbol{\eta}}^\varepsilon|_p \\
&< 4\varepsilon^2 + \varepsilon B'.
\end{aligned}$$

Thus we derive that

$$\|H_0\boldsymbol{s}^\varepsilon - \boldsymbol{c}\|_2 < \sqrt{4\varepsilon^2 + \varepsilon B'}.$$

Since matrix H_0 is nonsingular, we can show our statement by proving $\|H_0(\boldsymbol{s}^\varepsilon - \boldsymbol{\nu})\|_2 \to 0$ as $\varepsilon \to 0$. For arbitrary ε, we have

$$\|H_0(\boldsymbol{s}^\varepsilon - \boldsymbol{\nu})\|_2 \le \|H_0\boldsymbol{s}^\varepsilon - \boldsymbol{c}\|_2 + \|H_0\boldsymbol{\nu} - \boldsymbol{c}\|_2 < \sqrt{4\varepsilon^2 + \varepsilon B'} + \varepsilon.$$

Then the stability holds by letting $\varepsilon \to 0$. □

4.4 Comparison of Algorithm 4.1, 4.2 and 5.1

We implement the iterative deconvolution algorithms developed in in §4.1. We give here the simple illustration of the performance of these algorithms applied to 1D signals. The method is evaluated by the relative error (RE), the peak signal-to-noise ratio (PSNR) and signal-to-noise ratio (SNR). Relative error is defined by

$$\mathrm{RE} = \frac{\|\boldsymbol{v}_n - \boldsymbol{v}\|_2}{\|\boldsymbol{v}\|_2},$$

PSNR is defined by

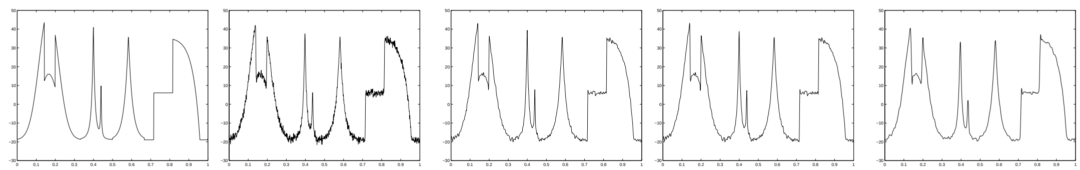
$$\mathrm{PSNR} = 10 \log_{10} \frac{N_0 \max_{k\in\mathbb{Z}}(\boldsymbol{v}_n[k])^2}{\|\boldsymbol{v}_n - \boldsymbol{v}\|_2^2},$$

where $\boldsymbol{v}_n[k]$ is the kth entry in data $\boldsymbol{v}_n$, and SNR is defined by

$$\mathrm{SNR} = 20 \log_{10} \frac{\|\boldsymbol{v}_n\|_2}{\|\boldsymbol{v}_n - \boldsymbol{v}\|_2},$$

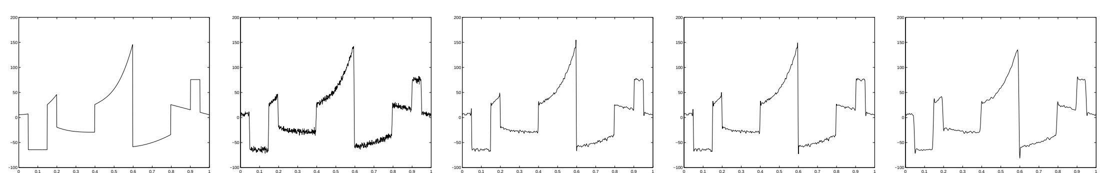
where $\boldsymbol{v}_n$ is the iterative solution, $\boldsymbol{v}$ is the original data and N_0 stands for the length of signal. We test the algorithms using the periodic boundary condition. For the simulation of 2D data set, the similar method can be carried out using the tensor product technique mentioned in §4.1 and we refer the readers to the numerical results available in [8, 10, 11] and the discussion in §5.1.

We take the signals which have sparse representation under the tight wavelet frame in the *WaveLab Toolbox* developed by Donoho's research group. Two of these signals are shown in Figure 4.1 (a) and Figure 4.2 (a). The signals are then blurred using the cubic spline and contaminated with white noise at SNR = 25, see Figure 4.1(b) and Figure 4.2(b). We process the contaminated signals by Algorithm 4.1, 4.2 and Algorithm 5.1, which will be given in next chapter, using periodic boundary condition with proper pre-extension of the data to ensure the convergence of the iterations (4.4) and (4.5). The results are given in Figure 4.1 and Figure 4.2. Other numerical results of different signals are given in Table 4.1.



(a) (b) (c) (d) (e)

Figure 4.1 Numerical results for periodic boundary condition after 12 iterations. (a) Original signal; (b) Signal blurred by filter in Example 1.2 and contaminated by white noise at SNR=25; (c) Reconstructed signal by Algorithm 4.1 (RE=0.048856, PSNR=34.113632dB, SNR=26.221678dB); (d) Reconstructed signal by Algorithm 4.2 (RE=0.049799, PSNR=33.855047dB, SNR=26.055584dB); (e) Reconstructed signal by Algorithm 5.1 (RE=0.221711, PSNR=20.171143dB, SNR=13.084247dB).



(a) (b) (c) (d) (d)

Figure 4.2 Numerical results for periodic boundary condition after 12 iterations. (a) Original signal; (b) Signal blurred by filter in Example 1.3 and contaminated by white noise at SNR=25; (c) Reconstructed signal by Algorithm 4.1 (RE=0.067718, PSNR=32.882431dB, SNR=23.385962dB); (d) Reconstructed signal by Algorithm 4.2 (RE=0.070475, PSNR=32.443317dB, SNR=23.039271dB); (e) Reconstructed signal by Algorithm 5.1 (RE=0.270130, PSNR=20.086519dB, SNR=11.368543dB).

Type of Signal	Algorithm 4.1			Algorithm 4.2			Algorithm 4.3		
	Rel. Err.	PSNR	SNR	Rel. Err.	PSNR	SNR	Rel. Err.	PSNR	SNR
HeaviSine	0.028460	33.217871	30.915279	0.028670	33.156742	30.851414	0.069625	25.455757	23.144756
Bumps	0.069254	39.577640	23.191044	0.075530	38.491668	22.437622	0.618064	17.970758	4.179333
Blocks	0.062193	30.895391	24.125186	0.067464	30.187947	23.418590	0.258222	19.087661	11.760150
Doppler	0.049995	30.580579	26.021500	0.048172	30.905828	26.344157	0.277670	15.688367	11.129433
Ramp	0.031575	32.332058	30.013089	0.039259	30.610982	28.121127	0.157946	17.146609	16.029819
Cusp	0.024914	36.003742	32.070983	0.024833	36.054969	32.099584	0.031825	33.847170	29.944680
Sing	0.089109	46.888308	21.001552	0.089637	46.860294	20.950295	0.805165	22.317274	1.882298
Piece-Polynomial	0.067718	32.882431	23.385962	0.070475	32.443317	23.039271	0.270130	20.086519	11.368543
Piece-Regular	0.048856	34.113632	26.221678	0.049799	33.855047	26.055584	0.221711	20.171143	13.084247

Table 4.1 Numerical Results of Three Algorithms.

From Figures 4.1 and 4.2 as well as Table 4.1, we can see that Algorithm 4.1 and Algorithm 4.2 give almost the same results on the given data sets with noise. Algorithm 5.1 can also recover the original data from the noised ones; however, the denoising effect is not compatible with Algorithm 4.1 and Algorithm 4.2. This may be due to the less of multiscale property in the algorithm. See Chapter 5 for detail.

Chapter 5

High Resolution Image Reconstruction via Deconvolution

In this chapter, we focus on the generalization of the algorithms given in §2.2 to the 2D case and the use of these deconvolution algorithms in high resolution image reconstructions.

5.1 Deconvolution in High Dimensional Space

In this section, the algorithms are extended to deal with the data set in 2D spaces. The higher dimensional cases other than 2D can be achieved by the similar approach inductively.

Here we assume the 2D low pass filter can be generated by two 1D low pass filters via tensor product. Suppose $\boldsymbol{h}_0^x$ and $\boldsymbol{h}_0^y$ are refinement masks (low pass filters) along x and y directions of the refinable functions ϕ^x and ϕ^y. Both of these two filters satisfy the assumption stated in (2.2) such that two separate MRA based tight frames can be constructed via the UEP. Let the nested MRA space sequences be $\{V_j^x\}$ and $\{V_j^y\}$ and $\boldsymbol{h}_\ell^x$, $\ell = 1, \ldots, r$ and $\boldsymbol{h}_{\ell'}^y$, $\ell' = 1, \ldots, r'$, be high pass filters obtained via the UEP from $\boldsymbol{h}_0^x$ and $\boldsymbol{h}_0^y$ respectively. For convenience we assume $r = r'$. The corresponding tight framelets are given by

$$\psi_\ell^x = \boldsymbol{h}_\ell^x *' \phi, \quad \text{and} \quad \psi_\ell^y = \boldsymbol{h}_\ell^y *' \phi, \quad \ell = 1, \ldots, r.$$

Here we introduce the concept of semi-convolution which is defined for a sequence $\boldsymbol{h}$ and a function f by

$$\boldsymbol{h} *' f = \sum_{k \in \mathbb{Z}} \boldsymbol{h}[k] f(\cdot - k).$$

The multiresolution analysis in $L_2(\mathbb{R}^2)$ is obtained by using the tensor product technique. Define $\mathbf{V}_j$ by

$$V_j^x \otimes V_j^y := \overline{\text{span}\{F(x,y) = f_1(x) f_2(y) : f_1 \in V_j^x, f_2 \in V_j^y\}},$$

and $\mathbf{W}_{\ell,j}$, $\ell = 1, \ldots, (r+1)^2 - 1$, by

$$V_j \otimes W_{\ell_2,j}^y,\ W_{\ell_1,j}^x \otimes V_j,\ W_{\ell_1,j}^x \otimes W_{\ell_2,j}^y, \quad \ell_1, \ell_2 = 0, \ldots, r,\ \ell_1 + \ell_2 \neq 0.$$

Then the space sequence $\{\mathbf{V}_j\}$ forms an MRA in $L_2(\mathbb{R}^2)$ and

$$\mathbf{V}_{j+1} \ominus \mathbf{V}_j = \mathbf{W}_j := \oplus_{\ell=1}^{(r+1)^2-1} \mathbf{W}_{\ell,j}.$$

The corresponding refinable functions and framelets are given by

$$\phi(x,y) := \phi^x(x)\phi^y(y),\ \psi_\ell(x,y) := \psi_{\ell_1}^x(x)\psi_{\ell_2}^y(y),\ \ell_1, \ell_2 = 0, \ldots, r,\ \ell = 1, \ldots, (r+1)^2 - 1.$$

The formulation of the 2D deconvolution problem can be carried out using the MRA in $L_2(\mathbb{R}^2)$. A similar result can be obtained. We omit the details here since we concern more on the discrete form of the algorithms.

The convolution equation we are to solve is $\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{c}$ with $\boldsymbol{h}_0 = \boldsymbol{h}_0^x \otimes \boldsymbol{h}_0^y$ where $\boldsymbol{v}$, $\boldsymbol{c} \in \mathbb{R}^2$. This is the convolution with indices in $\mathbb{Z}^2$ which is defined by

$$\boldsymbol{h}_0 * \boldsymbol{v}[i_1, i_2] = \sum_{p_1, p_2} \boldsymbol{h}_0[i_1 - p_1, i_2 - p_2] \boldsymbol{v}[p_1, p_2].$$

The corresponding high pass filters obtained via UEP and tensor product are

$$\boldsymbol{h}_\ell = \boldsymbol{h}_{\ell_1}^x \otimes \boldsymbol{h}_{\ell_2}^y, \quad \ell_1, \ell_2 = 0, \ldots, r,\ \ell = 1, \ldots, (r+1)^2 - 1.$$

Suppose the matrices generated from $\boldsymbol{h}_\ell^x$ and $\boldsymbol{h}_\ell^y$ be H_ℓ^x and H_ℓ^y, $\ell = 0, \dots, r$. Then the 2D convolution can be written in a matrix-vector multiplication form using the tensor product of these matrices. First the convolution equation is rewritten in the following matrix equation as

$$H_0^y \boldsymbol{v} (H_0^x)^* = \boldsymbol{c}.$$

This equation is equivalent to the linear system $H_0 \dot{\boldsymbol{v}} = \dot{\boldsymbol{c}}$ where $H_0 = H_0^x \otimes H_0^y$, $\dot{\boldsymbol{v}}$ and $\dot{\boldsymbol{c}}$ are obtained by rearranging the corresponding matrices $\boldsymbol{v}$ and $\boldsymbol{c}$ according to column, i.e. $\dot{\boldsymbol{v}} = [\boldsymbol{v}_{(1)}^*, \dots, \boldsymbol{v}_{(N_0)}^*]^*$ with $\boldsymbol{v} = [\boldsymbol{v}_{(1)}, \dots, \boldsymbol{v}_{(N_0)}]$ and $\boldsymbol{v}_{(k)}$, $k = 1, \dots, N_0$, the column vectors of $\boldsymbol{v}$; $\dot{\boldsymbol{c}}$ is obtained in a same manner. The 2D deconvolution problem is hence reduced to solving a 1D linear equation system. The matrix form of other 2D high pass filters is given by

$$H_\ell = H_{\ell_1}^x \otimes H_{\ell_2}^y, \ \ell_1, \ell_2 = 0, \dots, r, \ \ell = 1, \dots, (r+1)^2 - 1.$$

Moreover we have the following identity on H_ℓ, $\ell = 0, \dots, (r+1)^2 - 1$:

$$H_0^* H_0 + H_1^* H_1 + \dots + H_{(r+1)^2-1}^* H_{(r+1)^2-1} = I.$$

So all the analysis used in Chapter 2 and Chapter 3 can be carried over with some efforts. If the given data set is in $\ell_2(\mathbb{Z}^2)$, we can use Algorithm 2.2 and 2.3 to solve the 2D convolution equation $\boldsymbol{h}_0 * \boldsymbol{v} = \boldsymbol{c}$ or equivalent the linear system $H_0 \dot{\boldsymbol{v}} = \dot{\boldsymbol{c}}$ derived from this convolution equation. Moreover, we always have the convergence of Algorithm 2.2 and 2.3 with acceleration factor β.

When we only consider the finite data set, we can remove the acceleration factor by properly modifying the size of the given data. For the 1D case, we know by Proposition 4.1 that when the periodic boundary condition is considered, given a low pass filter, the length of data can be properly extended such that the matrix generated from the filter is nonsingular. Thus in the following we assume H_0^x and H_0^y are nonsingular. Because

$$H_0^* H_0 = ((H_0^x)^* \otimes (H_0^y)^*)(H_0^x \otimes H_0^y) = ((H_0^x)^* H_0^x) \otimes ((H_0^y)^* H_0^y),$$

the nonsingularity of $(H_0^x)^* H_0^x$ and $(H_0^y)^* H_0^y$ guarantees that $H_0^* H_0$ is also nonsingular (more detailed discussion can be found in [27]). It implies that the analysis used in the proof of Theorem 4.1 can be applied and the algorithms converge without the acceleration factor β. After we obtain the solutions $\dot{\boldsymbol{s}}$ to these algorithms, we can reverse the arrangement process to get the two dimensional solutions form $\boldsymbol{s}$.

The method using tensor product to extend the algorithms to deal with the 2D data set can be generalized for the data set in higher dimensional space by applying the tensor product technique inductively. Furthermore, as long as the finite data set with periodic boundary condition is considered, we always have the convergence of algorithms without using the acceleration factor β.

5.2 High Resolution Image Reconstruction

In this section, we will show how to transfer a high resolution image reconstruction problem into the 2D deconvolution problem.

High resolution images are desired in many situations, but made impossible because of hardware limitations. Increasing the resolution by image processing techniques is therefore of great importance. The high resolution image reconstruction can be obtained by mapping several low resolution images onto a single high resolution image plane, then interpolating it between the nonuniformly spaced samples. It can also be put into a Bayesian framework by using a Huber Markov random field. In this thesis, we follow the approach in [4] and consider creating high resolution images of a scene from the low resolution images of the same scene, where the low resolution images are obtained from sensor arrays which are shifted from each others with subpixel displacements.

Suppose the image of a given scene is obtained by the sensors with $N_1 \times N_2$ pixels with the length and width of each pixel being T_1 and T_2. Such sensors are called the low resolution sensors and the image captured by them are the low resolution images. We are going to construct a high resolution image from an array of $K_1 \times K_2$ low resolution images captured by a sensor array. The resolution of the constructed image depends on the array of the sensors, i.e. the image constructed is with $M_1 \times M_2$ pixels, where $M_1 = K_1N_1$ and $M_2 = K_2N_2$. Since the length and width of the original scene we are interested in are fixed, or say our region of interest is stationary, the length and width of each high resolution image pixel is

$$P_1 = \frac{T_1 N_1}{K_1 N_1} = \frac{T_1}{K_1} \quad \text{and} \quad P_2 = \frac{T_2 N_2}{K_2 N_2} = \frac{T_2}{K_2}.$$

In other words, the sampling block of the scene changes from $T_1 \times T_2$ to $P_1 \times P_2$, which is smaller and hence we can hope for a high resolution result of the scene.

Let the function $G(x_1, x_2)$ be the intensity function of the underlying scene (real existed image without sampling) in the region of interest with $0 \le x_1 < T_1 N_1$ and $0 \le x_2 < T_2 N_2$. Then the $M_1 \times M_2$ high resolution image means that we need to calculate for each sampling block:

$$\frac{K_1 K_2}{T_1 T_2} \int_{iT_1/K_1}^{(i+1)T_1/K_1} \int_{jT_2/K_2}^{(j+1)T_2/K_2} G(x_1, x_2)\,\mathrm{d}x_1 \mathrm{d}x_2, \quad 0 \le i < M_1,\ 0 \le j < M_2. \tag{5.1}$$

This can be viewed as we average the intensity of the original scene in each 2D sampling interval $P_1 \times P_2$.

To have enough information to construct such high resolution one, we introduce the subpixel displacements between each two consecutive sensor arrays. For sensor (k_1, k_2), $0 \le k_1 < K_1$, $0 \le k_2 < K_2$, the horizontal and vertical displacements with respect to the $(0, 0)$th sensor are given by

$$d^x_{k_1 k_2} = \Big(k_1 + \frac{1 - K_1}{2}\Big)\frac{T_1}{K_1} \quad \text{and} \quad d^y_{k_1 k_2} = \Big(k_2 + \frac{1 - K_2}{2}\Big)\frac{T_2}{K_2}.$$

Thus the intensity at (n_1, n_2)th pixel of the low resolution image captured by this sensor is given by:

$$G_{k_1 k_2}[n_1, n_2] = \frac{1}{T_1 T_2} \int_{T_1 n_1 + d^x_{k_1 k_2}}^{T_1(n_1+1)+d^x_{k_1 k_2}} \int_{T_2 n_2 + d^y_{k_1 k_2}}^{T_2(n_2+1)+d^y_{k_1 k_2}} G(x_1, x_2)\,\mathrm{d}x_1 \mathrm{d}x_2 + \epsilon_{k_1 k_2}[n_1, n_2], \tag{5.2}$$

where $0 << n_1 < N_1$, $0 \le n_2 < N_2$ and $\epsilon_{k_1 k_2}[n_1, n_2]$ is the noise term. By applying the mid-point quadrature rule, we have

$$G_{k_1 k_2}[n_1, n_2] \approx G\Big(\frac{T_1}{K_1}(K_1 n_1 + k_1 + \frac{1}{2}), \frac{T_2}{K_2}(K_2 n_2 + k_2 + \frac{1}{2})\Big).$$

Furthermore, we rearrange the indices of all $K_1 \times K_2$ low resolution images with the approximated intensity value given above in the following way:

$$\tilde{G}[K_1 n_1 + k_1, K_2 n_2 + k_2] = G_{k_1 k_2}[n_1, n_2],$$

to form an $M_1 \times M_2$ image $\tilde{G}$

$$\tilde{G}[i,j] \approx G\Big(\frac{T_1}{K_1}(i+\frac{1}{2}), \frac{T_2}{K_2}(j+\frac{1}{2})\Big), \quad 0 \le i < M_1,\ 0 \le j < M_2,$$

which can be viewed as applying the mid-point quadrature rule to approximate the integral value in (5.1). So the function $\tilde{G}$ obtained by such approximation is an approximation to the high resolution image modelled by (5.1) and it is already better than any one of $K_1 \times K_2$ low resolution images. The high resolution image obtained in such a way is called the observed high resolution image.

To get a much better image than the observed one, we use the quadrature rule of higher order to approximate the integral (5.2) for each sensor (k_1, k_2) and then add all the approximation of the sensor array together with weight matrices for the sensors. Here we take rectangular rule. By doing this we can set up a linear system of the high resolution pixel value $G(i,j)$ to the observed low resolution pixel value $\tilde{G}(i,j)$ with (i,j) the high resolution pixel in the region of interest. Since the region of interest is a finite domain, we need the boundary conditions to extrapolate G out of the boundary. We use the periodic boundary condition here. The rectangular quadrature rule is given in [4] and we omit the derivation in this thesis. The linear system obtained from the quadrature rule is defined by $M_1 \times M_1$ matrix H_0^x and $M_2 \times M_2$ matrix H_0^y with the rows given by

$$\frac{1}{K_1}\Big[\frac{1}{2}, \underbrace{1, \ldots, 1}_{K_1-1}, \frac{1}{2}\Big], \quad \text{and} \quad \frac{1}{K_2}\Big[\frac{1}{2}, \underbrace{1, \ldots, 1}_{K_2-1}, \frac{1}{2}\Big].$$

We call these two matrices the blurring matrix. They are the combination of blurring matrices for each sensor in the array using the weight matrices. The use of periodic boundary condition implies that H_0^x and H_0^y are circulant matrices generated by the low pass filters $\boldsymbol{h}_0^x$ and $\boldsymbol{h}_0^y$ which are exact the rows of these two matrices shown above. It can be easily checked that these two filters satisfy the UEP condition (1.5) and hence two corresponding tight frame system can be constructed.

The linear system obtained in such a way is $M_1M_2 \times M_1M_2$ and the known/unknown $M_1M_2 \times 1$ vectors are derived from G and $\tilde{G}$ by the way given in last section to convert the matrices G and $\tilde{G}$ to the vectors $\boldsymbol{G}$ and $\tilde{\boldsymbol{G}}$. More precise, let $G = [G_{(1)}, \ldots, G_{(M_1)}]$ and $\tilde{G} = [\tilde{G}_{(1)}, \ldots, \tilde{G}_{(M_1)}]$, where G_i and $\tilde{G}_i$ are column vectors of matrices G and $\tilde{G}$. Then the vectors generated from these two matrices are

$$\boldsymbol{G} = [G^*_{(1)}, \ldots, G^*_{(M_1)}]^* \quad \text{and} \quad \tilde{\boldsymbol{G}} = [\tilde{G}^*_{(1)}, \ldots, \tilde{G}^*_{(M_1)}]^*.$$

The linear system is

$$\big(H_0^x \otimes H_0^y\big)\boldsymbol{G} = \tilde{\boldsymbol{G}}.$$

By the analysis in last section, such linear system is equivalent to a 2D convolution equation because we have

$$\big(H_0^x \otimes H_0^y\big)\boldsymbol{G} \Longleftrightarrow H_0^y G (H_0^x)^*,$$

which is the convolution of G by filter $\boldsymbol{h}_0 := \boldsymbol{h}_0^x \otimes \boldsymbol{h}_0^y$. So the 2D deconvolution algorithms can be applied to solve out the value of $\boldsymbol{G}$. Finally, the high resolution image is constructed by rearranging $\boldsymbol{G}$ to G. The algorithms discussed before are also called the high resolution image reconstruction algorithms in [8, 10, 11], in which quite a lot of simulation results are available.

5.3 Differences in Denoising Schemes

In the end of this chapter, we discuss the algorithm used in papers [8, 10, 11] on high resolution image reconstructions with a comparison of the denoising scheme used in our algorithms given in previous chapters. The convergence of the algorithm, the minimization of the solution and the stability property are also given. In the later discussion, without further notification, the periodic boundary condition is implemented to the finite data set.

The algorithm used in high resolution image reconstructions in [8, 10, 11] applies a different decomposition operator in denoising scheme. Let $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the high pass filters generated from the known low pass filter $\boldsymbol{h}_0$ using UEP. Let $H_0, \ldots, H_r$ be the matrices generated from these filters. The decomposition operator used is defined as:

$$B_J = [H_0^J,\ H_1 H_0^{J-1}, \ldots, H_r H_0^{J-1}, \ldots, H_1, \ldots, H_r]^t \tag{5.3}$$

and the reconstruction operator is its adjoint operator

$$B_J^* = [(H_0^J)^*,\ (H_0^{J-1})^* H_1^*, \ldots, (H_0^{J-1})^* H_r^*, \ldots, H_1^*, \ldots, H_r^*]. \tag{5.4}$$

It can be easily seen that the difference between A_J and B_J is in the blocks. In A_J, each block is of form

$$H_{\ell,j} \prod_{j'=j-1}^{-1} H_{0,j'},$$

which is a product of matrices generated from up sampled filters $\boldsymbol{h}_{\ell,j}$, $\ell = 0, \ldots, r$; while in B_J, each block is of form

$$H_\ell H_0^j,$$

which is a product of matrices generated from filters $\boldsymbol{h}_\ell$, $\ell = 0, \ldots, r$, without up sampling. This difference implies the filters in decomposition B_J are stationary without up sampling process. Nevertheless, the identity $B_J^* B_J = I$ still hold. In fact, one can prove this identity easily by modifying the proof of Proposition 1.2. The denoising scheme is formed by applying the threshold operator $\mathcal{T}^p$ to $B_J \boldsymbol{v}$, i.e.

$$[H_0^J \boldsymbol{v},\ \mathcal{D}_{\lambda_J}^p (H_1 H_0^{J-1} \boldsymbol{v}), \ldots, \mathcal{D}_{\lambda_J}^p (H_r H_0^{J-1} \boldsymbol{v}), \ldots, \mathcal{D}_{\lambda_1}^p (H_1 \boldsymbol{v}), \ldots, \mathcal{D}_{\lambda_1}^p (H_r \boldsymbol{v})]^t.$$

With these we have the following algorithm which is used in high resolution image reconstructions, see [8, 10, 11].

Algorithm 5.1.

(i) *Choose an initial approximation* $\boldsymbol{v}_0$ *(e.g.* $\boldsymbol{v}_0 = \boldsymbol{c}$*);*

(ii) *Iterate on* n *until convergence:*

$$\boldsymbol{v}_{n+1} = H_0^* \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* (B_J^* \mathcal{T}^p B_J)\,(H_\ell \boldsymbol{v}_n). \tag{5.5}$$

Since $\boldsymbol{c}$ contains noise, it was suggested by the numerical simulations in [8, 10, 11], one needs to take a additional step of denoising from the final iteration:

(iii) $\boldsymbol{v} = B_J^* \mathcal{T}^p B_J(\boldsymbol{v}_{n_0})$.

The convergence of iteration (5.5) in Algorithm 5.1 can be proved based on the nonsingularity of the circulant matrix H_0. This convergence property is stated in the following result.

Theorem 5.1. *Let $\boldsymbol{h}_0$ be the low pass filter in the convolution equation and $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_r$ be the high pass filters generated from $\boldsymbol{h}_0$ via the UEP. The corresponding circulant matrices are $H_0, \ldots, H_r$ with $H_0^* H_0$ being nonsingular. Then iteration* (5.5) *in Algorithm 5.1 converges for any initial seed $\boldsymbol{v}_0$ and the limit satisfies*

$$\boldsymbol{s} = H_0^* \boldsymbol{c} + \sum_{\ell=1}^{r} H_\ell^* B_J^* \mathcal{T}^p B_J(H_\ell \boldsymbol{s}). \tag{5.6}$$

Proof. Because H_0 is a nonsingular circulant matrix and $I = H_0^* H_0 + \sum_{\ell=1}^r H_\ell^* H_\ell$, there exists a constant $\mu < 1$ such that $\|\sum_{\ell=1}^r H_\ell^* H_\ell\|_2 = \|I - H_0^* H_0\|_2 \leq \mu$. Let $H = [H_1, \ldots, H_r]^t$, then we have

$$\|H\|_2^2 = \max_{\|\boldsymbol{u}\|_2=1} \|H\boldsymbol{u}\|_2^2 = \max_{\|\boldsymbol{u}\|_2=1} \boldsymbol{u}^* \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{u} = \|\sum_{\ell=1}^{r} H_\ell^* H_\ell\|_2 \leq \mu.$$

Denote $\boldsymbol{g}_{(v,v')} = [(\mathcal{T}^p B_J H_1 \boldsymbol{v} - \mathcal{T}^p B_J H_1 \boldsymbol{v}'), \ldots, (\mathcal{T}^p B_J H_r \boldsymbol{v} - \mathcal{T}^p B_J H_r \boldsymbol{v}')]^t$ for any two vectors $\boldsymbol{v}$ and $\boldsymbol{v}'$. Following the proof of Theorem 3.1, given any $\boldsymbol{v}_0$, for any positive integers m and n,

$$\begin{aligned}
\|\boldsymbol{v}_{n+m} - \boldsymbol{v}_n\|_2 &= \|\sum_{\ell=1}^{r} H_\ell^* B_J^* (\mathcal{T}^p B_J H_\ell \boldsymbol{v}_{n+m-1} - \mathcal{T}^p B_J H_\ell \boldsymbol{v}_{n-1})\|_2 \\
&= \|H^* B_J^* \boldsymbol{g}_{(\boldsymbol{v}_{n+m-1}, \boldsymbol{v}_{n-1})}\|_2 \\
&\leq \|H^*\|_2 \|\boldsymbol{g}_{(\boldsymbol{v}_{n+m-1}, \boldsymbol{v}_{n-1})}\|_2 \\
&\leq \|H^*\|_2 \|H\|_2 \|\boldsymbol{v}_{n+m-1} - \boldsymbol{v}_{n-1}\|_2 \\
&\leq \mu \|\boldsymbol{v}_{n+m-1} - \boldsymbol{v}_{n-1}\|_2.
\end{aligned}$$

Similarly, one can prove by using (5.5)

$$\|\boldsymbol{v}_n\|_2 \le \|\boldsymbol{c}\|_2 + \mu\|\boldsymbol{v}_{n-1}\|_2 \le \frac{1}{1-\mu}\|\boldsymbol{c}\|_2 + \|\boldsymbol{v}_0\|_2.$$

Thus the iteration sequence $\{\boldsymbol{v}_n\}$ is a Cauchy sequence and the limit exists and satisfies (4.12). □

Let

$$\tilde{\boldsymbol{s}}_\ell = \mathcal{T}^p B_J(H_\ell \boldsymbol{s}),\ \ell = 1, \dots, r.$$

Since solution $\boldsymbol{s}$ of iteration (5.5) satisfies (5.6), we have the $(r+1)$-tuple $(\boldsymbol{s}, \tilde{\boldsymbol{s}}_1, \dots, \tilde{\boldsymbol{s}}_r)$ satisfies $\boldsymbol{s} = H_0^*\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* B_J^* \tilde{\boldsymbol{s}}_\ell$. We will prove this tuple of finite data sets satisfies a similar property to its infinite counterpart and hence $\boldsymbol{s}$ is a solution to (2.6).

Theorem 5.2. *Let $\boldsymbol{h}_1, \dots, \boldsymbol{h}_r$ be the high pass filters obtained from $\boldsymbol{h}_0$ by the UEP and $H_0, H_1, \dots, H_r$ be the corresponding circulant matrices of these filters. Then given fixed $1 \le p \le 2$, the $(r+1)$-tuple $(\boldsymbol{s}, \tilde{\boldsymbol{s}}_1, \dots, \tilde{\boldsymbol{s}}_r)$ satisfies the following inequality*

$$\begin{aligned} &\|H_0(\boldsymbol{s}+\boldsymbol{\eta}) - \boldsymbol{c}\|_2^2 + \sum_{\ell=1}^r \sum_{j=1}^J \sum_{k=0}^{N_0-1} \lambda_j |\tilde{s}_{\ell,j,k} + \tilde{\eta}_{\ell,j,k}|^p \\ &\qquad \ge \|H_0\boldsymbol{s} - \boldsymbol{c}\|_2^2 + \sum_{\ell=1}^r \sum_{j=1}^J \sum_{k=0}^{N_0-1} \lambda_j |\tilde{s}_{\ell,j,k}|^p, \end{aligned} \tag{5.7}$$

for any $(r+1)$-tuple $(\boldsymbol{\eta}, \tilde{\boldsymbol{\eta}}_1, \dots, \tilde{\boldsymbol{\eta}}_r)$ satisfying $\tilde{\boldsymbol{\eta}}_\ell = B_J(H_\ell \boldsymbol{\eta})$ for $\ell = 1, \dots, r$, where B_J is given in (5.3).

Proof. We prove this theorem by proving a more general inequality. For a given sequence $\boldsymbol{a}$, we define $\tilde{\boldsymbol{v}}_\ell^\star = \mathcal{T}^p B_J(H_\ell \boldsymbol{a})$ and $\boldsymbol{v}_\ell^\star = B_J^*(\tilde{\boldsymbol{v}}_\ell^\star)$ for $\ell = 1, \dots, r$. Next we denote $\boldsymbol{c}$ by $\boldsymbol{v}_0^\star$ and define $\boldsymbol{v}^\star$ by

$$\boldsymbol{v}^\star = B_1^*\{\boldsymbol{v}_\ell^\star\}_{\ell=0}^r = H_0^*\boldsymbol{c} + \sum_{\ell=1}^r H_\ell^* B_J^* \mathcal{T}^p B_J(H_\ell \boldsymbol{a}).$$

With this set up, we show that the inequality

$$\begin{aligned}
&\|H_0(\boldsymbol{v}^\star+\boldsymbol{\eta})-\boldsymbol{c}\|_2^2+\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1}\lambda_j|(\tilde{v}_\ell^\star)_{\ell',j,k}+(\tilde{\eta}_\ell)_{\ell',j,k}|^p+\sum_{\ell=1}^{r}\|H_\ell(\boldsymbol{v}^\star+\boldsymbol{\eta})-H_\ell\boldsymbol{a}\|_2^2\\
&\quad\geq\|H_0\boldsymbol{v}^\star-\boldsymbol{c}\|_2^2+\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1}\lambda_j|(\tilde{v}_\ell^\star)_{\ell',j,k}|^p+\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{v}^\star-H_\ell\boldsymbol{a}\|_2^2+\|\boldsymbol{\eta}\|_2^2,
\end{aligned} \tag{5.8}$$

holds for any $(r+1)$-tuple $(\boldsymbol{\eta},\tilde{\boldsymbol{\eta}}_1,\ldots,\tilde{\boldsymbol{\eta}}_r)$ with $\tilde{\boldsymbol{\eta}}_\ell=B_J(H_\ell\boldsymbol{\eta})$ for $\ell=1,\ldots,r$. Note that if we take $\boldsymbol{a}=\boldsymbol{s}$, where $\boldsymbol{s}$ is the limit to iteration (4.5) satisfying (4.12), then $\boldsymbol{v}^\star=H_0^*\boldsymbol{c}+\sum_{\ell=1}^{r}H_\ell^*B_J^*\mathcal{T}^pB_J(H_\ell\boldsymbol{s})=\boldsymbol{s}$, and inequality (5.7) can be easily deduced from (5.8). In the following we give the proof of (5.8), which is similar to that of Proposition 3.3.

Given the $(r+1)$-tuple $(\boldsymbol{\eta},\tilde{\boldsymbol{\eta}}_1,\ldots,\tilde{\boldsymbol{\eta}}_r)$, we expand the left hand side of (5.8) as follows:

$$\begin{aligned}
&\|H_0(\boldsymbol{v}^\star+\boldsymbol{\eta})-\boldsymbol{c}\|_2^2+\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1}\lambda_j|(\tilde{v}_\ell^\star)_{\ell',j,k}+(\tilde{\eta}_\ell)_{\ell',j,k}|^p+\sum_{\ell=1}^{r}\|H_\ell(\boldsymbol{v}^\star+\boldsymbol{\eta})-H_\ell\boldsymbol{a}\|_2^2\\
&\quad=\|H_0\boldsymbol{v}^\star-\boldsymbol{c}\|_2^2+\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1}\lambda_j|(\tilde{v}_\ell^\star)_{\ell',j,k}|^p+\sum_{\ell=1}^{r}\|H_\ell\boldsymbol{v}^\star-H_\ell\boldsymbol{a}\|_2^2+\|\boldsymbol{\eta}\|_2^2\\
&\qquad+\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1}\lambda_j\left(|(\tilde{v}_\ell^\star)_{\ell',j,k}+(\tilde{\eta}_\ell)_{\ell',j,k}|^p-|(\tilde{v}_\ell^\star)_{\ell',j,k}|^p\right)\\
&\qquad+2\langle\boldsymbol{\eta},\boldsymbol{v}^\star-H_0^*\boldsymbol{c}-\sum_{\ell=1}^{r}H_\ell^*H_\ell\boldsymbol{a}\rangle.
\end{aligned} \tag{5.9}$$

Compare with (5.9), we only need to show

$$\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1}\lambda_j\left(|(\tilde{v}_\ell^\star)_{\ell',j,k}+(\tilde{\eta}_\ell)_{\ell',j,k}|^p-|(\tilde{v}_\ell^\star)_{\ell',j,k}|^p\right)+2\langle\boldsymbol{\eta},\boldsymbol{v}^\star-H_0^*\boldsymbol{c}-\sum_{\ell=1}^{r}H_\ell^*H_\ell\boldsymbol{a}\rangle\geq 0. \tag{5.10}$$

Using the definition of $\boldsymbol{v}^\star$ and $B_J^* B_J = I$, we can simplify the inner product in (5.9) as

$$\begin{aligned}\langle \boldsymbol{\eta}, \boldsymbol{v}^\star - H_0^*\boldsymbol{c} - \sum_{\ell=1}^{r} H_\ell^* H_\ell \boldsymbol{a}\rangle &= \langle \boldsymbol{\eta}, \sum_{\ell=1}^{r} H_\ell^* (B_J^* \mathcal{T}^p B_J(H_\ell \boldsymbol{a}) - H_\ell \boldsymbol{a})\rangle \\ &= \sum_{\ell=1}^{r} \langle B_J(H_\ell \boldsymbol{\eta}), \mathcal{T}^p B_J(H_\ell \boldsymbol{a}) - B_J(H_\ell \boldsymbol{a})\rangle \\ &= \sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1} (\tilde{\eta}_\ell)_{\ell',j,k}((\tilde{v}_\ell^\star)_{\ell',j,k} - (a_\ell)_{\ell',j,k}).\end{aligned} \tag{5.11}$$

With this, (5.10) becomes

$$\begin{aligned}\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1} \lambda_j \left(|(\tilde{v}_\ell^\star)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p - |(\tilde{\boldsymbol{v}}_\ell^\star)_{\ell',j,k}|^p\right) \\ + 2\sum_{\ell=1}^{r}\sum_{\ell'=1}^{r}\sum_{j=1}^{J}\sum_{k=0}^{N_1-1} (\tilde{\eta}_\ell)_{\ell',j,k}((\tilde{v}_\ell^\star)_{\ell',j,k} - (a_\ell)_{\ell',j,k}) \geq 0,\end{aligned}$$

which we will be proven by showing each summand is nonnegative, i.e.

$$\lambda_j \left(|(\tilde{v}_\ell^\star)_{\ell',j,k} + (\tilde{\eta}_\ell)_{\ell',j,k}|^p - |(\tilde{v}_\ell^\star)_{\ell',j,k}|^p\right) + 2(\tilde{\eta}_\ell)_{\ell',j,k}((\tilde{v}_\ell^\star)_{\ell',j,k} - (a_\ell)_{\ell',j,k}) \geq 0.$$

The rest of the proof follows the same discussion in the proof of Proposition 3.3. □

Based on the minimization property of the solution, the stability of Algorithm 5.1 can be established using the similar technique in Proposition 4.2. We only list the result below.

Proposition 5.1. *Let $\boldsymbol{s}^\varepsilon$ be the limit of iteration* (5.5) *associated with error bound $\|\boldsymbol{\epsilon}\|_2 = \varepsilon$ and $\boldsymbol{\nu}$ be the exact solution to $H_0\boldsymbol{v} = \boldsymbol{b}$. Then we have*

$$\lim_{\varepsilon\to 0} \|\boldsymbol{s}^\varepsilon - \boldsymbol{\nu}\|_2 = 0.$$

Bibliography

[1] G. Beylkin, R. Coifman and V. Rokhlin, Fast Wavelet Transforms and Numerical Algorithms I, *Comm. Pure Appl. Math.* 44: 141-183, 1991.

[2] L. Borup, R. Grivonbal and M. Nielsen, Tight Wavelet Frames in Lebesgue and Sobolev Spaces, *J. Funct. Spaces Appl.* 2(3), 2004.

[3] L. Borup, R. Grivonbal and M. Nielsen, Bi-framelet Systems with Few Vanishing Moments Characterize Besov Spaces, *Appl. Comput. Harmon. Anal.* 17: 3-28, 2004.

[4] N. Bose and K. Boo, High Resolution Image Reconstruction with Multisensors, *Int. J. Imaging Syst. Technol.*, 9: 294-304, 1998.

[5] D. Chen, On the Splitting Trick and Wavelet Frame Packets, *SIAM J. Math. Anal.* 31(4): 726-739, 2000.

[6] C. de Boor, R. DeVore and A. Ron, On the Construction of Multivariate (Pre)wavelet, *Constr. Approx.* 9: 123-166, 1993.

[7] R. Chan, T. Chan, L. Shen and Z. Shen, A Wavelet Method for High-resolution Image Reconstruction with Displacement Errors, *Proceedings of the 2001 International Symposium of Intelligent Multimedia, Video and Speech Processing,* 24-27, 2001.

[8] R. Chan, T. Chan, L. Shen and Z. Shen, Wavelet Algorithms for High-resolution Image Reconstruction, *SIAM J. Sci. Comput.* 24(4): 1408-1432, 2003.

[9] R. Chan, T. Chan, L. Shen and Z. Shen, Wavelet Deblurring Algorithms for Sparially Varying Blur from High-resolution Image Reconstruction, *Linear Algebra Appl.* 366: 139-155, 2003.

[10] R. Chan, S. D. Riemenschneider, L. Shen and Z. Shen, Tight Frame: The Efficient Way for High-resolution Image Reconstruction, *Appl. Comput. Harmon. Anal.* 17: 91-115, 2004.

[11] R. Chan, S. D. Riemenschneider, L. Shen and Z. Shen, High-resolution Image Reconstruction with Displacement Errors: A Frame Approach, *Int. J. Imaging Syst. Technol.* 14: 91-104, 2004.

[12] R. Chan, Z. Shen and T. Xia, Resolution Enhancement for Video Clips: Tight Frame Approach, *Proceedings of IEEE International Conference on Advanced Video and Signal-Based Surveillance,* 406-410, Como, Italy, 2005.

[13] R. Chan, Z. Shen and T. Xia, Resolution Enhancement for Video Clips Using Tight Frame Systems, Research Report, the Chinese University of Hong Kong.

[14] C. K. Chui and W. He, Compactly Supported Tight Frames Associated With Refinable Functions, *Appl. Comput. Harmon. Anal.* 8(3): 293-319, 2000.

[15] C. K. Chui, W. He and J. Stöckler, Compact Supported Tight and Sibling Frames with Maximum Vanishing Moments, *Appl. Comput. Harmon. Anal.*, 13: 224-262, 2002.

[16] C. K. Chui, X. Shi and J. Stöckler, Affine Frames, Quasi-affine Frames and their Duals., *Adv. Comput. Math.* 8(1-2): 1–17, 1998.

[17] A. Cohen, M. Hoffmann and M. Reiss, Adaptive Wavelet Galerkin Methods for Linear Inverse Probelms, *SIAM J. Numer. Anal.* 42(4): 1479–1501, 2004.

[18] I. Daubechies, Ten Lectures on Wavelets, CBMS Conference Series in Applied Mathematics, 61, *SIAM, Philadelphia, PA,* 1992.

[19] I. Daubechies, M. Defrise and C. De Mol, An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constraint, *Comm. Pure Appl. Math.* 57: 1413-1541, 2004.

[20] I. Daubechies, B. Han, A. Ron and Z. Shen, Framelets: MRA-based Constructions of Wavelet Frames, *Appl. Comput. Harmon. Anal.* 14: 1-46, 2003.

[21] C. De Mol and M. Defrise, A Note on Wavelet-based Inversion Algorithms, *Inverse problems, image analysis, and medical imaging (New Orleans, LA, 2001),* M. Z. Nashed and O. Scherzer eds, 85–96, Contemp. Math. 313, *Amer. Math. Soc., Providence, RI,* 2002.

[22] B. Dong and Z. Shen, Pseudo-spline Wavelets and Framelets, *preprint*.

[23] D. Donoho, Nonlinear Solution of Linear Inverse Problems by Wavelet-Vaguelette Decomposition, *Appl. Comput. Harmon. Anal.* 2: 101-126, 1995.

[24] D. Donoho and M. Raimondo, Translation Invariant Deconvolution in a Periodic Setting, *Int. J. Wavelets Multiresolut. Inf. Process.* 2(4): 415-431, 2004.

[25] H. Engl, M. Hanke and A. Neubauer, Regularization of Inverse Problems, *Kluwer Academic Publishers, Dordrecht, Boston,* 1996.

[26] M. Foster, An Application of the Wiener-Kolmogorov Smoothing Theory to Matrix Inversion, *J. SIAM* 9(3): 387-392, 1961.

[27] R. Horn and C. Johnson, Topics in matrix analysis, *Cambridge; New York: Cambridge University Press,* 1991.

[28] R. Jia and C. Micchelli, Using the refinement equations for the construction of pre-wavelets. II. Powers of two. *Curves and surfaces (Chamonix-Mont-Blanc, 1990),* Academic Press, Boston, MA, 209–246, 1991.

[29] R. Jia and Z. Shen, Multiresolution and Wavelets, *Proc. Edinburgh Math. Soc.* 37: 271-300, 1994.

[30] J. Kalifa, S. Mallat and B. Rougé, Deconvolution by Thresholding in Mirror Wavelet Bases, *IEEE Trans. Image Process.* 12(4): 446–457, 2003.

[31] J. Kalifa and S. Mallat, Thresholding Estimators for Linear Inverse Problems and Deconvolutions, *Ann. Statist.* 31(1): 58-109, 2003.

[32] R. Long and W. Chen, Wavelet Basis Packets and Wavelet Frame Packets, *J. Fourier Anal. Appl.* 3(3): 239-256, 1997.

[33] S. Mallat, A Wavelet Tour of Signal Processing, second edition, *Academic Press, Inc., San Diego, CA,* 1998.

[34] A. Ron and Z. Shen, Affine Systems in $L_2(\mathbb{R}^d)$: the Analysis of the Analysis Operator, *J. Funct. Anal.* 148: 408-447, 1997.

[35] A. Ron and Z. Shen, Affine Systems in $L_2(\mathbb{R}^d)$ II: Dual Systems, *J. Fourier Anal. Appl.* 3: 617-637, 1997.

[36] A. N. Tikhonov, On the Solution of Incorrectly put Problems and the Regularization Method, *Soviet Math. Doklady* 4: 1035-1038, 1963.

[37] N. Wiener, Extrapolation, Interpolation and Smoothing of Stationary Time Series, *Wiley, New York,* 1949.