

Multilevel Methods for Discrete Ill-Posed Problems: Application to
Deblurring

A dissertation

submitted by

Malena Inés Español

In partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in

Mathematics

TUFTS UNIVERSITY

May 2009

ADVISOR: Misha E. Kilmer

Abstract

Multilevel Methods for Discrete Ill-Posed Problems: Application to Deblurring

by

Malena Inés Español

Discrete ill-posed problems occur frequently in the physical sciences. In this thesis, we present multilevel methods for a particular kind of discrete ill-posed problems, deblurring problems. Multigrid methods are well known as extremely efficient solvers for certain large-scale systems of equations, particularly those that result from the discretizations of partial differential equations and integral equations of the second kind. These have been extensively studied in recent years. However, for ill-posed problems, the classical multigrid approach is not immediately applicable. This work presents new wavelet-based multilevel methods for signal and image restoration problems as well as for blind deconvolution problems. In these methods, we use the orthogonal wavelet transform to define restriction and prolongation operators within a multigrid-type iteration. Specifically, the choice of the Haar wavelet operator has the advantage of preserving matrix structure, such as Toeplitz, between grids, which can be exploited to obtain faster solvers on each level where an edge-preserving Tikhonov regularization is applied. Moreover, when solving a blind deconvolution problem by means of a Structured Total Least Norm formulation, we have again at each level a Structured Total Least Norm problem to solve. We present results that indicate the promise of these approaches for restoration of signals and images with edges as well as restoration of the blurring operator in the case of blind deconvolution problems.

Acknowledgements

First of all, I would like to thank my advisor Misha Kimer for her guidance, support, and encouragement during my six years at Tufts. I feel very privileged to have had the opportunity of working with her and learning from her. Her passion for mathematics and, in particular, for numerical linear algebra has influenced my professional career enormously.

I would also like to express my sincere gratitude to Scott MacLachlan. He contributed with an immense amount of valuable inputs to improve this dissertation. His always open door reflects his willingness to give advice, discuss research, and share his passion on multigrid methods at any time.

I would also like to thank Professors Todd Quinto and Lothar Reichel for being part of the thesis committee. To Professor Dianne O'Leary for valuable suggestions on the Total Least Squares approach that initiated a work that was extended to create this thesis.

In addition, I am grateful to the faculty, staff, and graduate students in the Department of Mathematics for providing a great place to learn and work.

Outside Tufts, I want to thank the Greeks: Nikos, Andreas, Katerina, Eva and Alexander. Specially, to Eva and Alexander who made me redefined the meaning of friendship. Life in Boston would not have been nearly the same without them.

To Mara whose humility, kindness, and generosity made me more human.

I want to thank my family-in-law Ana, Josefina and Guido for their support.

To my father who encouraged me to study mathematics.

Finally, I want to thank Agustín. No doubt I would not be who I am today without him. I am forever grateful for his unconditional love and support. I made it, pibe!!!

“In memory of my mother Teresa Fisdell (1946 - 1982).”

Contents

Abstract	ii
Acknowledgements	iii
List of Figures	vii
List of Tables	ix
1 Introduction to Deblurring and Regularization	1
1.1 Deblurring Problems	1
1.1.1 Signal Restoration	1
1.1.2 Image Restoration	3
1.1.3 Blind Deconvolution	4
1.2 Discrete Linear Ill-Posed Problems	5
1.3 Regularization for the Least Squares Problem	7
1.3.1 Regularization by Truncation	7
1.3.2 Tikhonov Regularization	8
1.3.3 Iterative Regularization Methods	10
1.3.4 Total Variation Regularization	10
1.4 Regularized Total Least Squares	11
1.5 Contributions	12
1.6 Chapter-by-chapter overview	12
2 Introduction to Multilevel Methods	14
2.1 Introduction	14
2.2 Multilevel Basic Framework	14
2.3 Restriction and Prolongation Operators	15
2.4 Solvers	17
3 Multilevel Method for Signal Restoration Problems	21
3.1 Introduction	21
3.2 Algorithm	22
3.2.1 Haar Decomposition	22
3.2.2 Matrix Analysis	24
3.2.3 Pre-smoothing and Coarsest-Grid Correction	25
3.2.4 Residual Correction	27
3.3 Computational issues	28
3.3.1 Coarse-Scale Solve	31

3.3.2	Residual Correction Solve	32
3.4	Numerical Results	33
3.5	Conclusions	36
4	Multilevel Methods for Image Restoration Problems	37
4.1	Introduction	37
4.2	Algorithms	38
4.2.1	Two-Level Methods	38
4.2.2	Multilevel Methods	44
4.2.3	Non-separable Case	47
4.3	Computational Issues	47
4.4	Numerical Results	47
4.4.1	Example 1: Haar Wavelets	48
4.5	Conclusions	51
5	Modified Total Least Norm Problem	52
5.1	Introduction	52
5.2	Motivation	52
5.3	Scaling	54
5.4	Regularized Structured Total Least Norm	55
5.5	Modified Total Least Norm	55
5.6	Alternating Algorithm	56
5.7	Computational Issues	57
5.8	Numerical Results	58
5.8.1	Example 1	58
5.8.2	Example 2	59
5.9	Conclusions	61
6	Multilevel Methods for Blind Deconvolution	62
6.1	Introduction	62
6.2	Algorithm	63
6.3	Computational issues	66
6.3.1	Banded Case	68
6.3.2	Reconstructing E	69
6.4	Numerical Results	70
6.5	Conclusions	72
7	Conclusions and Future Work	73

List of Figures

1.1	Example of signal to be restored	2
1.2	Example of image to be restored	3
1.3	Plot of singular vectors	6
1.4	Discrete Picard condition	6
1.5	Least squares solution	6
1.6	Plot of spectral coefficients of least squares solution	7
1.7	Plot of generalized singular values and vectors	9
1.8	Semi-convergence behavior of LSQR	10
1.9	Total Variation solution	11
2.1	Full weighting of a signal	16
2.2	Wavelet decomposition of a signal	17
2.3	Example of signal	19
2.4	Plot of Jacobi's method solutions applied to free-noise problem	19
2.5	Plot of Jacobi's method solutions applied to problem with noise	19
2.6	Plot of LSQR solutions	20
2.7	Convergence behavior of Jacobi's method and LSQR	20
3.1	Plot of scaling coefficients and wavelet coefficients	23
3.2	Plot of maximum sine of canonical angles	25
3.3	Plot of spectral coefficients of residual after applying LSQR	26
3.4	Discrete Picard condition for coarse-grid	27
3.5	Comparison of different parameters and sections of new method	34
3.6	Comparison of different methods	36
4.1	Example of image	38
4.2	Wavelet decomposition of an image	40
4.3	Spectral analysis for images	41
4.4	Reconstruction of images using partial information	42
4.5	Spectral analysis for images	43
4.6	Diagram of different wavelet-packet basis	46
4.7	Sparsity patter of multilevel matrices	48
4.8	Image restoration example	49
4.9	Two-Level method solutions	49
4.10	Multilevel methods solutions	50
5.1	Example of signal to be restored	52
5.2	Picard condition in TLS model	53
5.3	Example1: Solutions of blind deconvolution problem	56
5.4	Example 1: Solution of alternating method with $c = 1$	59

5.5	Example 1: Solution of alternating method with $c = 0.1$	59
5.6	Example 2: Solutions of blind deconvolution problem	60
5.7	Example 2: Solution of alternating method	61
6.1	Plot of singular values of coarse operator	64
6.2	Example of blind deconvolution problem	70
6.3	Solutions obtained by standard methods	71
6.4	Solutions of different levels	71
6.5	Solution obtained by alternating method	72

List of Tables

2.1	Basic V-Cycle: MGM	15
3.1	Multilevel V-Cycle: MGM	29
3.2	Comparison of different levels and parts of the multilevel method	35
3.3	Comparison of different methods applied to different levels of noise	36
4.1	Two-Level Method	41
4.2	Multilevel V-Cycle: MGM	45
4.3	Comparison of different methods: Relative errors	49
4.4	Comparison of multilevel methods for noise level of 1%	50
4.5	Comparison of multilevel methods for noise level of 5%	50
4.6	Nonzero entries on matrices involved	50
5.1	Alternating Algorithm	57
5.2	Example 1: Relative errors obtained by applying different methods	59
5.3	Example 1: Relative errors obtained after 30 iterations of the alternating method for different values of c	59
5.4	Example 2: Relative errors obtained by applying different methods.	60
5.5	Example 2: Relative errors obtained by applying 30 iterations of the alternating method with different values of c	61
6.1	Multilevel V-Cycle: MGM	65
6.2	Relative errors obtained by solving STLN with different values of c	71
6.3	Relative errors obtained by applying different number of levels.	71
6.4	Alternating Algorithm with V-cycles	72
6.5	Relative errors obtained by applying alternating algorithm with a fixed value of $c = 0.5$	72

Chapter 1

Introduction to Deblurring and Regularization

Linear ill-posed problems in the form of Fredholm integral equations of the first kind occur frequently in the physical sciences [18]. They model inverse problems; that is, situations where hidden information is computed from external observations. In this thesis, we focus on a particular kind of inverse problem, deblurring problems. These signal and image restoration problems occur in a variety of important numerical applications such as remote sensing, medical imaging, and astronomical imaging.

1.1 Deblurring Problems

1.1.1 Signal Restoration

A signal restoration problem can be modeled by a first-kind Fredholm integral equation:

$$\int_{\alpha_1}^{\alpha_2} K(s, t)f(t)dt = g(s), \quad \beta_1 \leq s \leq \beta_2, \quad (1.1)$$

where the function K , called the *kernel*, is a known function of two variables s and t , and the right-hand side g is also known, while f is the unknown function that we wish to compute.

In particular, when the kernel $K(s, t)$ is *spatially invariant*, that is, its effect depends only on the distance between s and t , then the preceding equation represents a convolution integral, where $K(s, t) = K(s - t)$. In this case, we can say that $g(s)$ is the result of convolving $K(s)$ and $f(s)$.

The discretization of such an integral equation gives a *discrete ill-posed problem*¹ [26], taking the form

$$Ax^{true} = b^{true}, \quad (1.2)$$

¹In Section 1.2, we will explain what we mean by discrete ill-posed problem.

where the matrix A is a discretization of the integral operator, and the vectors x^{true} and b^{true} are the discretized versions of the continuous functions f and g . Then, given A and b^{true} , the corresponding discrete inverse problem is to recover x^{true} .

In fact, instead of having b^{true} , we have measured data, b , that contains additive noise, e , due to measurement and/or approximation error. In this thesis, we assume that e is Gaussian white noise. Then, we have the system of equations

$$Ax \approx b = b^{true} + e, \quad (1.3)$$

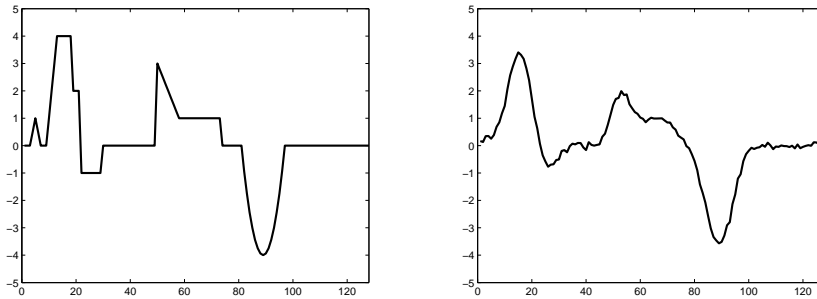


FIGURE 1.1: True solution x^{true} that represents the signal we want to recover (left), and blurred, noisy right-hand side b that represents measured data (right).

where the matrix A acts as a *blurring* operator on the “signal” x^{true} , and b denotes the blurred, noisy signal (see Figure 1.1). We are particularly interested in the case when the signal has “edges” that we want to recover. In this thesis, we restrict ourselves to the case when $A \in \mathbb{R}^{m \times m}$, with $m = 2^k$ for some integer k . Also, we will assume that the blurring function K is spatially invariant. Consequently, the matrix A will have a special structure such as being *Toeplitz*, *circulant*, or *Hankel*, depending on the imposed boundary conditions [29]. In this thesis, we consider the case of zero boundary conditions, which involves a Toeplitz matrix A whose elements are constant along diagonals, that is,

$$A = \begin{bmatrix} t_0 & t_{-1} & t_{-2} & \dots & t_{-(m-1)} \\ t_1 & t_0 & t_{-1} & \dots & t_{-(m-2)} \\ t_2 & t_1 & t_0 & \dots & t_{-(m-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{m-1} & t_{m-2} & t_{m-3} & \dots & t_0 \end{bmatrix}.$$

The vector $t = (t_{-(m-1)}, \dots, t_{-1}, t_0, t_1, \dots, t_{m-1})$ is called the *Toeplitz-vector* that generates A .

1.1.2 Image Restoration

An image restoration problem modeled by a convolution kernel K has the form

$$\int_{\alpha_1}^{\alpha_2} \int_{\gamma_1}^{\gamma_2} K(s-t, s'-t') f(t, t') dt dt' = g(s, s'). \quad (1.4)$$

After discretizing (1.4) we arrive at a linear system

$$Ax^{true} = b^{true}, \quad (1.5)$$

where the matrix A acts as a blurring operator on the true image represented by the vector x^{true} (see Figure 1.2), and the vector b^{true} denotes the blurred image. A two-dimensional (2D) image can be stored in a matrix of numbers between 0 and 255, where each entry corresponds to the color of each pixel. Then, both vectors x^{true} and b^{true} contain the elements of the 2D images by stacking its columns. We can state our image restoration problem as finding x such that

$$Ax \approx b = b^{true} + e, \quad (1.6)$$

where b is the blurred, noisy image (see Figure 1.2) and e is additive Gaussian noise. In this work, we

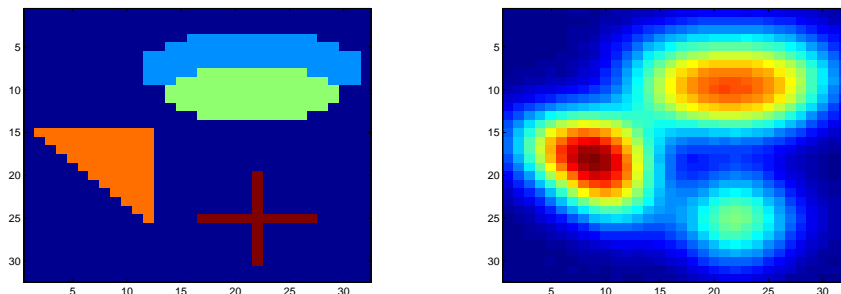


FIGURE 1.2: True solution x^{true} that represents the image we want to recover (left) and the blurred noisy right-hand side b that represents the captured image (right).

restrict ourselves to the case when images are of size $m \times m$, with $m = 2^k$ for some integer k . Then, the corresponding matrix A has size $m^2 \times m^2$. For example, a 256×256 image has a blurring matrix of size $65,536 \times 65,536$. These dimensions give us an idea of why storage and computational efficiency are of great importance in dealing with such problems. In particular, the complexity of restoration algorithms can be reduced by taking into account problem-based characteristics such as structured matrices. For instance, when considering a problem with zero boundary conditions, the matrix A has Block-Toeplitz-Toeplitz-Block (BTTB) structure [29], that is, an m by m block-Toeplitz matrix with each block A_i

being an m by m Toeplitz matrix,

$$A = \begin{bmatrix} A_0 & A_{-1} & A_{-2} & \dots & A_{-(m-1)} \\ A_1 & A_0 & A_{-1} & \dots & A_{-(m-2)} \\ A_2 & A_1 & A_0 & \dots & A_{-(m-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{m-1} & A_{m-2} & A_{m-3} & \dots & A_0 \end{bmatrix}.$$

Other structured matrices arise under different boundary conditions. For periodic boundary conditions, for example, the resulting matrix is block circulant with circulant blocks (BCCB) [29].

A special case occurs when K is separable, i.e. $K(s-t, s'-t') = k(s-t)w(s'-t')$. Moreover, we shall assume for simplicity that $k = w$. If $\tilde{A} \in \mathbb{R}^{m \times m}$ is the discretization of the integral operator k , then the corresponding discrete operator A of the kernel K is defined by

$$A = \tilde{A} \otimes \tilde{A},$$

where \otimes denotes the *Kronecker product* [17]. The Kronecker product of matrices B and C is given by

$$B \otimes C = \begin{bmatrix} b_{11}C & b_{12}C & \dots & b_{1m}C \\ b_{21}C & b_{22}C & \dots & b_{2m}C \\ \vdots & \vdots & & \vdots \\ b_{m1}C & b_{m2}C & \dots & b_{mm}C \end{bmatrix}.$$

Notice that if \tilde{A} is Toeplitz, then $\tilde{A} \otimes \tilde{A}$ is BTTB.

1.1.3 Blind Deconvolution

In the deblurring problems presented above, we assume that the operator A is known exactly. In many real-life applications, however, this is not a good assumption. For example, the blurring could be due to atmospheric conditions, to the motion of the object, or due to the motion of the person taking the picture. Therefore, we shall also study how, when given a blurred, noisy image, to recover an approximation to the true image without knowing the blurring operator exactly. If there is noise/errors in A as well as b , the model to be considered is

$$(A + E)x \approx b = b^{true} + e, \quad b^{true} = (A + E)x^{true}, \quad (1.7)$$

where e denotes unknown noise corrupting the perfect data b^{true} , E is an unknown perturbation of the assumed known blurring operator A , and b denotes the blurred, noisy signal.

1.2 Discrete Linear Ill-Posed Problems

The problems described above are discrete linear ill-posed problems, that is, problems defined by ill-conditioned linear systems $Ax = b$ with certain characteristics we shall now describe. In this thesis, we focus on finding numerical solutions of these problems. From now on, as an example, we will use the data that is described in the numerical examples section of Chapter 2. For simplicity, let $A \in \mathbb{R}^{m \times m}$ have full rank. Consider the *singular value decomposition* (SVD) [17] of A ,

$$A = U\Sigma V^T = \sum_{i=1}^m u_i \sigma_i v_i^T,$$

where $U = (u_1, \dots, u_m) \in \mathbb{R}^{m \times m}$ and $V = (v_1, \dots, v_m) \in \mathbb{R}^{m \times m}$ are matrices with orthonormal columns, $U^T U = V^T V = I$, and

$$\Sigma = \text{diag}(\sigma_1, \dots, \sigma_m), \quad \sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_m \geq 0.$$

The numbers σ_i are called the *singular values* of A and the vectors u_i and v_i are the *left* and *right singular vectors* of A , respectively.

The main characteristics of our problems are given in terms of the SVD of A :

- its singular vectors v_i become more and more oscillatory as i increases (see Figure 1.3),
- its singular values σ_i decay gradually to zero with no particular gap (see Figure 1.4),
- the *discrete Picard condition* [24, 26] is verified: the absolute value of the spectral coordinates of the true right-hand side b^{true} , $|u_i^T b^{true}|$, decreases to zero faster on average than the singular values, σ_i (see Figure 1.4).

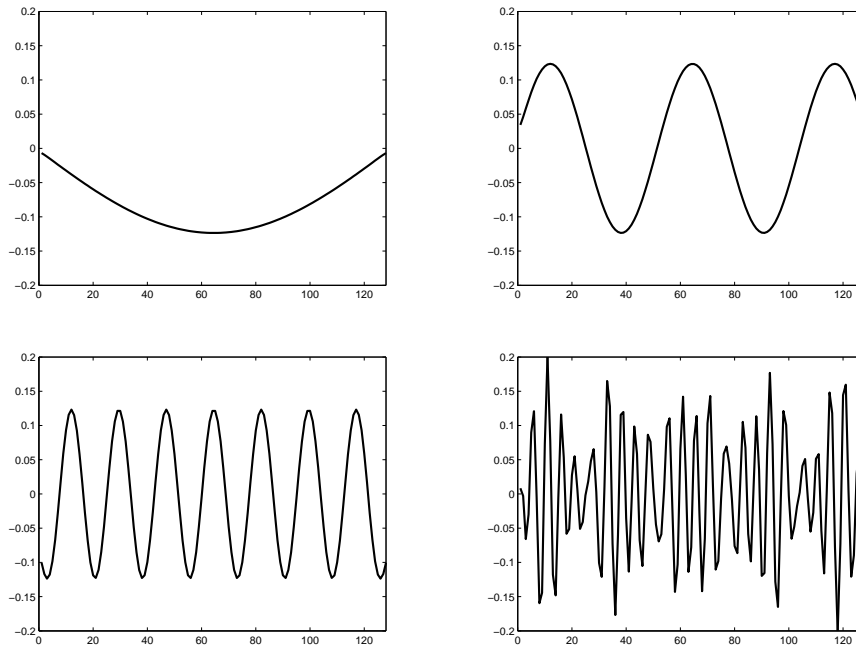
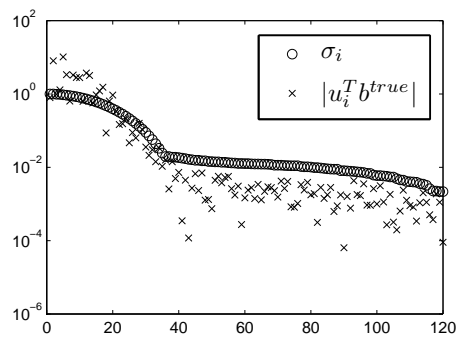
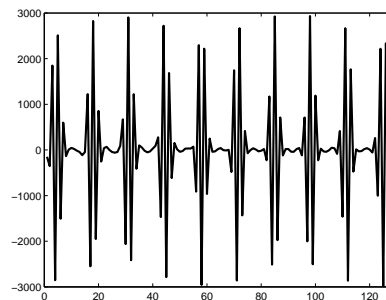
These features are essential to understand the difficulties we encounter when trying to solve restoration problems, as well as to help us to develop strategies to obtain their approximate solutions. If we solve the system (1.3) exactly, we obtain a very noisy solution, x^{LS} , that is not related to the true solution, x^{true} (see Figures 1.5 and 1.1 for comparison). To understand what goes wrong, we can write both solutions x^{LS} and x^{true} in terms of the SVD of A . Then, we have that

$$x^{true} = \sum_{i=1}^m \frac{u_i^T b^{true}}{\sigma_i} v_i$$

and

$$\begin{aligned} x^{LS} &= \sum_{i=1}^m \frac{u_i^T b}{\sigma_i} v_i = \sum_{i=1}^m \frac{u_i^T (b^{true} + e)}{\sigma_i} v_i \\ &= \sum_{i=1}^m \frac{u_i^T b^{true}}{\sigma_i} v_i + \sum_{i=1}^m \frac{u_i^T e}{\sigma_i} v_i = x^{true} + \text{noise}. \end{aligned}$$

Since we assume that the noise, e , is white, the terms $|u_i^T e|$ are more or less constant for all values of i . On the other hand, since b^{true} satisfies the discrete Picard condition, $|\frac{u_i^T b^{true}}{\sigma_i}|$ becomes smaller than

FIGURE 1.3: Some right singular vectors of A : v_1 , v_5 , v_{15} , and v_{50} FIGURE 1.4: Discrete Picard condition: $|u_i^T b^{true}|$ decreases faster than σ_i FIGURE 1.5: Least Squares solution, x^{LS}

$|\frac{u_i^T e}{\sigma_i}|$ for $i \leq j$ for some index $1 < j < m$ (see Figure 1.6). This is the point at which noise starts to dominate the solution, and because of the ordering of the singular values, this happens for very oscillatory singular vectors, making the *least squares* solution, x^{LS} , very oscillatory and useless. Thus, we seek an approximation to the solution, x^{true} , by methods that filter out the noise, e , while obtaining as much information as possible from the measured data, b .

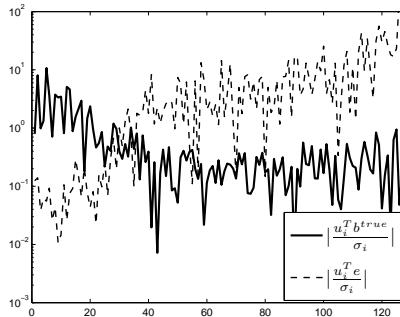


FIGURE 1.6: Absolute value of coefficients ($\frac{u_i^T b^{true}}{\sigma_i}$ and $\frac{u_i^T e}{\sigma_i}$) of x^{LS} as linear combination of singular vectors v_i .

1.3 Regularization for the Least Squares Problem

We can obtain an approximation to the solution x^{true} by solving a nearby problem that is not so ill-posed. This method is called *regularization*. Regularization methods [26] include regularization by truncation, Tikhonov regularization, early termination of certain iterative methods, such as conjugate gradient on the normal equations (CGLS), or Total Variation regularization.

1.3.1 Regularization by Truncation

When we look at the expansion of x^{LS} , one idea is to truncate the sum just before the noise starts corrupting the solution at $i = j$, that is, we can get an approximate solution of x^{true} by defining

$$x^{TSVD} = \sum_{i=1}^j \frac{u_i^T b}{\sigma_i} v_i.$$

This solution is called the *truncated singular value decomposition* (TSVD). It also can be obtained by solving the problem

$$\min_x \|A_j x - b\|_2^2, \quad (1.8)$$

where $A_j = \sum_{i=1}^j u_i \sigma_i v_i^T$. The truncation parameter, j , is called a *regularization parameter*. There is a large area of research on developing methods to choose an “optimal” regularization parameter, and we do not investigate such approaches in this thesis. Some well-known methods to estimate it are the discrepancy principle, L-curve [27, 26], and generalized cross validation (GCV) [26].

1.3.2 Tikhonov Regularization

If additional information about the solution is known (for example, that the solution is smooth), then, as an alternative to TSVD, we can solve the regularized problem

$$\min_x \{ \|Ax - b\|_2^2 + \lambda^2 \Omega(Lx) \}, \quad (1.9)$$

where $\lambda > 0$ is the regularization parameter and L is a *regularization operator* used to enforce a priori information about the solution. This method is called *Tikhonov regularization*. For example, in a one-dimensional problem, smoothness is often enforced with L being a discrete derivative operator such as

$$\Omega(Lx) = \|Lx\|_2^2 \quad \text{with} \quad L = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & -1 & 1 \end{pmatrix}. \quad (1.10)$$

To understand why (1.9) produces a reasonable approximate solution with 2-norm constraint, it is convenient to look at the *generalized singular value decomposition* (GSVD) [17, 26] of the matrix pair, (A, L) . First, consider the *reduced QR-factorization* of the stacked matrix:

$$\begin{bmatrix} A \\ L \end{bmatrix} = QR = \begin{bmatrix} Q_A \\ Q_L \end{bmatrix} R.$$

We have $A = Q_A R$ and $L = Q_L R$. Let $Q_A = \tilde{P} \tilde{C} \tilde{W}^T$ and $Q_L = \tilde{P} \tilde{S} \tilde{W}^T$ be the CS decomposition [17] of $\{Q_A, Q_L\}$, where $\tilde{P} \in \mathbb{R}^{m \times m}$, $\tilde{P} \in \mathbb{R}^{p \times p}$ and $\tilde{W} \in \mathbb{R}^{n \times n}$ are orthogonal matrices; $\tilde{C} \in \mathbb{R}^{m \times n}$ and $\tilde{S} \in \mathbb{R}^{p \times n}$ are block diagonal (not necessarily square) matrices satisfying $\tilde{C}^T \tilde{C} + \tilde{S}^T \tilde{S} = I$. The existence of the CS decomposition can be found in ([46], 22.1). Then the GSVD of $\{A, L\}$ is defined as follows

$$A = \tilde{P} \tilde{C} \underbrace{\tilde{W}^T R}_{G^{-1}} \quad \text{and} \quad L = \tilde{P} \tilde{S} \underbrace{\tilde{W}^T R}_{G^{-1}}, \quad (1.11)$$

where for convenience, we have ordered the entries in \tilde{C}, \tilde{S} so that the diagonal elements of \tilde{C} and \tilde{S} are ordered such that

$$1 \geq c_1 \geq \dots \geq c_n \geq 0, \quad 0 \leq s_1 \leq \dots \leq s_n \leq 1.$$

We note that this ordering is the opposite of the order usually found in the literature. We do this so that the generalized singular values are in decreasing order, similar to the standard SVD. Using this decomposition, the Tikhonov solution can be written

$$x^\lambda = \sum_{i=1}^n \frac{c_i^2}{c_i^2 + \lambda^2 s_i^2} \frac{\tilde{p}_i^T b}{c_i} g_i. \quad (1.12)$$

The values $|\tilde{p}_i^T b^{true}|$ should be decaying on average faster than c_i 's until we reach an index, j , such that noise components become dominant (see Figure 1.7). At that point, $|\tilde{p}_i^T b|$ becomes roughly constant. Since the c_i 's are decreasing and the noise is white, the signal is contained primarily in the subspace

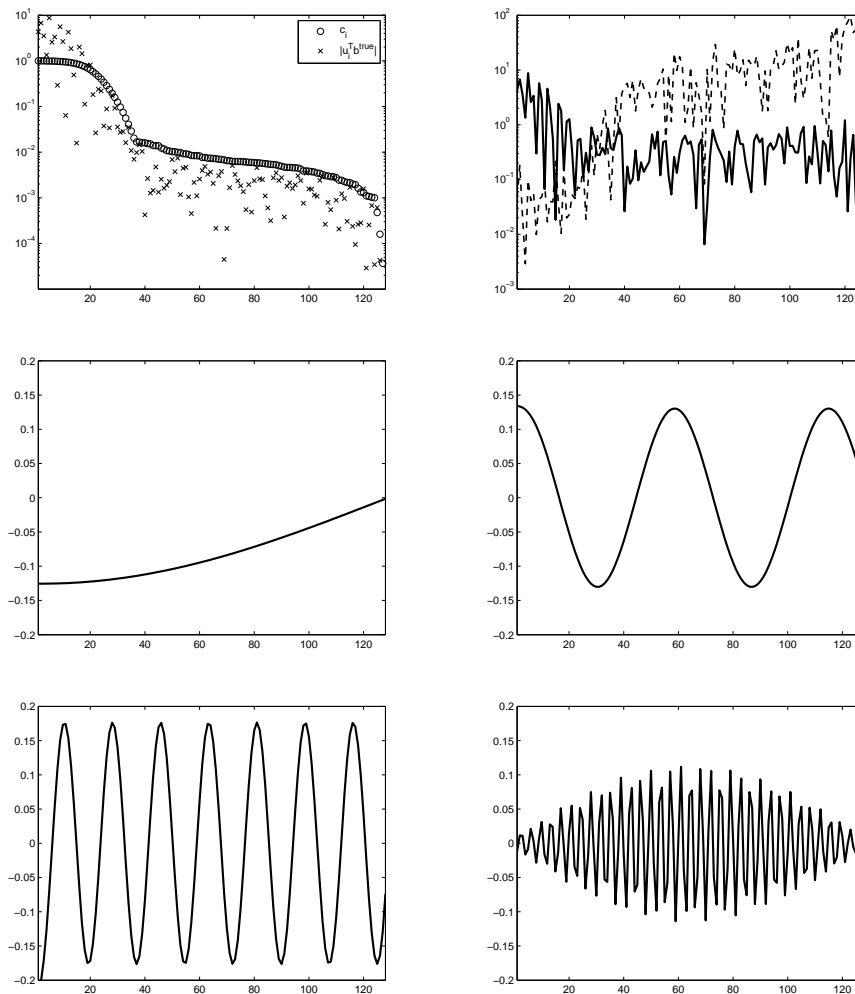


FIGURE 1.7: Values of c_i and $|\tilde{p}_i^T b^{true}|$ (top left). Comparison between $|\frac{u_i^T b^{true}}{c_i}|$ and $|\frac{u_i^T e}{c_i}|$ (Top right). Vectors g_1 (middle left), g_5 (middle right), g_{15} (bottom left), and g_{50} (bottom right).

spanned by the g_i 's for small i , and a good value of λ should have the effect of filtering the terms in the sum for larger values of i .

We can define the *Truncated GSVD* (TGSVD) [26] solution

$$x^{TGSVD} = \sum_{i=1}^j \frac{\tilde{p}_i^T b}{c_i} g_i,$$

and expect that this is a better approximation to x^{true} because the dominant noise components have been filtered out.

1.3.3 Iterative Regularization Methods

When the matrix A is very large, which is the case when solving 2D or 3D problems, computing the SVD to obtain the TSVD solution, x^{TSVD} , or solving a minimization problem like (1.9), can be computationally infeasible. In this case, we resort to iterative methods. In order to compute regularized solutions of (1.3), we compute a few steps of an iterative regularization method [22]. Such iterative methods present a *semi-convergent* behavior with respect to the relative error, $\|x^j - x^{true}\|/\|x^{true}\|$, where x^j is the approximate iteration at the j -th iteration. That is, the relative error decreases for a certain number of iterations, and after that, it starts increasing. In other words, if we stop the method when the relative error is small, we will obtain a good regularized solution. If we wait too many iterations to stop, we will incorporate noise into the solution. Therefore, j plays the role of regularization parameter. Some of the methods that are known to have this semi-convergent behavior are the *Conjugate Gradient* (CG) [26, 53], *Generalized Minimal Residual* (GMRES) [8, 53], *Minimal Residual* (MINRES) [34, 53], and LSQR [26, 42] algorithms. For example, Figure 1.8 shows the semi-convergence of LSQR and the corresponding solutions for different stopping number of iterations: before the optimal parameter j , at the optimal, and after. We see that once we pass the optimal parameter, the solution obtained is corrupted.

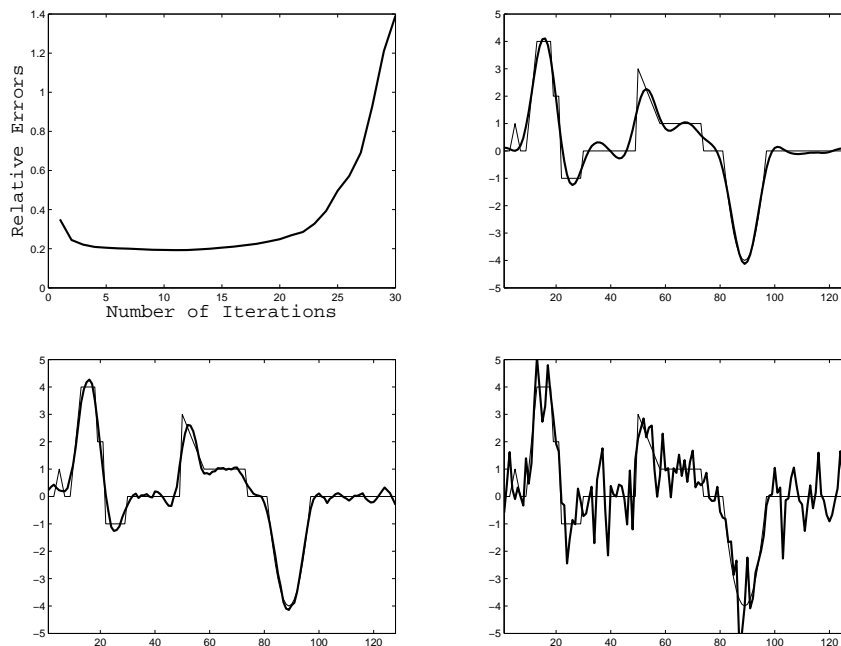


FIGURE 1.8: Semi-convergence of LSQR method (top left), and LSQR solutions with $j = 3$ (top right), 11 (bottom left), and 25 (bottom right). In this case, $j = 11$ is the optimal parameter.

1.3.4 Total Variation Regularization

So far we have described methods that return smooth solutions. If the signal/image we want to recover contains edges, the *Total Variation* (TV) [56] regularization method is a very common choice. It consists

of using the TV functional as a regularization term. For a one-dimensional function u , the TV functional is defined by

$$J_{TV}(u) = \int_0^1 \left| \frac{du}{dt} \right| dt$$

and for a two-dimensional function u is

$$J_{TV}(u) = \int_0^1 \int_0^1 |\nabla u| ds dt, \quad |\nabla u|^2 = \left(\frac{\partial u}{\partial s} \right)^2 + \left(\frac{\partial u}{\partial t} \right)^2.$$

For a discrete 1D problem, the TV solution is obtained by solving

$$\min_x \{ \|Ax - b\|_2^2 + \lambda \|Lx\|_1 \}, \quad (1.13)$$

where L is the discrete approximation of the first derivative operator (1.10). Notice that (1.13) is a form of Tikhonov regularization, where the regularization term uses the 1-norm. This allows the method to recover solutions with discontinuities, and therefore, it does a better job of recovering edges (see Figure 1.9). Another edge-preserving method is PP-TSVD [28]. However, these approaches are known

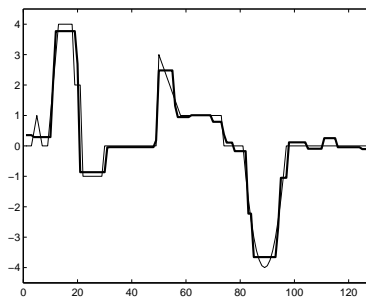


FIGURE 1.9: The TV solution

to be computationally expensive. This motivates our study, that is, to look for computationally efficient algorithms that can recover edges.

1.4 Regularized Total Least Squares

When trying to solve the problem (1.7), we also encounter difficulties. For example, if we assume that our approximate deconvolution operator A is invertible, the solution of (1.7) is $x = x^{LS}$ and $E = 0$. As discussed in Section 1.2, though, the least squares solution is worthless and regularization is needed. If Tikhonov regularization is applied, one must solve the following problem

$$\min_{E,x} \{ \|E\|_F^2 + \|(A + E)x - b\|_2^2 + \lambda^2 \Omega(Lx) \}, \quad (1.14)$$

where $\|\cdot\|_F$ is the *Frobenius norm* [17]. Typically, $\Omega(Lx) = \|Lx\|_2^2$ and this problem is known as the *Tikhonov-Regularized Total Least Squares Problem* [30]. This is a minimization problem with respect to

the unknown perturbation in the blurring operator and the desired solution. Moreover, it is non-convex and non-linear. As such, it is a much more difficult and computationally expensive problem to solve than the previous problems. There are also other regularization approaches to solve (1.7). One example is the *Truncated* TLS [48], which involves the SVD of the matrix $[A|b]$. This method is applied to rectangular $m \times n$ matrices with $m > n$, which is not the case we consider in this thesis. Another approach is the Regularized TLS (R-TLS) method presented in [14]

$$\min_{E,x} \{ \|E\|_F^2 + \|(A + E)x - b\|_2^2 \} \quad \text{subject to } \|Lx\|_2^2 \leq \delta. \quad (1.15)$$

In this latter article, the authors emphasize the importance of choosing a regularization operator $L \neq I$. Otherwise, the solution of (1.15) is the same solution of the regularized least squares problem (1.9) when $\Omega = \|\cdot\|_2^2$.

1.5 Contributions

The main contribution of this dissertation consists of presenting new multilevel methods for deblurring problems. More specifically, this dissertation introduces the following.

1. A theoretical analysis of coarse-scale equations corresponding to discrete ill-posed problems, with particular analysis on coarsening produced by the Haar wavelet transform.
2. Multilevel approaches for signal/image restoration problems that have the capability to recover some edge information at a smaller expense than other traditional methods.
3. Theoretical results of Haar wavelet transform of Toeplitz matrices and their applications for efficient implementation of Newton-like solvers.
4. A reformulation of the regularized total least norm problem as a modified regularized total least norm problem, where *modified* refers to the incorporation of initial guesses into the model, as well as a scaling factor.
5. Extensions of multilevel approaches and computational techniques to modified structured total least norm problems, with emphasis on banded Toeplitz matrices.
6. A new alternating method for solving blind deconvolution problems that combines the new modified regularized total least norm problem with its corresponding multilevel extension, offering efficiency and qualitatively better solutions than up-to-date methods.

1.6 Chapter-by-chapter overview

The structure of the dissertation is as follows. In Chapter 2, we give an introduction to multilevel methods and motivate the development of multilevel methods for discrete ill-posed problems. In Chapter 3, we

present an edge-preserving multilevel algorithm for signal restoration problems and address some of the computational issues. Chapter 4 presents several extensions of the algorithm to the 2D case, the image deblurring problem. Chapter 5 contains an analysis of a modified total least norm problem, and Chapter 6 presents a multilevel methods for this problem. Chapter 7 summarizes conclusions and future work.

Chapter 2

Introduction to Multilevel Methods

2.1 Introduction

Multigrid methods [7, 21, 38, 55] are well known as extremely efficient solvers for certain large-scale systems of equations, particularly those that result from the discretizations of certain classes of partial differential equations and integral equations of the second kind. These have been extensively studied in recent years. However, for ill-posed problems, the classical multigrid approach is not immediately applicable. In this chapter, we define some terminology used throughout the thesis and give some motivation for developing new multilevel approaches for deblurring problems.

2.2 Multilevel Basic Framework

The main idea of a multilevel method is, for a given system $Ax = b$, to define a sequence of systems of equations decreasing in size,

$$A^i x^i = b^i, \quad 0 \leq i \leq n,$$

where the superscript i denotes the i -th level we have processed. In particular, $i = 0$ corresponds to the finest level ($A^0 = A$), and $i = n$ to the coarsest level. We call these systems (for $i > 0$) *coarse-grid equations*. The *restriction operator*, R , and the *prolongation operator*, P , define the transfers from finer to coarser grids and vice versa. The operator at each level can be defined by

$$A^{i+1} = R^i A^i P^i,$$

or can be obtained by direct discretization.

Then, the solution process consists of computing an approximation to the solution of the fine-scale system (often called pre-smoothing), computing the residual, transferring the residual from fine to coarser grid, solving the coarsest-grid correction equation (coarsest-grid correction), interpolation of corrections from

TABLE 2.1: Basic V-Cycle: MGM

Input: A^i, b^i
Output: x^i
1. If $i = n$
2. Solve $A^i x^i = b^i$ (Coarsest-grid Correction)
3. Else
4. Solve $A^i x_{pre}^i = b^i$ (Pre-smoothing)
5. $x^i = x_{pre}^i$
6. $r^i = b^i - A^i x^i$
7. $b^{i+1} = R^i r^i$
8. $y^{i+1} = MGM(A^{i+1}, b^{i+1})$
9. $x^i = x^i + P^i y^{i+1}$
10. $r^i = b^i - A^i x^i$
11. Solve $A^i x_c^i = r^i$ (Residual Correction)
12. $x^i = x^i + x_c^i$
13. End If

the coarse grid, and computing an approximation to the solution of the updated residual equation on the finer grid (residual correction). The residual-correction step corresponds to post-smoothing in multigrid literature. We summarize the method recursively in Table 2.1. In particular, when $n = 1$, we have a Two-Level method.

In order to give a complete definition of a multilevel method, we need to define the restriction and prolongation operators, and also what we mean by pre-smoothing, coarsest-grid correction and residual correction; that is, we need to define how we “solve” the systems on lines 2, 4, and 11 of the method in Table 2.1. We address these issues in the next sections.

2.3 Restriction and Prolongation Operators

To move between grids, we need to define restriction and prolongation operators. We start by defining the standard operators found in the multigrid literature to motivate the search of new type of operators. The following is the restriction operator called the *full weighting* matrix in 1D [7],

$$R = \frac{1}{4} \begin{pmatrix} 1 & 2 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & 1 & 2 & 1 \end{pmatrix}. \quad (2.1)$$

The corresponding prolongation operator P is $P = 2R^T$. First, we see that, by doing full weighting, we smooth the original solution x^{true} (see Figure 2.1), losing some important edge features as we move between grids, which is counter to what we desire to recover. Furthermore, consider a two-level V-cycle without pre-smoothing, that is, the algorithm of Table 2.1 with $n = 1$ and $x_{pre}^0 = 0$. Assume that our approximation y (obtained by solving the coarsest-grid correction) is very good. Then, we see that

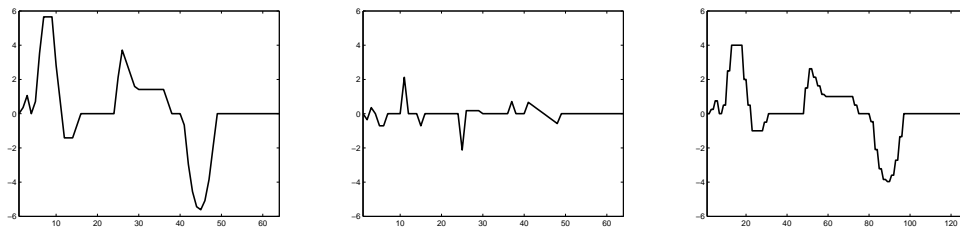


FIGURE 2.2: The scaling coefficient vector \hat{x}_1 (left), the wavelet coefficient vector \hat{x}_2 (middle), and the approximate solution $\hat{x} = W[\hat{x}_1; 0]$ (right).

We consider two DWT: the Haar Wavelet transform (HWT), where $(h_0, h_1) = (\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$ and $(g_0, g_1) = (\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2})$. In the interest of space, we will not give detailed background on wavelets here, but refer the interested reader to [6] to see why wavelets are useful for decomposing signals with edges. However, we will show in the next chapter that the HWT has nice properties for a multilevel approach.

In Chapter 4, we present a multilevel method for image restoration problems. For that method, we shall use a 2D DWT. The 2D DWT, \bar{W} , can be defined as a Kronecker product of the 1D DWT W , that is, $\bar{W} = W \otimes W$. To get a convenient matrix representation we use $\bar{W} = \bar{P}W$, where \bar{P} is a permutation such that

$$\bar{P}\bar{W}^T = \begin{bmatrix} W_1^T \otimes W_1^T \\ W_2^T \otimes W_1^T \\ W_1^T \otimes W_2^T \\ W_2^T \otimes W_2^T \end{bmatrix}. \quad (2.3)$$

With this DWT, we decompose the image x in four terms,

$$x = (W_1 \otimes W_1)\hat{x}_1 + (W_1 \otimes W_2)\hat{x}_2 + (W_2 \otimes W_1)\hat{x}_3 + (W_2 \otimes W_2)\hat{x}_4.$$

More details of this image decomposition will be discussed in Chapter 4.

2.4 Solvers

Once we have defined restriction and prolongation operators, we need to define how we “solve” the systems on lines 2, 4, and 11 of the method in Table 2.1 to have a complete description of a multilevel approach. For multilevel methods developed to solve systems arising from discretizations of partial differential equations, the most common choices to approximate solutions of these systems are simple iterative methods such as Gauss-Seidel or Jacobi [53]. However, an important difference between the discrete operators arising from discretization of partial differential equations and first-kind Fredholm integral equations is that, for the latter, vectors in the span of the singular vectors corresponding to the largest singular values are smooth, while the singular vectors corresponding to the small singular values represent the “high frequency” information. Since the integral operator is a smoothing operator, but the noise vector is assumed to be white, the expansion coefficients of b^{true} decay on average faster than the

singular values while the expansion coefficients of the noise vector are roughly constant in absolute value. Thus, the noise perturbs the high frequency components of the right-hand side and makes it impossible to recover x^{true} through the use of typical multilevel ‘smoothers’ such as Jacobi or Gauss-Seidel, because they can be shown to incorporate noise into the solution immediately. To see this, consider the special case of an ill-conditioned symmetric Toeplitz matrix normalized such that $\|A\|_2 = \sigma_1 = 1$. The Jacobi iteration, in this case, is defined by

$$x_{k+1} = (I - cA)x_k + cb,$$

where $c = 1/a_{11}$, or equivalently by $x_k = \sum_{j=1}^k (I - cA)^{j-1} cb$ if $x_0 = 0$. Let $A = U\Sigma V^T$ be the SVD of A . Then, we can look at the spectral coordinates of x_k ,

$$(U^T x_k)_i = c \left[\sum_{j=1}^k \left(1 - \frac{\sigma_i}{a_{11}}\right)^{j-1} \right] (U^T b)_i$$

and see that for a fixed k , the solution contains noise (included inside the vector b). To see this, consider i large (σ_i small), so that $(1 - \frac{\sigma_i}{a_{11}})^{j-1} \approx 1$ and $(U^T b)_i \approx (U^T e)_i$, where e represents noise. In particular, the i -th coordinate of x_k is dominated by noise. Therefore, we do not want to apply these methods. Instead, we want to apply iterative regularization methods such as LSQR. A concrete example will help us to understand why we need different approaches than the standard ones for PDEs.

Numerical Example

We define a discrete ill-posed problem to show what the effect of noise is when applying the Jacobi iterative method to it. We consider a 128×128 positive definite matrix A whose condition number is $\kappa(A) = 2.3 \times 10^{11}$. The exact solution, represented by x^{true} , is the vector of length 128 shown in Figure 2.3. The noise-free blurred signal, represented by b^{true} , is computed as $b^{true} = Ax^{true}$ (see Figure 2.3). The elements of the noise vector e are normally distributed with zero mean, and the standard deviation is chosen such that $\frac{\|e\|_2}{\|b^{true}\|_2} = 0.05$. The noisy right-hand side of our system is defined by $b = b^{true} + e$; (see Figure 2.3).

We applied a weighted Jacobi method [53] to the noise-free system $Ax = b^{true}$. We see that the method converges to the solution (see Figure 2.4). However, when we applied a weighted Jacobi method to the system $Ax = b$ (with right-hand side containing noise, e), we see that even after 1 iteration, the approximate solution has some oscillatory parts (see Figure 2.5). After 10, we clearly see the effect of noise in the approximation and after 100 iterations the solution is meaningless. Therefore, Jacobi is not appropriate as a solver for discrete ill-posed problems. For that reason, we will use LSQR in our algorithms. Figure 2.6 shows that the approximate solutions given by LSQR are not immediately contaminated with noise as Jacobi. Moreover, LSQR has the semi-convergence property that we mention in Section 1.3.3. We can see in Figure 2.7 that while LSQR reduces the relative error $\|x^{LSQR} - x^{true}\|_1 / \|x^{true}\|_1$ first, and then increases it, Jacobi method increases the relative error $\|x^J - x^{true}\|_1 / \|x^{true}\|_1$ since the first iterations.

Multigrid methods were developed to accelerate the convergence of methods such as Jacobi [7]. In this context, not only is Jacobi an inappropriate method for our fine-grid problem, but also we do not want to follow the standard multigrid idea of improving the convergence over the subspace related to small

singular values because this is where the noise dominates. Therefore, we follow the multilevel method described at the beginning of this chapter, but we are very careful in how we define the restriction and prolongation operators as well as what solvers we apply.

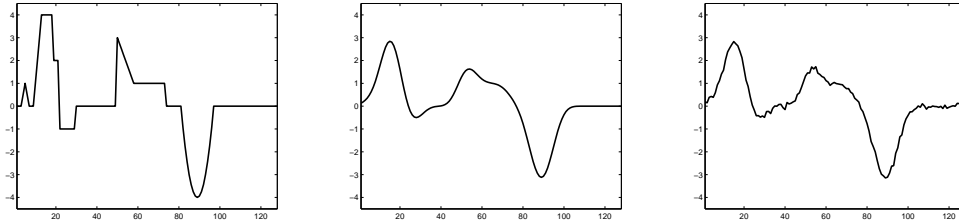


FIGURE 2.3: The true solution x^{true} (left), the corresponding right-hand side $b^{true} = Ax^{true}$ (middle), and the blurred, noisy right-hand side $b = b^{true} + e$ (right).

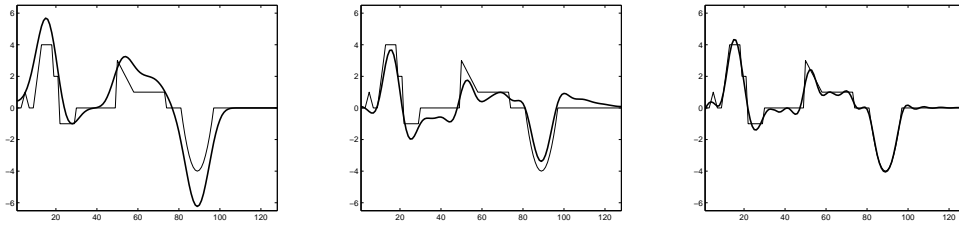


FIGURE 2.4: Approximate solutions of x given by 1 (left), 10 (middle) and 100 (right) iterations of Jacobi applied to the system $Ax = b^{true}$.

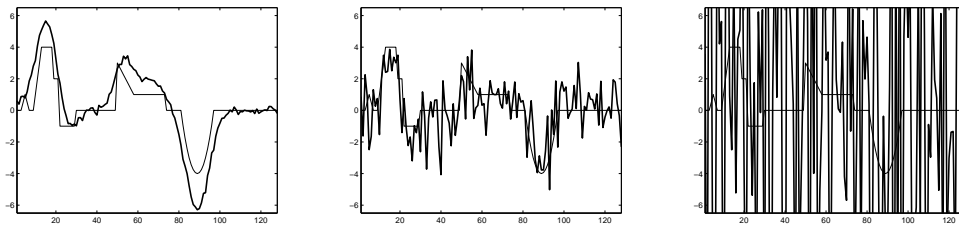


FIGURE 2.5: Approximate solutions of x given by 1 (left), 10 (middle) and 100 (right) iterations of Jacobi applied to the system $Ax = b$.

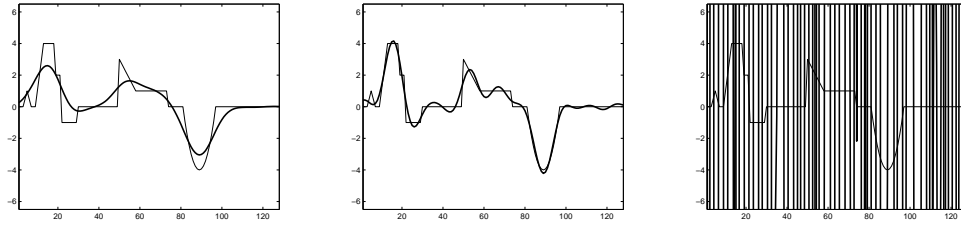


FIGURE 2.6: Approximate solutions of x given by 1 (left), 10 (middle) and 100 (right) iterations of LSQR applied to the system $Ax = b$.

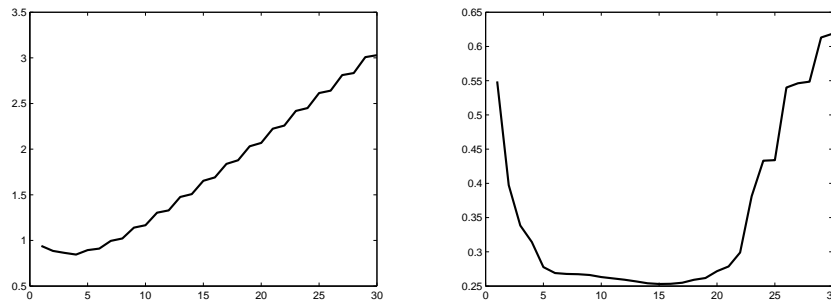


FIGURE 2.7: Convergence: Relative errors of Jacobi (left) and LSQR (right) applied to the system $Ax = b$, for the first 30 iterations.

Chapter 3

Multilevel Method for Signal Restoration Problems

3.1 Introduction

Multigrid methods have already been considered for solving ill-posed problems [11, 23, 31, 35, 40, 49, 50]. The two-level and multilevel methods presented in [23, 50] are applied directly to the regularized Tikhonov-form of the problem. That is, the regularization of the problem is obtained by the starting model. The approaches presented in [11, 31, 35, 40, 49], on the other hand, present multilevel methods applied to the unregularized problem. Then, the regularization is obtained by the method itself. We present a new multilevel approach that follows the overall ideas found in [11], where the authors used a standard multigrid V-cycle scheme and standard (with respect to the PDE literature) prolongation and coarse-grid correction operators. The smoothers advocated were those such as CGLS, which are known to capture the smooth component of the signal at each level without mixing in noise (high frequency information) as long as they are terminated after a small number of iterations. However, their approach did not include a residual-correction (post-smoothing) step on the upwards part of the V-cycle, and rather used only interpolation to move back up to the fine grid. They advocated very few iterations at each level (i.e. over-regularization), and multiple V-cycles. As a result of this choice of smoother, in many cases, the edges in the restored images are not well recovered. However, the work in [11] illustrates the potential for a multilevel approach to regularized signal and image deblurring.

Therefore, in the present work, we turn to the difficult task of recovering edge information in 1D signals using a multilevel approach. It is well-known in the signal processing literature that one can restore edge information by using, for example, a Tikhonov based approach where the regularization functional is a Total-Variation (TV) operator [52] (see Section 1.3.4). Another edge-preserving approach is the PP-TSVD approach [28]. These approaches, although they can work very well, can be quite computationally expensive. Therefore, we develop a multilevel approach that has the capability to recover some edge information at a smaller expense than other traditional methods. The authors in [40] had the same goal in mind. Their approach differs from ours in two respects: 1) they work with a full-multigrid approach

[21] rather than a traditional V-cycle with pre-smoothing, and 2) the edge preservation is obtained through use of a nonlinear edge-preserving prolongation operator, whereas our edge preservation is built in through the choice of solvers for the coarse-grid and residual corrections.

In this Chapter, we present a method that has a standard multigrid V-cycle scheme, but uses a different restriction operator than that in [11]. It also uses a residual-correction step which actually lives on a coarse grid but regularizes over the next finer grid. In particular, we choose to move between grids using a Haar wavelet operator. We show that the choice of the Haar wavelet operator has the advantage of preserving matrix structure between grids, which can be exploited to obtain faster solvers on each level, and allows for a straightforward analysis of errors and residuals as we move through the V-cycle. For other multigrid methods for Toeplitz matrices, see [1, 2].

3.2 Algorithm

In this section, we describe a multilevel method for the signal restoration problem. The overall idea follows a classical MG approach [7], such as was also used in [11]. We start by giving some details of how we move between grids.

3.2.1 Haar Decomposition

Recall that, in our problem, A is an $m \times m$ matrix with $m = 2^k$. In this chapter, we will use the special case of a DWT (see Section 2.2) to define our coarse-grid problem and move between grids. Consider the Haar wavelet transform (HWT) defined by the matrix

$$W^T = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 & 0 & 0 & \dots & \dots & 0 & 0 \\ 0 & 0 & 1 & 1 & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \dots & 1 & 1 \\ 1 & -1 & 0 & 0 & \dots & \dots & 0 & 0 \\ 0 & 0 & 1 & -1 & \dots & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \dots & 1 & -1 \end{pmatrix} = \begin{bmatrix} W_1^T \\ W_2^T \end{bmatrix}. \quad (3.1)$$

Our problem $Ax = b$ can be written in the wavelet domain as

$$\hat{A}\hat{x} = \hat{b}.$$

In other words, if W^T is the HWT¹, then $\hat{A} = W^T A W$, $\hat{x} = W^T x$, and $\hat{b} = W^T b$.

¹We use W^T rather than W for ease in reading the linear algebraic equation.

We partition the transformed problem into blocks of size 2^{k-1} to obtain the following problem

$$\begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_2 \end{bmatrix}, \quad (3.2)$$

with $\hat{A}_{ij} = W_i^T A W_j$, $\hat{x}_i = W_i^T x$, and $\hat{b}_i = W_i^T b$ for $i, j = 1, 2$.

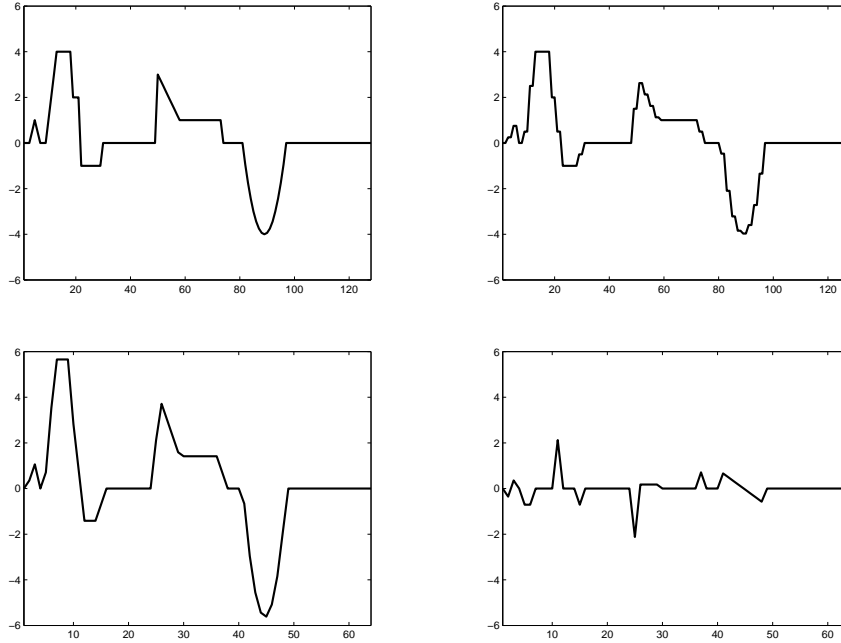


FIGURE 3.1: True solution x^{true} (top left), approximate solution $\tilde{x} = W[\hat{x}_1; 0]$ (top right), scaling coefficients \hat{x}_1 (bottom left), and wavelet coefficients \hat{x}_2 (bottom right).

We now discuss how to use this partitioning to our advantage. Notice that, given a vector v , $W_1^T v$ is formed by weighted averaging of neighboring points, while $W_2^T v$ is formed by weighted differencing. Not surprisingly, we observe from Figure 3.1 that the plot of the scaling coefficient vector \hat{x}_1 captures a similar amount of detail of the original image, while the plot of the wavelet coefficient vector \hat{x}_2 is spiky. This leads us naturally to say that if we could recover the “downsampled” signal \hat{x}_1 , we would have a rather good approximation to the fine-scale vector (in the sense that setting \hat{x}_2 to 0 and $\tilde{x} = W[\hat{x}_1; 0]$ (see Figure 3.1) would already give an okay (but blocky) representation of the signal). So our first goal is to try to recover \hat{x}_1 .

From the first block equation of (3.2), we have

$$\hat{A}_{11}\hat{x}_1 = \hat{b}_1 - \hat{A}_{12}\hat{x}_2. \quad (3.3)$$

We do not know \hat{x}_2 , of course. In the next section, we discuss why it is appropriate to find a *regularized solution* to

$$\hat{A}_{11}\hat{x}_1 = \hat{b}_1 \quad (3.4)$$

for our pre-smoothing and coarsest-grid correction problems, and we discuss what regularization methods are appropriate. Then, we will discuss how to recover \hat{x}_2 given our estimates of \hat{x}_1 .

3.2.2 Matrix Analysis

Let us compare the submatrices \hat{A}_{ij} . Recall that A is a discretized version of a blurring operator. Let

$$A = U\Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix},$$

be the singular value decomposition (SVD) of A . The matrices U_i and V_i with $i = 1, 2$ have size $2^k \times 2^{k-1}$, and the matrices Σ_i with $i = 1, 2$ have size $2^{k-1} \times 2^{k-1}$. The singular values of A , σ_i , are decreasing from 1 (assuming A has been normalized) to a very small number, with no significant gap. Recall from Section 1.2 that the column vectors in U and V increase in frequency from left to right, so we can think of the vectors in U_1 and V_1 as the lower-frequency vectors while U_2 and V_2 are the higher frequency. Then

$$\hat{A} = \begin{bmatrix} W_1^T U_1 & W_1^T U_2 \\ W_2^T U_1 & W_2^T U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T W_1 & V_1^T W_2 \\ V_2^T W_1 & V_2^T W_2 \end{bmatrix}. \quad (3.5)$$

If we look at the columns of $W_1^T U_1$, we find low-frequency vectors due to the fact that U_1 contains low-frequency column vectors and W_1^T smooths them even more. Since we know that the columns of U_1 are smoother than the columns of U_2 , the matrix $W_1^T U_1$ contains vectors with lower frequency than those of $W_1^T U_2$. As we mentioned before, W_2^T acts as a weighted difference. Then, the columns of $W_2^T U_1$ and $W_2^T U_2$ are higher-frequency vectors. Thus, $W_1^T U_1$ retains the character of the lowest-frequency vectors of U . A similar analysis can be done for the blocks on the rightmost matrix of (3.5), and we therefore conclude that $V_1 W_1^T$ represents best, among all four subblocks, the characteristics of the lowest-frequency vectors of V . In particular, we can compute

$$\hat{A}_{11} = W_1^T U_1 \Sigma_1 V_1^T W_1 + W_1^T U_2 \Sigma_2 V_2^T W_1, \quad (3.6)$$

which can be written as $\hat{A}_{11} = W_1^T U_1 \Sigma_1 V_1^T W_1 + E$. We have the following lemma:

Lemma 3.1. *Given the partitioning above,*

$$|\sigma_s(\hat{A}_{11}) - \sigma_s(W_1^T U_1 \Sigma_1 V_1^T W_1)| \leq \sigma_{2^{k-1}+1}, \quad s = 1, \dots, 2^{k-1},$$

where the notation $\sigma_s(B)$ means the s -largest singular value of the argument B .

Proof. Since W, U, V are matrices with orthonormal columns, it is easy to show that $\|W_i^T U_i\|_2 = 1$ and $\|V_i^T W_i\|_2 = 1$ for $i = 1, 2$. Therefore, $\|E\|_2 = \sigma_{2^{k-1}+1}$. By Corollary 8.6.2 in [17], we know that $|\sigma_s(\hat{A}_{11}) - \sigma_s(W_1^T U_1 \Sigma_1 V_1^T W_1)| \leq \|E\|_2$. Then, the result of the lemma follows. \square

Typically, in these problems, the singular value $\sigma_{2^{k-1}+1}$ of A is small, so we can say that the term $W_1^T U_1 \Sigma_1 V_1^T W_1$ is dominant in (3.6). This matrix is almost already in its SVD form as the matrices $W_1^T U_1$ and $V_1^T W_1$ are close to being orthogonal, and so if Σ_1 is somewhat ill-conditioned, it follows that \hat{A}_{11} inherits these characteristics as well. Specifically, let $\hat{A}_{11} = QDP^T$ be the SVD of \hat{A}_{11} . Let $T_1 = Q_{1:j}$ denote the first j columns of Q , and $T_2 = (W_1^T U_1)_{1:j}$ denote the first j columns of that matrix. Not surprisingly, since the first few (orthonormal) columns of U_1 were smooth and $W_1^T U_1$ is just the averaged version of U_1 , the sine of the canonical angles between the subspaces spanned by T_1 and T_2 has to be small (where small is relative to the size of j). (A similar statement holds for the left singular vectors.) This is illustrated in Figure 3.2 for the blurring operator in our numerical results section. By Corollary I.5.4 in [54], we can define

$$\cos \Theta(\text{Range}(T_1), \text{Range}(T_2)) = \text{diag}(\tilde{\sigma}_1, \dots, \tilde{\sigma}_j),$$

where $\tilde{\sigma}_l$ are the singular values of $T_1^T T_2$. Then,

$$\sin \Theta = \sqrt{1 - \tilde{\sigma}_l^2} \quad l = 1, \dots, j.$$

In Figure 3.2, we plot the maximum sine of the canonical angles as a function of j .

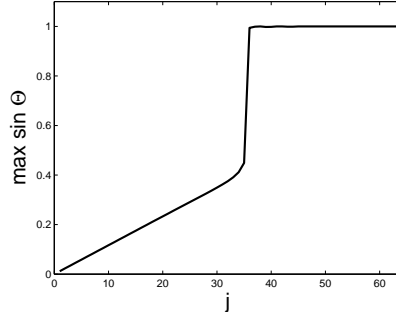


FIGURE 3.2: Maximum sine of the canonical angles between the subspaces spanned by T_1 and T_2 for $j = 1, \dots, 2^6$.

If we extend this analysis for the operator A^i at each level, then we expect \hat{A}_{11}^i to still be somewhat ill-conditioned on the finest grids (small i), but getting better conditioned the coarser we go (increasing i), and that \hat{A}_{11}^i always captures the lowest frequency characteristics of the original matrix A^0 .

3.2.3 Pre-smoothing and Coarsest-Grid Correction

It is well known that Krylov subspace algorithms such as LSQR [42] work as regularization methods if stopped early (i.e., before converging to the solution of the system) (see Section 1.3.3). This is because these methods, when applied to a discrete ill-posed problem, try to reduce the residual in the first few iterations over the subspace corresponding to the largest singular values, and these are precisely the smooth modes [34]. In other words, if r denotes the residual vector after j iterations of LSQR applied to a discrete ill-posed problem with white noise, a plot of the components of $U^T r$ shows the first few have

been decreased (compared to the initial residual), while the last few have remained relatively untouched (meaning noise has not been mixed in to the solution), as shown in Figure 3.3.

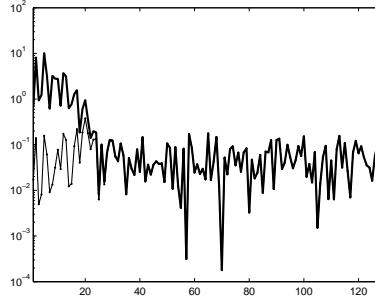


FIGURE 3.3: Spectral coefficients of b (thicker line) and spectral coefficients of $r = b - Ax$ (thinner line), where x is given by 3 iterations of LSQR.

The matrix analysis in the previous section has the following implication for our multigrid approach. Suppose we do 2 or 3 steps of LSQR on $Ax = b$, form $r = b - Ax$, and then form the residual equation in wavelet space:

$$\begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} \\ \hat{A}_{21} & \hat{A}_{22} \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} \hat{r}_1 \\ \hat{r}_2 \end{bmatrix}.$$

Consider the system

$$\hat{A}_{11}\hat{x}_1 = \hat{r}_1.$$

Looking at $Q^T\hat{r}_1$ where Q is the matrix of left singular vectors of \hat{A}_{11} , this is equal to

$$Q^T W_1^T r = Q^T W_1^T U U^T r = [Q^T W_1^T U_1, Q^T W_1^T U_2](U^T r).$$

Recall from the previous section that $Q_{1:j}^T (W_1^T U_1)_{1:j} \approx I$. So at least the first j components of this vector are the same in magnitude as they were before – that is, if they were made small on a previous grid they remain so.

We will consider the term $\hat{A}_{12}\hat{x}_2$ that appears in (3.3) as noise and use LSQR to find an (overregularized) approximation to the system

$$\hat{A}_{11}\hat{x}_1 = \hat{b}_1, \tag{3.7}$$

where we use b instead of r for consistency with the previous section. We can consider the term $\hat{A}_{12}\hat{x}_2$ as noise for the following reason. Recall that \hat{x}_2 is the “spiky” wavelet coefficient vector. Applying the operator \hat{A}_{12} smoothes \hat{x}_2 a little (by amplifying a slightly larger low-frequency component to it) but not much because \hat{A}_{12} is not quite a “smoother” like \hat{A}_{11} is. We also know that $\hat{b}_1 = W_1^T(b^{true} + e) = \hat{b}_1^{true} + \hat{e}_1$. Then, $\hat{A}_{12}\hat{x}_2$ is not white noise, but (spectrally) it is below the additive white noise, \hat{e}_1 , in our applications, so we can ignore it. Figure 3.4 shows the spectral coordinates of \hat{b}_1^{true} , \hat{e}_1 , and $\hat{A}_{12}\hat{x}_2$ for comparison. By discarding the term $\hat{A}_{12}\hat{x}_2$, a method based on solving (3.7) cannot compute the exact solution of the noise-free problem (1.2).

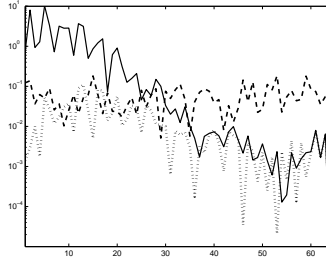


FIGURE 3.4: The spectral coordinates of \hat{b}_1^{true} (line), \hat{e}_1 (dashed line) and $\hat{A}_{12}\hat{x}_2$ (dotted).

Again, this analysis is extended to any level. We apply a few iterations of LSQR to obtain an (over-regularized) approximation, x_{pre}^i , to any intermediate system $A^i x^i = b^i$. At the coarsest level ($i = n$), though, while we may have adequately reduced the residual at the smoothest modes, we may not have obtained as much information over the higher-frequency modes of that level's operator (some of the highest frequencies relative to the finest level have, of course, disappeared by this level). We see that if the original signal has edges, the coarsest-grid problem may still have edges, since this is the system for the scaling coefficients. To recover that information here, we solve a regularized problem:

$$\min_{\hat{x}_1^n} \left\{ \|\hat{A}_{11}^n \hat{x}_1^n - \hat{b}_1^n\|_2^2 + \lambda^q \|L(\hat{x}_1^n + \hat{x}_0^n)\|_q^q \right\}, \quad (3.8)$$

where L is a discrete first-derivative operator, $1 < q < 2$, and \hat{x}_0^n is the initial guess, at the n -th level, which is the corresponding projection of the original initial guess x_0^0 and the successive pre-smoothing solutions x_{pre}^i with $i = 1, \dots, n-1$. This allows us to recover some edge information on this grid, but, since we are at a coarse level, this is a much more computationally tractable problem than if we had tried to employ it on the finest grid. On the other hand, if we are at a coarse-enough grid, we can replace this simply by a direct solve, as there are no longer any sufficiently small singular values to magnify any remaining noise in the right-hand side (and, in practice, the noise is barely perceptible in the right-hand side at this level).

In summary, we have:

- Pre-smoothing: take a few iterations of LSQR on $A^i x^i = b^i$, $i \neq n$,
- Coarsest-grid Correction: If \hat{A}_{11}^n is somewhat ill-conditioned, solve (3.8), otherwise, solve $\hat{A}_{11}^n \hat{x}_1^n = \hat{b}_1^n$ exactly by a direct or iterative method.

We discuss how to compute both these steps efficiently in Section 3.3.

3.2.4 Residual Correction

While the above steps can lead to good recovery of the scaling coefficients in the wavelet expansion, we need to recover the wavelet coefficients \hat{x}_2 as well in order to adequately recover edges but not produce a

solution that looks too artificially blocky. For this part, we define a residual-correction step. Assuming that we have an approximation \hat{x}_1^* of \hat{x}_1 from the coarse-scale Equation (3.7), then we can rewrite system (3.2) as follows,

$$\begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix} \hat{x}_2 = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_2 \end{bmatrix} - \begin{bmatrix} \hat{A}_{11} \\ \hat{A}_{21} \end{bmatrix} \hat{x}_1^*. \quad (3.9)$$

We denote the right-hand side of Equation (3.9) by \hat{r}_{new} . To solve this problem, we again use the following Tikhonov formulation to keep certain smoothness in the solution,

$$\min_{\hat{x}_2} \left\{ \left\| \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix} \hat{x}_2 - \hat{r}_{new} \right\|_2^2 + \lambda^q \|L(x_{new} + W_2 \hat{x}_2)\|_q^q \right\}, \quad (3.10)$$

where L is a first-derivative operator, $1 < q < 2$ as in Equation (3.8), and $x_{new} = x_0 + x_{pre} + W_1 \hat{x}_1^*$ where x_0 is the initial guess before pre-smoothing on the current level, x_{pre} is the correction given by the pre-smoothing step, and $W_1 \hat{x}_1^*$ is the prolongation of the coarse-scale solution \hat{x}_1^* . Notice that this is different from the standard multigrid scheme where x_0 is set to zero at each iteration.

The superscript i is omitted to simplify notation; however, this step is applied to each level. From a computational perspective, it is very important to note that the regularization is applied to the reconstructed signal. In Section 3.3.2, we explain how to find an approximate solution of (3.10) efficiently.

Our multilevel algorithm is summarized in Table 3.1, where iter in line 5 is the number of iterations of LSQR. As in [11], we do not pre-smooth on the finest level. The reason is mainly due to computational cost since pre-smoothing at the finest level will in fact improve the solution given by this multilevel approach.

3.3 Computational issues

One nice feature of the Haar decomposition is that it preserves Toeplitz matrix structure. This implies that we can have fast matrix-vector products of the intermediate-level interior linear least squares problems that must be solved iteratively (see next two subsections). In [36], the discrete wavelet transform (DWT) of Toeplitz matrices is discussed. We restrict our study to the Haar wavelet transform (HWT), however, and apply a different technique than the one in [36] to prove a stronger result.

Theorem 3.2. *Let A be an $m \times m$ matrix with Toeplitz structure, and $m = 2^k$. Then, the $2^{k-1} \times 2^{k-1}$ submatrices defined in (3.2), \hat{A}_{11} , \hat{A}_{12} , \hat{A}_{21} , and \hat{A}_{22} , also have a Toeplitz structure. Moreover, we can define the Toeplitz-vector of each submatrix \hat{A}_{ij} as a function of the Toeplitz-vector of A .*

TABLE 3.1: Multilevel V-Cycle: MGM

Input: A^i, b^i, x_0^i
Output: x^i
1. If $i = n$
2. $x^i = \text{direct solve or Newton}(A^i, b^i, x_0^i)$ (see Section 3.3.1)
3. Else
4. If $i \neq 0$
5. $x_{pre}^i = LSQR(A^i, b^i, \text{iter})$
6. $x^i = x_0^i + x_{pre}^i$
7. Else
8. $x^i = x_0^i$
9. End If
10. $r^i = b^i - A^i x^i$
11. $b^{i+1} = W_1^T r^i$
12. $x_0^{i+1} = W_1^T x^i$
13. $A^{i+1} = W_1^T A^i W_1$ (see Section 3.3)
14. $\hat{x}_1^{i+1} = MGM(A^{i+1}, b^{i+1}, x_0^{i+1})$
15. $x_{new}^i = x^i + W_1 \hat{x}_1^{i+1}$
16. $r_{new}^i = b^i - A^i x_{new}^i$
17. $\hat{x}_2^{i+1} = \text{Newton}(W A^i W_2, W r_{new}^i, x_{new}^i)$ (see Section 3.3.2)
18. $x^i = x_{new}^i + W_2 \hat{x}_2^{i+1}$
19. End If

Proof. We shall use the representation of a Toeplitz matrix introduced in [33]. Consider the following Toeplitz matrix,

$$A = \begin{bmatrix} t_0 & t_{-1} & t_{-2} & \dots & t_{-(m-1)} \\ t_1 & t_0 & t_{-1} & \dots & t_{-(m-2)} \\ t_2 & t_1 & t_0 & \dots & t_{-(m-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{m-1} & t_{m-2} & t_{m-3} & \dots & t_0 \end{bmatrix}.$$

The vector $t = (t_{-(m-1)}, \dots, t_{-1}, t_0, t_1, \dots, t_{m-1})$ is called the *Toeplitz-vector* that generates A . If Z_m is the $(2m-1) \times (2m-1)$ “downshift” matrix (ie. a Toeplitz matrix whose Toeplitz-vector t is a vector of all zeros except $t_{-1} = 1$) and P_m is the $m \times (2m-1)$ matrix $[I_m, 0]$, then

$$A = [P_m Z_m^{m-1} t, \dots, P_m Z_m^1 t, P_m Z_m^0 t]. \quad (3.11)$$

If $\text{vec}(E)$ unstacks matrix E by columns to create a column vector, then we have that $\text{vec}(W^T A W) = (W^T \otimes W^T) \text{vec}(A)$ (see Lemma 8.5.3 in [46]). Then,

$$\begin{aligned} \text{vec}(\hat{A}) &= (W^T \otimes W^T) \text{vec}(A) = (W^T \otimes I) \begin{bmatrix} W^T P_m Z_m^{m-1} t \\ \vdots \\ W^T P_m Z_m^0 t \end{bmatrix} = (W^T \otimes I) \begin{bmatrix} \bar{c}_1 \\ \vdots \\ \bar{c}_m \end{bmatrix} \\ &= \frac{1}{\sqrt{2}} \begin{bmatrix} \bar{c}_1 + \bar{c}_2 \\ \bar{c}_3 + \bar{c}_4 \\ \vdots \\ \bar{c}_{m-1} + \bar{c}_m \\ \bar{c}_1 - \bar{c}_2 \\ \bar{c}_3 - \bar{c}_4 \\ \vdots \\ \bar{c}_{m-1} - \bar{c}_m \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} W^T P_m Z_m^{m-2} (Z_m^1 + Z_m^0) t \\ W^T P_m Z_m^{m-4} (Z_m^1 + Z_m^0) t \\ \vdots \\ W^T P_m Z_m^0 (Z_m^1 + Z_m^0) t \\ W^T P_m Z_m^{m-2} (Z_m^1 - Z_m^0) t \\ W^T P_m Z_m^{m-4} (Z_m^1 - Z_m^0) t \\ \vdots \\ W^T P_m Z_m^0 (Z_m^1 - Z_m^0) t \end{bmatrix}. \end{aligned}$$

Define $\bar{t} = (Z_m + I)t$ and $\tilde{t} = (Z_m - I)t$. It follows that,

$$\hat{A} = \frac{1}{\sqrt{2}} W^T P_m [Z_m^{m-2} \bar{t}, Z_m^{m-4} \bar{t}, \dots, \bar{t}, Z_m^{m-2} \tilde{t}, Z_m^{m-4} \tilde{t}, \dots, \tilde{t}].$$

Then, we have

$$\hat{A} = \frac{1}{2} \left[\begin{array}{c|c} P_{m/2} Z_{m/2}^{\frac{m}{2}-1} \bar{t}_{11}, \dots, P_{m/2} Z_{m/2}^0 \bar{t}_{11} & P_{m/2} Z_{m/2}^{\frac{m}{2}-1} \bar{t}_{12}, \dots, P_{m/2} Z_{m/2}^0 \bar{t}_{12} \\ \hline P_{m/2} Z_{m/2}^{\frac{m}{2}-1} \tilde{t}_{21}, \dots, P_{m/2} Z_{m/2}^0 \tilde{t}_{21} & P_{m/2} Z_{m/2}^{\frac{m}{2}-1} \tilde{t}_{22}, \dots, P_{m/2} Z_{m/2}^0 \tilde{t}_{22} \end{array} \right], \quad (3.12)$$

where the vectors \bar{t}_{11} , \bar{t}_{12} , \tilde{t}_{21} , and \tilde{t}_{22} are defined (using MATLAB notation) as follows

$$\begin{aligned} t_{11} &= (I + Z_m) \bar{t}, & t_{12} &= (I + Z_m) \tilde{t}, & t_{21} &= (I - Z_m) \bar{t}, & t_{22} &= (I - Z_m) \tilde{t}, \\ \bar{t}_{11} &= t_{11(1:2:2m-3)}, & \bar{t}_{12} &= t_{12(1:2:2m-3)}, & \tilde{t}_{21} &= t_{21(1:2:2m-3)}, & \tilde{t}_{22} &= t_{22(1:2:2m-3)}. \end{aligned}$$

From Equation (3.12) and from the characterization of Toeplitz matrices (3.11), the result of the theorem follows. Moreover, we have shown that the vectors $\frac{1}{2} \bar{t}_{11}$, $\frac{1}{2} \bar{t}_{12}$, $\frac{1}{2} \tilde{t}_{21}$, and $\frac{1}{2} \tilde{t}_{22}$ are their respective Toeplitz-vectors. \square

There are two important computational issues deduced from Theorem 3.2:

- We can compute the entries of \hat{A}_{ij} with $i, j = 1, 2$ without explicitly computing the matrices,
- Since \hat{A}_{ij} are Toeplitz, they admit fast matrix-vector products. This is a computational advantage for LSQR on the downward part of the V-cycle and also for solving the system or regularized problem on the coarsest level (as noted below).

Furthermore, we can show that the bandwidth of the Toeplitz matrices at each grid level is reduced. Then, the coarsest-grid operator can be reduced as far as to a tridiagonal matrix depending on the initial bandwidth and how many levels are used, again reducing the overall computational burden further down the V-cycle.

Theorem 3.3. *If A is an $m \times m$ banded Toeplitz matrix for $m = 2^k$ with upper bandwidth k_u and lower bandwidth k_l , then \hat{A}_{ij} is an $m/2 \times m/2$ banded Toeplitz matrix with bandwidths $\lceil k_u/2 \rceil$ and $\lceil k_l/2 \rceil$.*

Proof. As in the preceding proof, we deal directly with the Toeplitz-vectors. From the proof above, we know that Toeplitz-vectors that generate \hat{A}_{ij} can be written as $\bar{t}_{ij} = t_{ij}(1 : 2 : 2m - 3)$ where t_{ij} is a $2m - 1$ length vector $t_{ij} = \alpha It + \beta Z_m t + \gamma Z_m^2 t$ with α and γ either plus or minus 1 and β either 0, 2.

Thus each t_{ij} is easily shown to have $m - k_u - 3$ leading zeros, followed by $k_l + k_u + 3$ nonzeros, followed by $m - k_l - 1$ trailing zeros. Since \bar{t}_{ij} is obtained by subsampling t_{ij} in every other entry starting with the first, there must be $\lceil \frac{m - k_u - 3}{2} \rceil$ leading zeros in \bar{t}_{ij} . Clearly, the upper bandwidth of the $m/2 \times m/2$ matrices \hat{A}_{ij} is obtained by subtracting the number of leading zeros plus 1 (to account for the main diagonal) from $m/2$. Given the number of nonzeros in the “middle” of the Toeplitz-vector \bar{t}_{ij} and the upper bandwidth, we can compute the lower bandwidth.

Case 1: $k_u = 2s + 1$ for positive integer s . The upper bandwidth of \hat{A}_{ij} is easily shown to be $s + 1$ (equivalently, $\lceil \frac{k_u}{2} \rceil$). Since k_u is odd, given the subsampling pattern, there must be $\lceil \frac{k_l + k_u + 3}{2} \rceil = s + 2 + \lceil \frac{k_l}{2} \rceil$ nonzeros after the leading zeros in \bar{t}_{ij} . Hence, the lower bandwidth of \hat{A}_{ij} is $s + 2 + \lceil \frac{k_l}{2} \rceil - 1 - (s + 1) = \lceil \frac{k_l}{2} \rceil$.

Case 2: $k_u = 2s$, which implies the upper bandwidth of each \hat{A}_{ij} is $\frac{m}{2} - (\frac{m}{2} - s - 1) - 1 = s = \frac{k_u}{2}$. Since k_u is even there are $\lfloor \frac{k_u + k_l + 3}{2} \rfloor = s + \lfloor \frac{k_l + 3}{2} \rfloor$ nonzeros after the leading zeros in \bar{t}_{ij} . Thus the lower bandwidth is $s + \lfloor \frac{k_l + 3}{2} \rfloor - 1 - s$ which is easily shown to be $\lceil \frac{k_l}{2} \rceil$ regardless of whether k_l is even or odd. \square

3.3.1 Coarse-Scale Solve

Recall that we may need to solve Equation (3.8) on the coarsest grid if not enough levels are used. If $q = 2$, we could use a hybrid approach [32] to solve the Tikhonov problem and simultaneously select the regularization parameter in an efficient manner. However, we typically take $1 < q < 2$, with $q \ll 2$ to best preserve the edges at the coarsest grid. To solve the minimization problem (3.8), we use Newton’s method with line search. The recursion of this method is defined by $x_{k+1} = x_k + \alpha d_k$, where α is the length of the step² and d_k is its direction. Recall that if we minimize a function $s(x)$, then the direction d_k is the solution to the linear system

$$H(x)d = -\nabla s(x), \quad (3.13)$$

where $\nabla s(x)$ is the gradient of s with respect to x , and $H(x)$ is the *Hessian* matrix, containing second derivatives: $H_{ij} = \partial^2 s(x) / \partial x_i \partial x_j$. In our particular case, we have that

$$s(\hat{x}_1) = \|\hat{A}_{11}\hat{x}_1 - \hat{b}_1\|_2^2 + \lambda^q \|L(\hat{x}_1 + \hat{x}_0)\|_q^q,$$

Then, the corresponding gradient and Hessian are as follows,

$$\nabla s(\hat{x}_1) = 2\hat{A}_{11}^T \bar{r}_1 + \lambda^q q L^T \text{sign}(\bar{r}_2) \cdot |\bar{r}_2|^{q-1},$$

²The value of α is chosen using a standard line search.

and

$$H(\hat{x}_1) = 2\hat{A}_{11}^T \hat{A}_{11} + \lambda^q q(q-1)L^T \text{diag}(|\bar{r}_2|^{q-2})L,$$

where, $\bar{r}_1 = \hat{A}_{11}\hat{x}_1 - \hat{b}_1$, $\bar{r}_2 = L(\hat{x}_1 + \hat{x}_0)$ and for a vector v , $\text{sign}(v)$ and $|v|$ are the vectors containing the sign and absolute value of each entry of v , and $\text{diag}(v)$ is the diagonal matrix with entries given by v ; also for two vectors v and w , $v.*w$ is their coordinate-wise product. Then, the system of equation (3.13) can be written as

$$(\hat{A}_{11}^T \hat{A}_{11} + L^T D L)d = -(\hat{A}_{11}^T \bar{r}_1 + \frac{1}{q-1}L^T D \bar{r}_2), \quad (3.14)$$

where $D = \frac{\lambda^q}{2}q(q-1)\text{diag}(|\bar{r}_2|^{q-2})$. To solve Equation (3.14), we can apply an iterative method such as CG or MINRES [17] or we can also solve its equivalent linear least squares problem,

$$\min_d \left\| \begin{bmatrix} \hat{A}_{11} \\ D^{1/2}L \end{bmatrix} d - \begin{bmatrix} -\bar{r}_1 \\ -\frac{1}{q-1}D^{1/2}\bar{r}_2 \end{bmatrix} \right\|_2^2, \quad (3.15)$$

by applying an iterative method as LSQR or CGLS that needs only the matrix vector product $\hat{A}_{11}v$ and $D^{1/2}Lv$.

Choosing a good regularization parameter specifically for (3.8) is beyond the scope of this work. In our numerical section, we present results for which λ was chosen to be optimal over a small, discrete set of choices. Choosing the parameter at this level is the subject of current research.

3.3.2 Residual Correction Solve

To solve the minimization problem (3.10), we also use Newton's method with line search. In this case, we have that

$$s(\hat{x}_2) = \|\tilde{r}_1\|_2^2 + \|\tilde{r}_2\|_q^q,$$

where

$$\tilde{r}_1 = \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix} \hat{x}_2 - \hat{r}_{new} \quad \text{and} \quad \tilde{r}_2 = \lambda(Lx_{new} + LW_2\hat{x}_2).$$

Then, the corresponding gradient and Hessian are as follows

$$\nabla s(\hat{x}_2) = 2 \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix}^T \tilde{r}_1 + q(\lambda LW_2)^T \text{sign}(\tilde{r}_2) .* |\tilde{r}_2|^{q-1},$$

and

$$H(\hat{x}_2) = 2 \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix}^T \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix} + q(q-1)(\lambda LW_2)^T \text{diag}(|\tilde{r}_2|^{q-2})(\lambda LW_2).$$

Then, the system of equation (3.13) can be written as

$$A_L^T D A_L d = A_L^T D w, \quad (3.16)$$

where

$$A_L = \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \\ \lambda L W_2 \end{bmatrix}, \quad D = \begin{bmatrix} 2I & 0 \\ 0 & D_q \end{bmatrix} \quad \text{and} \quad w = \begin{bmatrix} -\tilde{r}_1 \\ -\frac{1}{q-1}\tilde{r}_2 \end{bmatrix},$$

with $D_q = q(q-1)\text{diag}(|\tilde{r}_2|^{q-2})$. To solve Equation (3.16), we can apply an iterative method such as CG or MINRES [17]. We do not need to construct the matrix $A_L^T D A_L$ explicitly. We only need the matrix-vector product. Since D is diagonal and A_L has Toeplitz and sparse blocks, this is efficiently solved using iterative solvers that capitalize on this structure for matrix-vector products. We can also solve its equivalent linear least squares problem,

$$\min_d \|D^{1/2} A_L d - D^{1/2} w\|_2^2, \quad (3.17)$$

by applying an iterative method such as LSQR or CGLS. Also in this case, we do not need to construct the matrix $D^{1/2} A_L$ explicitly.

Recall that \hat{x}_2 lives on a coarser grid than where we are enforcing the regularization (at the next finest grid level). As we have seen in Figure 3.1, the wavelet coefficients are ‘spiky’. It is still worthwhile to use q as close to 1 as possible to recover wavelet coefficients accurately enough to represent the edges appropriately on the next finest grid level. As mentioned in the previous subsection, the selection of an optimal regularization parameter is beyond the scope of the present work.

3.4 Numerical Results

We conclude with illustrations of the performance of our algorithm in signal restoration problems. All computations were performed in MATLAB. In the following examples, we aim to recover the original signal/image from a blurred and noisy signal/image.

The blurring operator is defined by an $m \times m$ symmetric banded Toeplitz matrix A , where its first row is defined by the vector

$$z = \frac{1}{2\pi\sigma^2} [\exp(-([0 : \text{band} - 1].^2)/(2\sigma^2)); \text{zeros}(1, m - \text{band})]. \quad (3.18)$$

The parameter band specifies the half-bandwidth of the matrix A and the parameter σ corresponds to the variance of the Gaussian point spread function

$$h(x) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right).$$

We use $m = 128$, and the parameters $\sigma = 3$ and $\text{band} = 7$. The matrix A is normalized such that $\|A\|_2 = 1$. The condition number of A is $\kappa(A) = 4.8 \times 10^5$. The exact solution, represented by x^{true} , is the vector of length 128 shown in Figure 3.1. The noise-free blurred signal, represented by b^{true} , is computed as $b^{true} = Ax^{true}$. The elements of the noise vector e are normally distributed with zero mean,

and the standard deviation is chosen such that $\frac{\|e\|_2}{\|b^{true}\|_2} = 0.05$. In this case, we say that the level of noise is 5%. The noisy right-hand side of our system is defined by $b = b^{true} + e$; (see Figure 3.5).

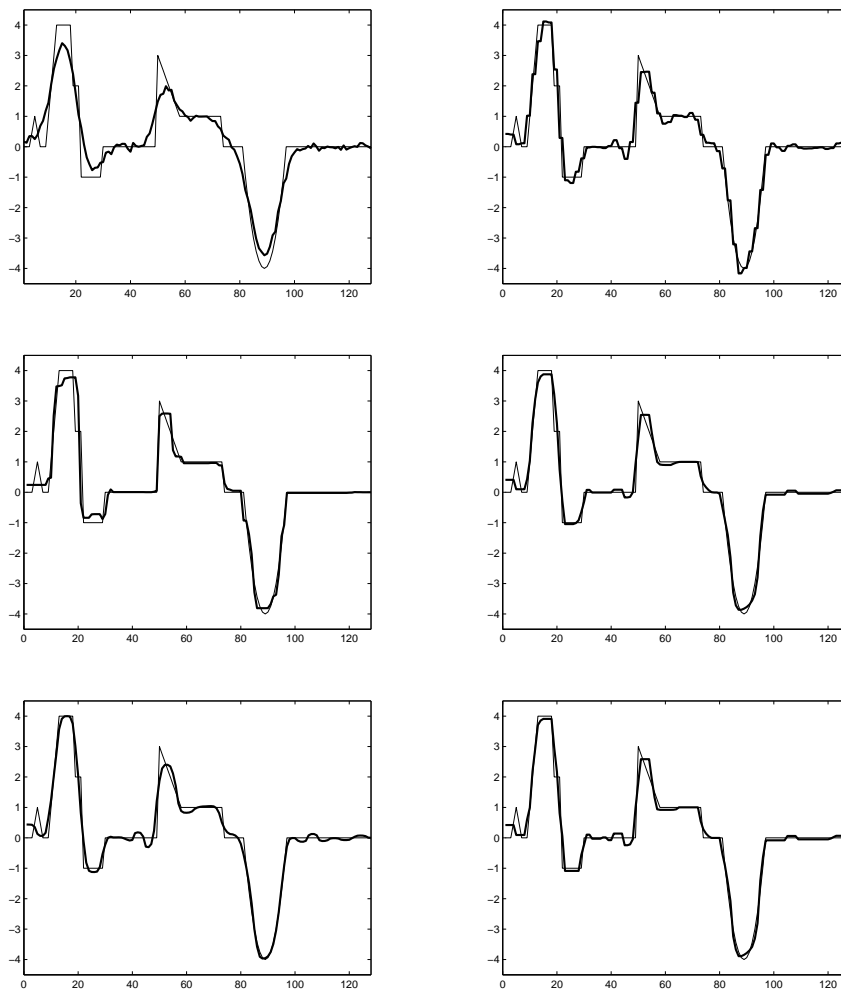


FIGURE 3.5: Blurred Noisy right-hand side, b , (top left), 3-MGM solution without residual-correction step (top right), 3-MGM solution without pre-smoothing step (no LSQR iteration) (middle left), 3-MGM solution with $q = 1.1$ (middle right), 3-MGM solution with $q = 1.5$ (bottom left) and 3-MGM solution with $q = 1.01$. In all figures, the exact solution x^{true} appears as a thin line.

For the solutions shown in Figure 3.5, we define the initial guess $x_0^0 = 0$, and we applied 9 iterations of LSQR on the pre-smoothing step. The operator at the coarsest level has size 32×32 (which means we apply a 3-level MGM), and a Newton method is applied to solve the coarsest-grid correction. For the regularization operator in Equations (3.8) and (3.10), we use the matrix

$$L = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & -1 & 1 \end{pmatrix},$$

which is a scaled discrete approximation to the first derivative operator on a regular grid, with no assumptions on boundary conditions. We define 10 logarithmically spaced values of λ from 10^{-3} to 1. We

TABLE 3.2: Comparison of different levels and parts of the multilevel method

level	LSQR iter	q	Newton iter	λ_{opt}	$\frac{\ x^{true} - x^{MG M}\ _1}{\ x^{true}\ _1}$
1	-	1.1	34	λ^7	0.1821
2	9	1.1	31,29	$\lambda^{4,10}$	0.1963
3	9	1.1	19,27,28	$\lambda^{4,5,10}$	0.1736
3	9	1.1	23,0,0	$\lambda^{4,-,-}$	0.2397
3	0	1.1	23,29,34	$\lambda^{8,6,5}$	0.1902
3	9	1.01	47,90,45	$\lambda^{3,4,10}$	0.1796
3	9	1.5	5,9,7	$\lambda^{5,7,9}$	0.1819

loop for all values of λ at each level and choose the combination of λ values that makes $\frac{\|x^{true} - x^{MG M}(\lambda^{ijk})\|_1}{\|x^{true}\|_1}$ minimum. We test the method with $q = 1.1$ in both minimization problems (3.8) and (3.10). We observe that using both the pre-smoothing and residual-correction steps are essential to recover the signal. Without pre-smoothing, we miss some edge information, and we obtain a blocky solution if we do not use the residual-correction step in our algorithm.

We also test the method for different number of levels, that is, we compute the solution of a 1-level, 2-level, and 3-level MGMs. In Table 3.4, we report the number of iterations for Newton's method at each level, the optimal values of λ for each level, and the relative errors computed by $\frac{\|x^{true} - x^{MG M}\|_1}{\|x^{true}\|_1}$. The leftmost Newton iteration number is the number of iterations on the coarsest grid, and they read from left to right with respect to coarseness of the grid. Recall that for multiple levels, the last Newton iteration is on a residual correction equation where the solution is on the next-to-finest grid and only the regularization is applied at the finest level. The notation λ^{ijk} means that for the coarsest level we need $\lambda(i)$, for level 1 we need $\lambda(j)$, and for the finest level we need $\lambda(k)$.

We also compare solutions for different values of q since the algorithm is sensitive to changes in it. When q is very close to 1, then the number of iterations needed for Newton's method to converge increases. On the other hand, when $q = 1.5$, only a few iterations are needed. However, we can see in Figure 3.5 that, in the case when $q = 1.5$, the solution is smooth, losing some edge information. One could opt to increase q with decreasing coarseness in an attempt to achieve more balance, but we did not investigate that possibility here.

For different levels of noise (1%, 5%, and 10%), we compute an LSQR solution, $x^{LSQR}(k)$, and a TV solution, $x^{TV}(\lambda)$, at the finest level for comparison. To compute x^{TV} we applied a Primal-Dual Newton Method (see Section 8.2.5 in [56]). We define 10 logarithmically spaced values of λ from 10^{-4} to 1. The optimal parameters k and λ are chosen such that $\frac{\|x^{true} - x^{LSQR}(k)\|_1}{\|x^{true}\|_1}$ and $\frac{\|x^{true} - x^{TV}(\lambda)\|_1}{\|x^{true}\|_1}$ are minimums. For a quantitative comparison see Table 4.4. For qualitative comparison in the case when the noise level is 10%, see Figure 3.6.

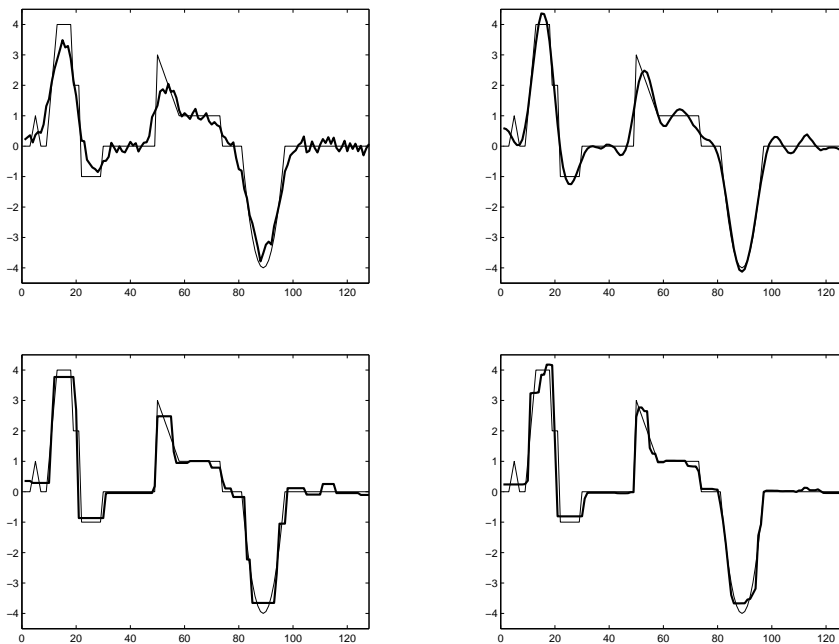


FIGURE 3.6: Blurred Noisy right-hand side, b , (top left), LSQR solution (top right), TV solution (bottom left), MGM solution (bottom right). In all figures, the exact solution x^{true} appears as a thin line.

TABLE 3.3: Comparison of different methods applied to different levels of noise

Noise Level	LSQR (k)	TV (λ_i)	MGM (k, λ^{ijk})
1%	0.1742 (24)	0.1637 (λ_2)	0.1340 (24, $\lambda^{5,5,10}$)
5%	0.2165 (8)	0.2065 (λ_4)	0.1736 (9, $\lambda^{4,6,10}$)
10%	0.2191 (6)	0.2187 (λ_7)	0.2049 (2, $\lambda^{7,7,8}$)

3.5 Conclusions

In this chapter, a new edge-preserving multilevel algorithm for signal restoration problems is presented. The transformation of the problem to the wavelet domain using Haar wavelets leads to a multilevel algorithm that preserves matrix structures. In particular, we have shown that Toeplitz structure is preserved between grids, which leads to intermediate linear system solves inside damped Newton iterations that are structured. This means that iterative methods can be efficiently applied to the linear systems. Numerical experiments illustrate that this is a promising technique for signal restoration problem with edges. Future work will include efficient parameter selection techniques, choosing the number of smoothing steps at intermediate grids, and the number of V-cycles.

Chapter 4

Multilevel Methods for Image Restoration Problems

4.1 Introduction

In this chapter, we present new multilevel approaches for image restoration problems that extend the multilevel method developed in the previous chapter for 1D signal restoration problems. Multigrid methods have already been considered for image restoration problems [11, 40, 57, 58]. In the introduction of Chapter 3, we have already discussed about the differences between our approach and those presented in [11, 40]. In [57], the authors present a wavelet-based multigrid V-cycle scheme very similar to the one we present here. However, it is applied to a 2-norm Tikhonov regularization problem, restoring smooth images. Therefore, in the present chapter, we turn to the difficult task of recovering edge information in images using a wavelet-based multilevel approach. Other similar wavelet-based *algebraic* multigrid methods [7] have been developed [15, 44]. They also use the DWT to move between grids, but standard smoothers such as Gauss-Seidel were applied since they were developed to solve shifted linear systems or to be used as preconditioners for GMRES, instead of solving image restoration problems.

The structure of this chapter is as follows. First, we will describe two-level methods for a separable case where it is easier to gain some intuition. Then, we will discuss how to develop their corresponding multilevel methods. And finally we will cover the non-separable case. Numerical results at the end of this chapter will illustrate that this is a promising technique for image restoration problems.

4.2 Algorithms

4.2.1 Two-Level Methods

Recall that the image restoration problem we defined in Section 1.1.2 consists of finding x such that

$$Ax = b = b^{true} + e, \quad (4.1)$$

where the $m^2 \times m^2$ matrix A is a known blurring operator and b denotes the blurred, noisy image. Figure 4.1 shows an example of an image that we want to recover, x^{true} , from the given blurred, noisy image, b . Recall also that the 2D HWT, $W^T \in \mathbb{R}^{m^2 \times m^2}$, can be defined in terms of the 1D HWT blocks, W_1^T ,

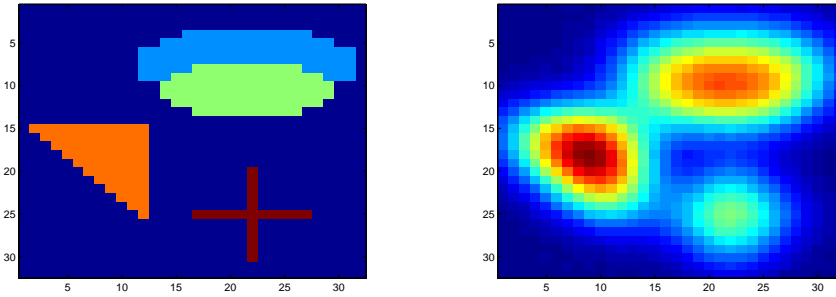


FIGURE 4.1: Image we want to recover, x^{true} (left), and blurred, noisy image, b (right).

$W_2^T \in \mathbb{R}^{\frac{m}{2} \times m}$, as

$$W^T = \begin{bmatrix} W_1^T \otimes W_1^T \\ W_2^T \otimes W_1^T \\ W_1^T \otimes W_2^T \\ W_2^T \otimes W_2^T \end{bmatrix}.$$

Then, the system (4.1) can be written in the wavelet domain as

$$\hat{A}\hat{x} = \hat{b}, \quad (4.2)$$

where $\hat{A} = W^T A W$, $\hat{x} = W^T x$, and $\hat{b} = W^T b$.

First, we focus on the case where A is separable, that is,

$$A = \tilde{A} \otimes \tilde{A} \in \mathbb{R}^{m^2 \times m^2},$$

with $m = 2^k$. By properties of Kronecker products [46], we have that

$$(W_i^T \otimes W_j^T)(\tilde{A} \otimes \tilde{A})(W_{i'} \otimes W_{j'}) = (W_i^T \tilde{A} W_{i'}) \otimes (W_j^T \tilde{A} W_{j'}),$$

for $i, i', j, j' = 1, 2$. In this case, the system (4.2) can be written as

$$\begin{bmatrix} \hat{A}_1 \otimes \hat{A}_1 & \hat{A}_2 \otimes \hat{A}_1 & \hat{A}_1 \otimes \hat{A}_2 & \hat{A}_2 \otimes \hat{A}_2 \\ \hat{A}_3 \otimes \hat{A}_1 & \hat{A}_4 \otimes \hat{A}_1 & \hat{A}_3 \otimes \hat{A}_2 & \hat{A}_4 \otimes \hat{A}_2 \\ \hat{A}_1 \otimes \hat{A}_3 & \hat{A}_2 \otimes \hat{A}_3 & \hat{A}_1 \otimes \hat{A}_4 & \hat{A}_2 \otimes \hat{A}_4 \\ \hat{A}_3 \otimes \hat{A}_3 & \hat{A}_4 \otimes \hat{A}_3 & \hat{A}_3 \otimes \hat{A}_4 & \hat{A}_4 \otimes \hat{A}_4 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_3 \\ \hat{x}_2 \\ \hat{x}_4 \end{bmatrix} = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_3 \\ \hat{b}_2 \\ \hat{b}_4 \end{bmatrix}, \quad (4.3)$$

where \hat{x}_i , \hat{b}_i , and \hat{A}_i for $i = 1, \dots, 4$, are such that

$$x = (W_1 \otimes W_1)\hat{x}_1 + (W_1 \otimes W_2)\hat{x}_2 + (W_2 \otimes W_1)\hat{x}_3 + (W_2 \otimes W_2)\hat{x}_4,$$

$$b = (W_1 \otimes W_1)\hat{b}_1 + (W_1 \otimes W_2)\hat{b}_2 + (W_2 \otimes W_1)\hat{b}_3 + (W_2 \otimes W_2)\hat{b}_4,$$

and

$$\begin{bmatrix} \hat{A}_1 & \hat{A}_2 \\ \hat{A}_3 & \hat{A}_4 \end{bmatrix} = \begin{bmatrix} W_1^T \tilde{A} W_1 & W_1^T \tilde{A} W_2 \\ W_2^T \tilde{A} W_1 & W_2^T \tilde{A} W_2 \end{bmatrix}.$$

The system (4.3) can be split in different ways. Depending on how we split it, we will get different two-level methods. Notice the order of the coordinates of \hat{x} and \hat{b} in (4.3).

Two-Level Method TL1

The first two-level method we will describe is the most natural extension of the multilevel method presented in Chapter 3 for signal restoration problems. Following the same analysis used to develop that method, we can make these observations:

- the vector \hat{x}_1 is a coarse approximation of the original image x ,
- the matrix $\hat{A}_1 \otimes \hat{A}_1$ reflects the ill-conditioning in A , and
- the vector \hat{e}_1 is still Gaussian white noise.

As we can see in Figure 4.2, the 2D HWT takes our image x in Figure 4.1 and maps it to a new image consisting of four blocks, corresponding to the \hat{x}_i s. The upper left block \hat{x}_1 is a lower resolution version of x , the upper right block \hat{x}_2 emphasizes vertical edges since they correspond to jumps in the horizontal direction, and the lower blocks \hat{x}_3 and \hat{x}_4 emphasize horizontal and diagonal edges, respectively. This suggests that if we could recover the “downsampled” image \hat{x}_1 , we would have a rather good approximation of the fine-scale vector x (in the sense that setting \hat{x}_2 , \hat{x}_3 , and \hat{x}_4 to 0 and $\tilde{x} = W[\hat{x}_1; 0]$ (see Figure 4.3) would already give a decent representation of the image).

We have shown for the 1D case that $\hat{A}_1 = W_1^T \tilde{A} W_1$ is still ill-conditioned and that it captures the lowest frequency characteristics of \tilde{A} . Thus, we expect the matrix $\hat{A}_1 \otimes \hat{A}_1$ to be somewhat ill-conditioned since its condition number $\kappa(\hat{A}_1 \otimes \hat{A}_1)$ (that is, $\|\hat{A}_1 \otimes \hat{A}_1\| \|(\hat{A}_1 \otimes \hat{A}_1)^{-1}\|$) is $\kappa(\hat{A}_1)^2$, and we also expect to capture the lowest frequency characteristics of A .

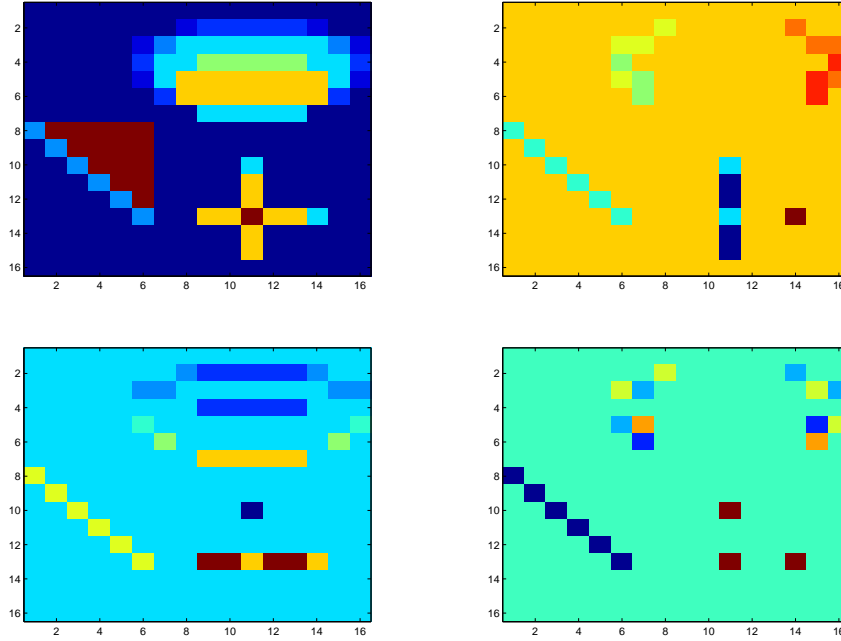


FIGURE 4.2: Blocks of the Haar wavelet transformation of the original image of Figure 4.1: \hat{x}_1 (top left), \hat{x}_2 (top right), \hat{x}_3 (bottom left), and \hat{x}_4 (bottom right)

We can also see that, when the entries of the vector e are independent and identically distributed with zero mean and standard deviation s , the vector \hat{e} also has the same distribution, that is, it has a multivariate normal distribution with zero mean matrix and covariance matrix equal to sI . It is important to note that this property is due to the fact that the 2D HWT, W^T , is orthogonal.

With these observations in mind we split the system (4.2) as follows

$$\begin{bmatrix} \tilde{A}_1 & \tilde{A}_2 \\ \tilde{A}_3 & \tilde{A}_4 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix} = \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \end{bmatrix}, \quad \tilde{x}_1 = \hat{x}_1 \in \mathbb{R}^{\frac{1}{4}m^2}, \quad \tilde{x}_2 = \begin{bmatrix} \hat{x}_3 \\ \hat{x}_2 \\ \hat{x}_4 \end{bmatrix} \in \mathbb{R}^{\frac{3}{4}m^2}, \quad (4.4)$$

with similar definitions for \tilde{b}_1 and \tilde{b}_2 , and \tilde{A}_i for $i = 1, \dots, 4$ having the corresponding size. From the first equation of the system (4.4), we have that

$$\tilde{A}_1 \tilde{x}_1 + \tilde{A}_2 \tilde{x}_2 = \tilde{b}_1. \quad (4.5)$$

As we did in the 1D case, we discard $\tilde{A}_2 \tilde{x}_2$ and find a regularized solution of $\tilde{A}_1 \tilde{x}_1 = \tilde{b}_1$. Once we have an approximation of \tilde{x}_1 , we obtain \tilde{x}_2 . The algorithm that follows these basic ideas is contained in the

TABLE 4.1: Two-Level Method

Input: A, b, x_0
Output: x
1. $\tilde{b}_1 = \widetilde{W}_1^T (b - Ax_0)$
2. $\tilde{x}_0 = \widetilde{W}_1^T x_0$
3. $\tilde{A}_1 = \widetilde{W}_1^T A \widetilde{W}_1$
4. $\tilde{x}_1 = \min_{\tilde{x}_1} \left\{ \ \tilde{A}_1 \tilde{x}_1 - \tilde{b}_1\ _2^2 + \lambda^q \ L \widetilde{W}_1 (\tilde{x}_1 + \tilde{x}_0)\ _q^q \right\}$
5. $x_{new} = x_0 + \widetilde{W}_1 \tilde{x}_1$
6. $\tilde{r}_{new} = W^T (b - Ax_{new})$
7. $\tilde{x}_2 = \min_{\tilde{x}_2} \left\{ \left\ \begin{bmatrix} \tilde{A}_2 \\ \tilde{A}_4 \end{bmatrix} \tilde{x}_2 - \tilde{r}_{new} \right\ _2^2 + \lambda^q \ L(x_{new} + \widetilde{W}_2 \tilde{x}_2)\ _q^q \right\}$
8. $x = x_{new} + \widetilde{W}_2 \tilde{x}_2$

Two-Level Method of Table 4.1, where

$$\widetilde{W}_1^T = W_1^T \otimes W_1^T \quad \text{and} \quad \widetilde{W}_2^T = \begin{bmatrix} W_2^T \otimes W_1^T \\ W_1^T \otimes W_2^T \\ W_2^T \otimes W_2^T \end{bmatrix},$$

and the regularization operators L in lines 4 and 7 are 2D discrete first derivative operators defined by

$$L = \begin{bmatrix} I & \otimes & L_1 \\ L_1 & \otimes & I \end{bmatrix}, \quad \text{with} \quad L_1 = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & & \ddots & \\ & & & & \ddots & \\ & & & & & -1 & 1 \end{pmatrix}. \quad (4.6)$$

So far, the extension to the 2D case seems straightforward by following the same steps used to develop

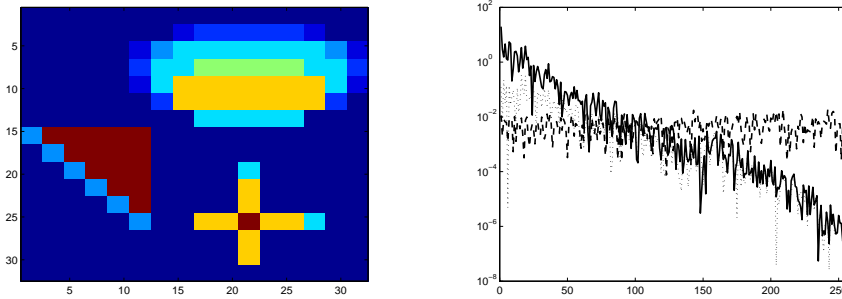


FIGURE 4.3: Restoration by defining $\tilde{x} = W[\hat{x}_1; 0]$ (left) and spectral analysis comparing the values of $\hat{A}_2 \tilde{x}_2$ (dotted line), \hat{b}_1^{true} (solid line) and \tilde{e}_1 (dashed line) (right)

the algorithm for the 1D case. However, we have not yet done a spectral analysis to justify our approach in this case. Looking more carefully at our first equation (4.5), we have that it is equivalent to

$$(\hat{A}_1 \otimes \hat{A}_1) \hat{x}_1 + (\hat{A}_2 \otimes \hat{A}_1) \hat{x}_3 + (\hat{A}_1 \otimes \hat{A}_2) \hat{x}_2 + (\hat{A}_2 \otimes \hat{A}_2) \hat{x}_4 = \hat{b}_1.$$

Our way of thinking has been as follows: since we do not know \hat{x}_3 , \hat{x}_2 , nor \hat{x}_4 , we discard the three terms involving these values. We want to see whether we can always consider what we discard to be noise. For that, we will compute the singular value decomposition (SVD) of $\hat{A}_1 \otimes \hat{A}_1$, and compare the spectral coordinates of the vector $(\hat{A}_2 \otimes \hat{A}_1)\hat{x}_3 + (\hat{A}_1 \otimes \hat{A}_2)\hat{x}_2 + (\hat{A}_2 \otimes \hat{A}_2)\hat{x}_4$ with those of \hat{b}_1^{true} and \hat{e}_1 . Figure 4.3 shows that what we discard can be relevant in the subspace associated with the smoothest modes (i.e. associated with the largest singular values). Suppose we know the values of \hat{x}_2 , \hat{x}_3 , and \hat{x}_4 . If we take these values to the right-hand side, and put them together with b^{true} , we see that the regularized solution of \hat{x}_1 is recovered from information belonging to the subspace associated with the smoothest modes where the noise does not dominate. Then, we need to be careful because the spectral coordinates of what we discarded is information we need to recover \hat{x}_1 . We conclude not only that important information could be lost by this model simplification, but also that we might be incorporating more noise. Nonetheless, this approach still computes a regularized solution, and we will show some numerical results in Section 4.4.

Two-Level Method TL2

Since we are particularly interested in recovering an image with edges, we want to be careful about using all the information we have in the measured data. Our next strategy considers the horizontal edges to be more important than the vertical ones. We note that it is trivial to modify the algorithm to stress the vertical over the horizontal edges. Notice in Figure 4.4, that if we could recover the “downsampled” image \hat{x}_1 and also \hat{x}_3 (see Figure 4.2), we would have a better approximation of the fine-scale vector x than only recovering \hat{x}_1 , that is, setting \hat{x}_2 , \hat{x}_4 to 0 and $\tilde{x} = W[\hat{x}_1; \hat{x}_3; 0; 0]$ (see Figure 4.3) would approximate horizontal features that are missing when defining $\tilde{x} = W[\hat{x}_1; 0; 0; 0]$ (Figure 4.3). With this

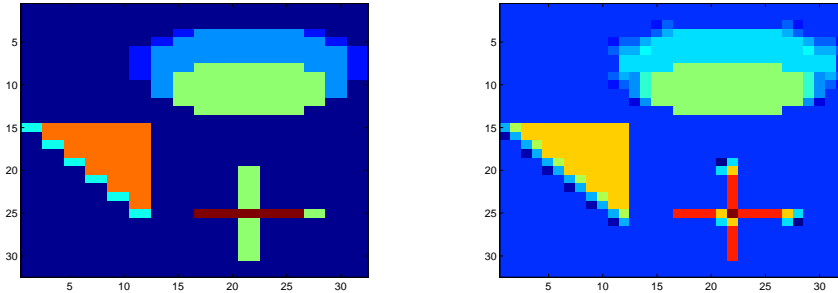


FIGURE 4.4: Two reconstructions we consider: interpolating \hat{x}_1 and \hat{x}_3 (left) and, interpolating \hat{x}_1 , \hat{x}_2 , and \hat{x}_3 (right). Compare with Figure 4.3

observation in mind, we split the system (4.2) in four blocks of equal size, as we did for the 1D case, as follows

$$\begin{bmatrix} \tilde{A}_1 & \tilde{A}_2 \\ \tilde{A}_3 & \tilde{A}_4 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix} = \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \end{bmatrix}, \tilde{x}_1 = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_3 \end{bmatrix}, \tilde{x}_2 = \begin{bmatrix} \hat{x}_2 \\ \hat{x}_4 \end{bmatrix}. \quad (4.7)$$

Then, we apply the Two-Level Method of Table 4.1, where, for this split, we have

$$\tilde{W}_1^T = \begin{bmatrix} W_1^T \otimes W_1^T \\ W_2^T \otimes W_1^T \end{bmatrix} \quad \text{and} \quad \tilde{W}_2^T = \begin{bmatrix} W_1^T \otimes W_2^T \\ W_2^T \otimes W_2^T \end{bmatrix}.$$

Then our first block equation from (4.7) would be

$$\begin{bmatrix} \hat{A}_1 \otimes \hat{A}_1 & \hat{A}_2 \otimes \hat{A}_1 \\ \hat{A}_3 \otimes \hat{A}_1 & \hat{A}_4 \otimes \hat{A}_1 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_3 \end{bmatrix} = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_3 \end{bmatrix} - \begin{bmatrix} \hat{A}_1 \otimes \hat{A}_2 & \hat{A}_2 \otimes \hat{A}_2 \\ \hat{A}_3 \otimes \hat{A}_2 & \hat{A}_4 \otimes \hat{A}_2 \end{bmatrix} \begin{bmatrix} \hat{x}_2 \\ \hat{x}_4 \end{bmatrix}.$$

Following the approach we used in the 1D case, ignoring the matrix-vector product of the right (considering that vector to be noise), we try to find a regularized solution to the resulting system. If the image we want to recover has only horizontal details, this approach can work well. However, if the image we want to recover has vertical details, then when doing a spectral analysis we find that some information gets lost. For example, Figure 4.5 shows that, for values of i small that correspond to smooth frequencies, the spectral coefficients of $\tilde{A}_2 \tilde{x}_2$ are larger than those of $\tilde{e}_1 = [\hat{e}_1; \hat{e}_3]$.

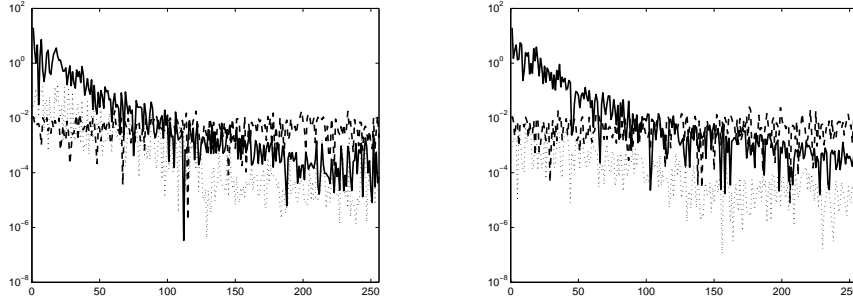


FIGURE 4.5: Spectral analysis comparing the values of $\tilde{A}_2 \tilde{x}_2$ (dotted line), \tilde{b}_1^{true} (solid line) and \tilde{e}_1 (dashed line) for the TL2 method (left) and for the TL3 method (right)

Two-Level Method TL3

In our final approach, we split the system (4.2) as follows

$$\begin{bmatrix} \tilde{A}_1 & \tilde{A}_2 \\ \tilde{A}_3 & \tilde{A}_4 \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{bmatrix} = \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \end{bmatrix}, \tilde{x}_1 = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_3 \\ \hat{x}_2 \end{bmatrix}, \tilde{x}_2 = \hat{x}_4, \quad (4.8)$$

with $\tilde{b}_1, \tilde{x}_1 \in \mathbb{R}^{\frac{3}{4}m^2}$, $\tilde{b}_2, \tilde{x}_2 \in \mathbb{R}^{\frac{m^2}{4}}$, and \tilde{A}_i for $i = 1, \dots, 4$ having the corresponding size.

Our first block equation from (4.8) is

$$\begin{bmatrix} \hat{A}_1 \otimes \hat{A}_1 & \hat{A}_2 \otimes \hat{A}_1 & \hat{A}_1 \otimes \hat{A}_2 \\ \hat{A}_3 \otimes \hat{A}_1 & \hat{A}_4 \otimes \hat{A}_1 & \hat{A}_3 \otimes \hat{A}_2 \\ \hat{A}_1 \otimes \hat{A}_3 & \hat{A}_2 \otimes \hat{A}_3 & \hat{A}_1 \otimes \hat{A}_4 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_3 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_3 \\ \hat{b}_2 \end{bmatrix} - \begin{bmatrix} \hat{A}_2 \otimes \hat{A}_2 \\ \hat{A}_4 \otimes \hat{A}_2 \\ \hat{A}_2 \otimes \hat{A}_4 \end{bmatrix} \hat{x}_4. \quad (4.9)$$

In this case, we expect not to lose much information. That can be seen in Figure 4.5 where it is clear that the spectral coefficients of $\tilde{A}_2 \tilde{x}_2$ are always smaller than those of $\tilde{e}_1 = [\hat{e}_1; \hat{e}_3; \hat{e}_2]$. Therefore, we

expect to get better solutions by applying this final approach that is summarized in Table 4.2 where,

$$\widetilde{W}_1^T = \begin{bmatrix} W_1^T \otimes W_1^T \\ W_2^T \otimes W_1^T \\ W_1^T \otimes W_2^T \end{bmatrix} \quad \text{and} \quad \widetilde{W}_2^T = W_2^T \otimes W_2^T.$$

Clearly, this last approach has a matrix \tilde{A}_1 of larger size than that of our first approach. However, recall that we also need to solve a system involving \tilde{A}_2 and \tilde{A}_4 to get \tilde{x}_2 .

We will discuss pros and cons of each approach further. But first, let us see how we can construct multilevel methods from the Two-Level Approaches TL1, TL2, and TL3 respectively.

4.2.2 Multilevel Methods

Multilevel Method MGM1

We want to extend the Two-Level Method TL1 to a multilevel approach. To do that, we notice that the coarse-grid problem $\tilde{A}_1 \tilde{x}_1 = \tilde{b}_1$ or its equivalent equation

$$(\hat{A}_1 \otimes \hat{A}_1) \hat{x}_1 = \hat{b}_1,$$

looks like the original problem. Therefore, the extension to a multilevel method is straightforward by applying the method again to solve the equation that will appear with this form at each level. We summarize this multilevel approach in Table 4.2 with the following restriction and prolongation operators

$$(\widetilde{W}_1^i)^T = (W_1^i)^T \otimes (W_1^i)^T \quad \text{and} \quad (\widetilde{W}_2^i)^T = \begin{bmatrix} (W_2^i)^T \otimes (W_1^i)^T \\ (W_1^i)^T \otimes (W_2^i)^T \\ (W_2^i)^T \otimes (W_2^i)^T \end{bmatrix},$$

where $(W_1^i)^T, (W_2^i)^T \in \mathbb{R}^{\frac{m}{2^{i+1}} \times \frac{m}{2^i}}$. In this case, the regularization term of lines 2 and 17 in Table 4.2 can be defined by $L^{i+1} = L^i \widetilde{W}_1^i$ or by using the same L , defined in (4.6), on all the levels. We have these two choices in this case because the regularization affects a coarse version of the solution in both the coarse-grid problem and residual correction.

Multilevel Method MGM2

To extend the Two-Level Method TL2 to a multilevel approach, we look again at the coarse-grid problem

$$\begin{bmatrix} \hat{A}_1 \otimes \hat{A}_1 & \hat{A}_2 \otimes \hat{A}_1 \\ \hat{A}_3 \otimes \hat{A}_1 & \hat{A}_4 \otimes \hat{A}_1 \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_3 \end{bmatrix} = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_3 \end{bmatrix}.$$

We will add superscript indices to the vectors to identify them with the level they are related to. We see that we have two blocks from the original image in wavelet domain: \hat{x}_1^0 and \hat{x}_3^0 . When developing TL2, we assumed \hat{x}_1^0 and \hat{x}_3^0 to have the most relevant information to recover x . Now, we want to apply the same

TABLE 4.2: Multilevel V-Cycle: MGM

Input: A^i, b^i, x_0^i
Output: x^i
1. If $i = n$
2. $x^n = \min_{x^n} \{ \ A^n x^n - b^n\ _2^2 + \lambda^q \ L^n(x^n + x_0^n)\ _q^q \}$
3. Else
4. If $i \neq 0$
5. $x_{pre}^i = \text{LSQR}(A^i, b^i, k)$
6. $x^i = x_0^i + x_{pre}^i$
7. Else
8. $x^i = x_0^i$
9. End If
10. $r^i = b^i - A^i x^i$
11. $\tilde{b}_1^{i+1} = (\tilde{W}_1^i)^T r^i$
12. $\tilde{x}_0^{i+1} = (\tilde{W}_1^i)^T x^i$
13. $\tilde{A}_1^{i+1} = (\tilde{W}_1^i)^T A^i (\tilde{W}_1^i)$
14. $\tilde{x}_1^{i+1} = \text{MGM}(\tilde{A}_1^{i+1}, \tilde{b}_1^{i+1}, \tilde{x}_0^{i+1})$
15. $x_{new}^i = x^i + \tilde{W}_1^i \tilde{x}_1^{i+1}$
16. $\tilde{r}_{new}^i = (\tilde{W}^i)^T (b^i - A^i x_{new}^i)$
17. $\tilde{x}_2^{i+1} = \min_{\tilde{x}_2} \left\{ \left\ \begin{bmatrix} \tilde{A}_2^{i+1} \\ \tilde{A}_4^{i+1} \end{bmatrix} \tilde{x}_2^{i+1} - \tilde{r}_{new}^i \right\ _2^2 + \lambda^q \ L^i(x_{new}^i + \tilde{W}_2^i \tilde{x}_2^{i+1})\ _q^q \right\}$
18. $x^i = x_{new}^i + \tilde{W}_2^i \tilde{x}_2^{i+1}$
19. End If

reasoning to recover \hat{x}_1^0 and \hat{x}_3^0 . For this, we transform \hat{x}_1^0 to the wavelet domain, obtaining four vectors $\hat{x}_{11}^1, \hat{x}_{12}^1, \hat{x}_{13}^1$, and \hat{x}_{14}^1 . As before, we assume that only \hat{x}_{11}^1 and \hat{x}_{13}^1 have relevant information to recover \hat{x}_1^0 . The same idea is applied to \hat{x}_3^0 . In the next level, we want to recover $\hat{x}_{11}^1, \hat{x}_{13}^1, \hat{x}_{31}^1$ and \hat{x}_{33}^1 . So we will transform each of these to the wavelet domain and keep only the information we consider relevant, that is, $\hat{x}_{111}^2, \hat{x}_{113}^2, \hat{x}_{131}^2, \hat{x}_{133}^2, \hat{x}_{311}^2, \hat{x}_{313}^2, \hat{x}_{331}^2$, and \hat{x}_{333}^2 . This process is summarized in the diagram of Figure 4.6 where it is easy to see on how many levels each part of the original image x is represented. This idea is not new; we can also see this in terms of *wavelet-packet* basis (see Chapter 8 in [37] for a formal definition). In [10, 39], algorithms to find the best wavelet-packet basis are presented. The basis presented here might not be the optimal wavelet-packet basis. However, we want to emphasize that this is an heuristic to see if we gain something from the wavelet-based approach, and it is also cheaper than computing the optimal wavelet-packet basis.

Then, to be able to use the scheme describe in Table 4.2, we need to define the operators \tilde{W}_1^i and \tilde{W}_2^i that follow the image decomposition described above. First, we define the operators \tilde{W}_1^0 and \tilde{W}_2^0 that take us from level 0 to level 1:

$$(\tilde{W}_1^0)^T = \begin{bmatrix} (W_1^0)^T \otimes (W_1^0)^T \\ (W_2^0)^T \otimes (W_1^0)^T \\ (W_1^0)^T \otimes (W_1^0)^T \\ (W_2^0)^T \otimes (W_1^0)^T \end{bmatrix} \quad \text{and} \quad (\tilde{W}_2^0)^T = \begin{bmatrix} (W_1^0)^T \otimes (W_2^0)^T \\ (W_2^0)^T \otimes (W_2^0)^T \\ (W_1^0)^T \otimes (W_2^0)^T \\ (W_2^0)^T \otimes (W_2^0)^T \end{bmatrix}.$$

Notice that $(\widetilde{W}^0)^T = [(\widetilde{W}_1^0)^T; (\widetilde{W}_2^0)^T] = \tilde{P}(I_2 \otimes W)$, where $W \in \mathbb{R}^{m \times m}$ was defined in (2.3), and \tilde{P} is the corresponding permutation matrix. In general, for level i ,

$$(\widetilde{W}^i)^T = [(\widetilde{W}_1^i)^T; (\widetilde{W}_2^i)^T] = \tilde{P}(I_{2^i} \otimes W),$$

with $W \in \mathbb{R}^{\frac{m}{2^i} \times \frac{m}{2^i}}$, and where $(\widetilde{W}_1^i)^T$ contains the first half of the rows of $(\widetilde{W}^i)^T$ and $(\widetilde{W}_2^i)^T$ contains the second half of the rows of $(\widetilde{W}^i)^T$.

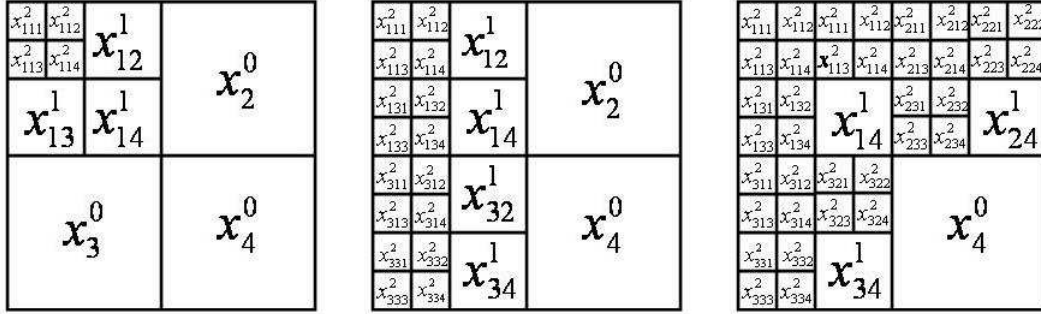


FIGURE 4.6: Wavelet transformation of the image with three levels of the MGM1(left), MGM2(middle), and MGM3(right) multilevel methods

The regularization operator is defined by

$$L^{i+1} = L^i \widetilde{W}_1^i.$$

In doing so, we ensure the appropriate coarse version of the fine-grid L^0 is defined by (4.6) using this change of basis.

Multilevel Method MGM3

Finally, we want to extend the Two-Level Method TL3 to a multilevel approach. We can follow the same approach used to extend TL2 to MGM2. The corresponding diagram in Figure 4.6 shows what we consider to be relevant to reconstruct at each level. Then the corresponding restriction and prolongation operators are defined by

$$(\widetilde{W}^i)^T = \tilde{P}(I_{3^i} \otimes W),$$

with $W \in \mathbb{R}^{\frac{m}{3^i} \times \frac{m}{3^i}}$. In this case, $(\widetilde{W}_1^i)^T$ contains the first three-fourths of the rows of $(\widetilde{W}^i)^T$ and $(\widetilde{W}_2^i)^T$ contains the last fourth of the rows of $(\widetilde{W}^i)^T$.

Again, we define the regularization operator by $L^{i+1} = L^i(\widetilde{W}_1^i)$. Once we have defined \widetilde{W} , \widetilde{W}_1^i , \widetilde{W}_2^i , and L^{i+1} , we can apply the scheme of Table 4.2.

In Section 4.4, we show numerical results that compare these three multilevel methods quantitatively and qualitatively.

4.2.3 Non-separable Case

So far, we have assumed that the blurring operator A is separable. In this section, we work with the non-separable case. To simplify notation, we will define $W_{ij} = W_i \otimes W_j$. Then the system (4.1) in the new basis $W^T A W W^T x = W^T b$ is

$$\begin{bmatrix} W_{11}^T A W_{11} & W_{11}^T A W_{21} & W_{11}^T A W_{12} & W_{11}^T A W_{22} \\ W_{21}^T A W_{11} & W_{21}^T A W_{21} & W_{21}^T A W_{12} & W_{21}^T A W_{22} \\ W_{12}^T A W_{11} & W_{12}^T A W_{21} & W_{12}^T A W_{12} & W_{12}^T A W_{22} \\ W_{22}^T A W_{11} & W_{22}^T A W_{21} & W_{22}^T A W_{12} & W_{22}^T A W_{22} \end{bmatrix} \begin{bmatrix} W_{11}^T x \\ W_{21}^T x \\ W_{12}^T x \\ W_{22}^T x \end{bmatrix} = \begin{bmatrix} W_{11}^T b \\ W_{21}^T b \\ W_{12}^T b \\ W_{22}^T b \end{bmatrix}, \quad (4.10)$$

or

$$\begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} & \hat{A}_{13} & \hat{A}_{14} \\ \hat{A}_{21} & \hat{A}_{22} & \hat{A}_{23} & \hat{A}_{24} \\ \hat{A}_{31} & \hat{A}_{32} & \hat{A}_{33} & \hat{A}_{34} \\ \hat{A}_{41} & \hat{A}_{42} & \hat{A}_{43} & \hat{A}_{44} \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \end{bmatrix} = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_2 \\ \hat{b}_3 \\ \hat{b}_4 \end{bmatrix}. \quad (4.11)$$

We are assuming here that the observations made for the separable case are valid for the non-separable case. For example, we expect \hat{A}_{11} to still be ill-conditioned and to capture the lowest frequency characteristics of A . All of the methods discussed above can be applied to this case. Moreover, we expect to see similar results.

4.3 Computational Issues

Similarly to the 1D case, the Haar wavelet transform preserves the BTTB matrix structure. This implies that we can have fast matrix-vector products of the intermediate-level interior linear least squares problems that must be solved iteratively. Matrix-vector products involving BTTB matrices can be computed in $O(n^2 \log n)$ operations [9]. Figure 4.7 shows the sparsity of the matrix \hat{A}_1^1 corresponding to each different method. It is cheaper to compute the LSQR iterations for MGM1 than for the other two methods since its \hat{A}_1^i is smaller than those for the other methods. For the MGM1 method, after n levels, \hat{A}_1^n has dimension $\frac{m^2}{2^n} \times \frac{m^2}{2^n}$. For the other methods, \hat{A}_1^n is much larger. Thus, for the latter method, it takes more levels to obtain coarsest matrix of a given size. However, the MGM3 method produces the smallest matrices \hat{A}_2^i and \hat{A}_4^i needed in the residual-correction step, which is more expensive than using LSQR in the pre-smoothing step.

4.4 Numerical Results

We conclude with illustrations of the performance of the algorithms presented in this chapter. All computations were performed in MATLAB.

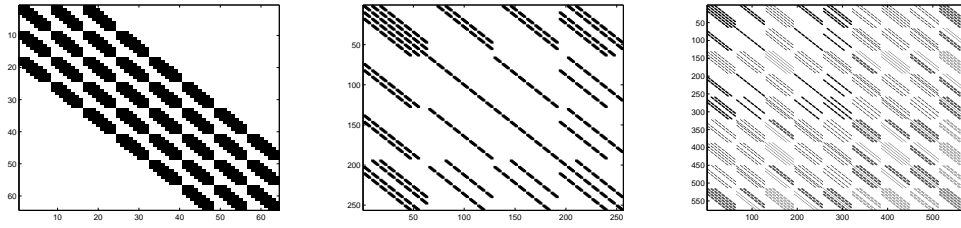


FIGURE 4.7: Sparsity of coarsest matrices involved in the 3 different splits: MGM1 (left), MGM2 (middle), and MGM3 (right)

4.4.1 Example 1: Haar Wavelets

In this first example, the blurring operator is defined by an $m^2 \times m^2$ matrix $A = (2\pi\sigma^2)\tilde{A} \otimes \tilde{A}$, where \tilde{A} is an $m \times m$ symmetric banded Toeplitz matrix whose first row is defined by the vector

$$z = \frac{1}{2\pi\sigma^2} [\exp(-([0 : \text{band} - 1].^2)/(2\sigma^2)); \text{zeros}(1, m - \text{band})].$$

We use $m = 32$, and parameters $\sigma = 3$ and $\text{band} = 9$. The matrix A has a condition number $\kappa(A) = 3.75 \times 10^{11}$. The original image, obtained from the Regularization Tools package by Hansen [25], is shown in Figure 4.8. It is a 32×32 image that is columnwise stacked in the vector x^{true} . The vector b^{true} , containing the noise-free blurred image, is computed as $b^{true} = Ax^{true}$. The entries of the error vector e are normally distributed with zero mean, and normalized such that $\frac{\|e\|_2}{\|b^{true}\|_2} = 0.01$. The contaminated right-hand side of our system is defined by $b = b^{true} + e$ (see Figure 4.8).

For the multilevel solutions shown in Figures 4.9 and 4.10, we define the initial guess $x_0^0 = 0$. The images at the coarsest level have size 8×8 and a Newton method is applied to solve the coarsest-grid correction equations. The regularization operator at the finest grid is defined by

$$L^0 = \begin{bmatrix} I & \otimes & L_1 \\ L_1 & \otimes & I \end{bmatrix}, \quad \text{with} \quad L_1 = \begin{pmatrix} -1 & 1 & & \\ & -1 & 1 & \\ & & \ddots & \ddots \\ & & & -1 & 1 \end{pmatrix}.$$

We define 10 logarithmically spaced values of λ from 10^{-4} to 1. We loop for all values of λ at each level and choose the combination of λ values that make $\frac{\|x^{true} - x^{MGM}(\lambda^{ijk})\|_1}{\|x^{true}\|_1}$ smallest. The notation λ^{ijk} means that for the coarsest level we need $\lambda(i)$, for level 1 we need $\lambda(j)$, and for the finest level we need $\lambda(k)$. Figures 4.9 and 4.10 show the reconstructed images.

We also compute both $x^{LSQR}(k)$ and 1-MGM (λ_i) solutions for comparison (see Figure 4.8). The optimal parameters, k and λ , are chosen such that the 1-norm relative error are minimized. Table 4.3 shows optimal parameters and relative errors. Notice that the LSQR solution is very smooth compared to the solution generated by Newton's method. Therefore, it is more useful to compare qualitatively the solutions obtained by these multilevel approaches with the solution produced by Newton's method to equation in line 2 of Table 4.2 on the fine grid ($n = 0$).

TABLE 4.3: Comparison of different methods: Relative errors

LSQR (41)	Newton (λ^2)	3-MGM (50, $\lambda^{5,6,8}$)
0.6650	0.4925	0.5459

We observe that the multilevel solutions and the fine-grid solution have similar quality. However, the multilevel methods apply fewer Newton iterations on each level (see Tables 4.4 and 4.5). We also observe that MGM3 is the most competitive method in terms of quality and computational cost. Although, to have a more fair comparison, we need to run more levels of MGM3.

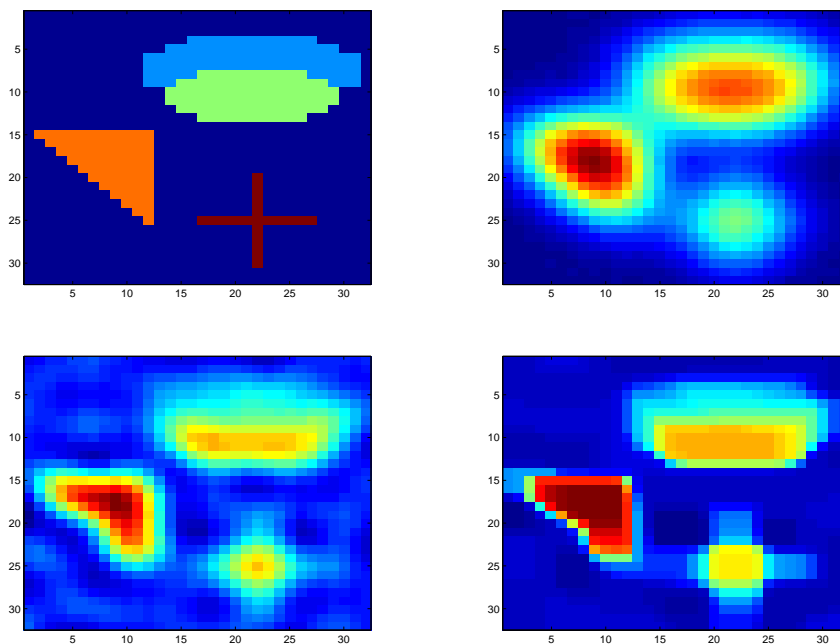


FIGURE 4.8: Image we want to recover, x^{true} (top left), blurred, noisy image, b (top right), LSQR solution (bottom left) and Newton (bottom right).

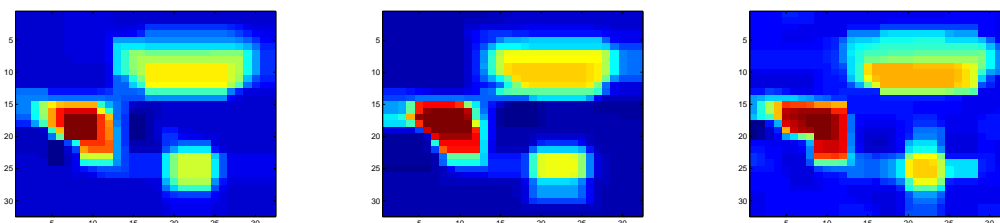


FIGURE 4.9: Two-Level solutions: TL1 (left), TL2 (middle), and TL3 (right)

In terms of computational cost for different multilevel methods, we note that, when using sparsity to compute matrix-vector products, it is more expensive to solve the last residual correction step of MGM1 than to compute a few iterations of LSQR in MGM3. Table 4.5 shows the number of nonzeros to give an idea of the cost of the intermediate solves.

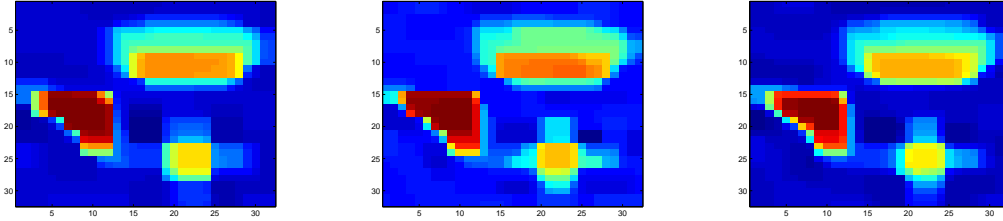


FIGURE 4.10: Three-Level solutions: MGM1 (left), MGM2 (middle), and MGM3 (right)

TABLE 4.4: Comparison of multilevel methods for noise level of 1%

level	MGM split	Newton iter	$\lambda_{opt}(k)$	$\frac{\ x^{true} - x^{MGM}\ _1}{\ x^{true}\ _1}$
1	-	34	λ^5	0.4926
2	1	10,5	$\lambda^{3,10}$	0.5173
2	2	10,5	$\lambda^{1,10}$	0.5048
2	3	9,5	$\lambda^{2,1}$	0.4977
3	1	5,9,7	$\lambda^{5,6,8}(6)$	0.5806
3	2	9,5,5	$\lambda^{7,6,5}(7)$	0.5860
3	3	9,6,6	$\lambda^{2,4,3}(7)$	0.5177

TABLE 4.5: Comparison of multilevel methods for noise level of 5%

level	MGM split	Newton iter	$\lambda_{opt}(k)$	$\frac{\ x^{true} - x^{MGM}\ _1}{\ x^{true}\ _1}$
1	-	34	λ^5	0.5695
2	1	10,5	$\lambda^{5,10}$	0.5621
2	2	10,5	$\lambda^{5,10}$	0.5731
2	3	9,5	$\lambda^{5,4}$	0.5646
3	1	23,29,34	$\lambda^{8,6,5}(6)$	0.5842
3	2	10,9,9	$\lambda^{3,4,10}(7)$	0.5829
3	3	9,4,5	$\lambda^{5,7,4}(1)$	0.5697

TABLE 4.6: Nonzero entries on matrices involved

Problem	MGM1	MGM2	MGM3
LSQR	15376	60512	129600
Coarsest	1156	18136	88348
Correction	12500	34432	38344
Correction	166860	112784	56512

4.5 Conclusions

In this chapter, new edge-preserving multilevel algorithms for image restoration problems are presented. They are extensions of the multilevel approach presented in Chapter 3. The transformation of the problem to the wavelet domain using orthogonal wavelets leads to different multilevel algorithms. In particular, we have discussed three approaches that have the same structure and differ in the prolongation and restriction operators used to move between grids. Numerical experiments illustrate that these multilevel approaches are promising techniques for image restoration problems with edges.

Chapter 5

Modified Total Least Norm Problem

5.1 Introduction

It is well-known that the Total Least Squares (TLS) methodology [16] is an efficient technique for solving blind deconvolution problems. There are several papers on the treatment of the Regularized Total Least Squares Problem [3, 4, 19, 41, 45]. In this chapter, we analyze different methods for solving blind deconvolution problems arising in signal and image restoration.

5.2 Motivation

Assume that A denotes a known $m \times m$ blurring operator, E denotes an unknown perturbation to the blurring matrix, and x^{true} is a vector representation of the signal we want to recover. Our assumption is that $\|A\|_2 \geq \|E\|_2$; we discuss the importance of the relative norms of A and E more below. The blurred exact data, b^{true} , is related to these quantities by $(A + E)x^{true} = b^{true}$. However, what is measured is actually the data vector b , where $b = b^{true} + e$, with e representing additive white noise, such as data measurement error.

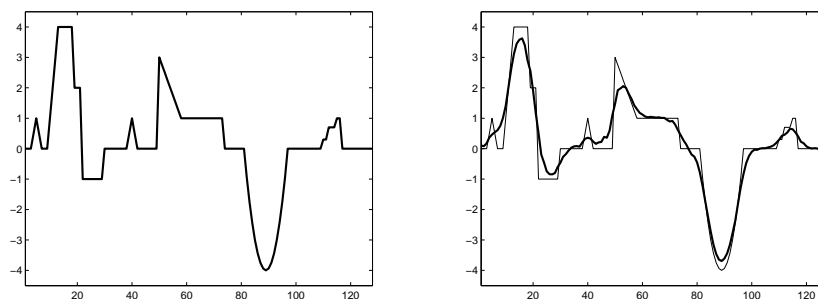


FIGURE 5.1: True solution represented by x^{true} (left) and blurry and noisy measured data, b (right)

Thus, under appropriate assumptions on $A + E$, our goal becomes one of finding a regularized solution to the system

$$(A + E)x = b. \quad (5.1)$$

The need for regularization is justified by the following argument. Since A is a blurring matrix, the singular values of A decay towards 0 with no significant gap in the singular value spectrum. If $\|E\|_2$ is small enough, we expect $A + E$ to still be ill-conditioned because of Corollary 8.6.2 in [17] that says that $|\sigma_i(A + E) - \sigma_i(A)| \leq \|E\|_2$. Figure 5.2 shows the singular values of A and $A + E$ for comparison. Then, solving $(A + E)x = b$, assuming E were known, will give a contaminated solution, as explained in Chapter 1.

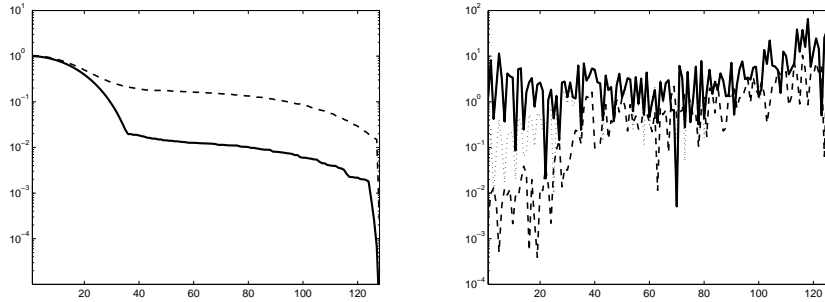


FIGURE 5.2: Singular values of A (solid line) vs. singular values of $A + E$ (dashed line) (left) and comparison of coefficients $|\frac{u_i^T e}{\sigma_i}|$ (dashed line), $|\frac{u_i^T b^{true}}{\sigma_i}|$ (solid line), and $|\frac{u_i^T E x^{true}}{\sigma_i}|$ (dotted line) (right).

We will analyze the influence of E in the system and derive an important restriction on the matrix E that justifies the approach we present in this chapter. First, we rearrange (5.1) to write

$$Ax = b - Ex = b^{true} + e - Ex.$$

Viewed in this way, the right-hand side consists of the noise-free data, b^{true} , additive white noise, e , and signal-correlated noise, Ex . Suppose we have $A = \sum_{i=1}^n u_i \sigma_i v_i^T$. Then, we can write

$$x = \sum_{i=1}^n \left(\frac{u_i^T b^{true}}{\sigma_i} + \frac{u_i^T e}{\sigma_i} + \frac{u_i^T Ex}{\sigma_i} \right) v_i.$$

For small values of i , we expect $|\frac{u_i^T b^{true}}{\sigma_i}|$ to be dominant. However, for a certain index j and $i > j$, the data and noise get mixed (see Figure 5.2). If $|u_i^T Ex| < |u_i^T e|$, then one might as well consider $e - Ex$ as unknown noise, ignore E and try to find a regularized solution to $Ax = b$, which is a computationally easier problem. We therefore assume that although $\|A\|_2 \geq \|E\|_2$, E is not too small in the sense that $\|Ex\|_2 > \|e\|_2$. With this in mind, we propose to solve the following regularized total least norm (RTLNL) problem:

$$\min_{E,x} \{ c \|E\|_F^2 + \|(A + E)x - b\|_2^2 + \lambda^p \|Lx\|_p^p \}, \quad (5.2)$$

where L is taken to be a discrete derivative operator, λ is the regularization parameter, and p is taken such that $1 < p \leq 2$. As we are interested in recovering edge information, we typically use $p \approx 1$. The

choice of scaling constant c will be discussed in context further. In the case where A is invertible, $e = 0$, and $\lambda = 0$, but $E^{true} \neq 0$, the solution to (5.2) is $x = A^{-1}b$ and $E = 0$. In other words, it is not possible to recover x^{true} and E^{true} from this formulation. However, as we can see from the previous discussion, even if $e = 0$, such a solution is worthless because of the effects of term Ex . Indeed, it is the inclusion of the regularization term that, in the case that A is invertible, enables our recovery of a decent approximation of x^{true} , and hopefully, of E^{true} .

5.3 Scaling

The choice of the scaling constant was already considered in [43, 47], for an unregularized version of (5.2) was called the *Scaled Total Least Squares* (Scaled TLS) problem. The authors of this article assumed that no information is available about the absolute size of the error. We make this assumption as well. However, we want to find the correct model for these circumstances. In fact, following this logic, when A is invertible, one might guess that $c = 1$ may not always be an appropriate choice, even though λ is allowed to vary. If $c > 1$, solutions with smaller $\|E\|_F$ but larger misfit are encouraged. When $c < 1$, solutions with larger $\|E\|_F$ and smaller misfit are encouraged.

When $c = 1$, we notice that the quality of the solution of (5.2) for an optimal value of λ is similar to (or slightly worse than) the solution of

$$\min_x \{ \|Ax - b\|_2^2 + \lambda^p \|Lx\|_p^p \}. \quad (5.3)$$

See Figure 5.3 for comparison. A quantitative comparison in terms of relative errors appears in the results section, Section 5.8. We can explain this phenomenon from a statistical point of view. Assuming the errors E_{ij} and e_i are uncorrelated and $E_{ij} \sim N(0, s_1^2)$ and $e_i \sim N(0, s_2^2)$, then we should solve

$$\min_{E,x} \left\{ \frac{1}{s_1^2} \|E\|_F^2 + \frac{1}{s_2^2} \|(A + E)x - b\|_2^2 + \lambda^p \|Lx\|_p^p \right\}, \quad (5.4)$$

or

$$\min_{E,x} \left\{ \frac{s_2^2}{s_1^2} \|E\|_F^2 + \|(A + E)x - b\|_2^2 + \tilde{\lambda}^p \|Lx\|_p^p \right\}, \quad (5.5)$$

where $\tilde{\lambda}^p = s_2^2 \lambda^p$. Notice that c is, then, $\frac{s_2^2}{s_1^2}$. If we assume that $\frac{\|E\|_2}{\|A+E\|_2} > \frac{\|e\|_2}{\|b^{true}\|}$, then $s_1^2 > s_2^2$. Therefore, $c < 1$ seems more appropriate. The assumption that $\frac{\|E\|_2}{\|A+E\|_2} > \frac{\|e\|_2}{\|b^{true}\|}$ is enforced in the numerical examples of this chapter and the next one. There, we shall show the importance of the scaling constant c to get better approximations of x and E . More about this assumption can be found in Section 4.1 in [26], although the motivation is different. Also, in Section 3.6.2 of [30], there is a discussion of what the optimal conditions are for TLS to work well.

5.4 Regularized Structured Total Least Norm

The optimization problem (5.2) is computationally very difficult since we are minimizing with respect to E and x , that is, we have a total of $n^2 + n$ unknowns and only n data components. Furthermore, there is the question of whether we can recover E^{true} . Therefore, we assume the matrix A is Toeplitz, and then ask that E is also Toeplitz. By doing this, we reduce the number of unknowns to $3n - 1$. Many articles have been written on solving (5.2) subject to the constraint that E is Toeplitz or otherwise structured [45, 51]. As noted in previous chapters, when we use wavelets of the sort mentioned in Chapter 3 to derive the coarse-scale and correction equations, Toeplitz structure of the coarse-scale matrices is inherited from the Toeplitz structure of the fine-grid matrix. This suggests that we will ultimately be able to propose an efficient multilevel approach for solving the optimization problem under the restriction that E remains Toeplitz. Indeed, this is the subject of Chapter 6 of this thesis.

Consider the following Toeplitz matrix,

$$E = \begin{bmatrix} t_0 & t_{-1} & t_{-2} & \cdots & t_{-(m-1)} \\ t_1 & t_0 & t_{-1} & \cdots & t_{-(m-2)} \\ t_2 & t_1 & t_0 & \cdots & t_{-(m-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{m-1} & t_{m-2} & t_{m-3} & \cdots & t_0 \end{bmatrix}.$$

Notice that E is generated by the Toeplitz-vector $t = (t_{-(m-1)}, \dots, t_{-1}, t_0, t_1, \dots)$. Thus, the minimization problem (5.2) can be written as

$$\min_{t,x} \{c\|E(t)\|_F^2 + \|(A + E(t))x - b\|_2^2 + \lambda^p \|Lx\|_p^p\}. \quad (5.6)$$

We refer to this problem as the *regularized, structured Total Least Norm*(RSTLN) problem.

5.5 Modified Total Least Norm

Assume that we have initial guesses E_0 and x_0 . We can write a residual as $r_0 = b - (A + E_0)x_0$. We look for corrections E_c and x_c such that $E = E_0 + E_c$ and $x = x_0 + x_c$. Then, we have

$$(A + E_0 + E_c)(x_0 + x_c) = b,$$

which is equivalent to

$$(A + E_0 + E_c)x_c + E_c x_0 = r_0.$$

Defining $\tilde{A} = A + E_0$, $\tilde{b} = r_0$, and $\tilde{E} = E_c$, we obtain what we refer to as the *residual equation*

$$(\tilde{A} + \tilde{E})\tilde{x} + \tilde{E}x_0 = \tilde{b}.$$

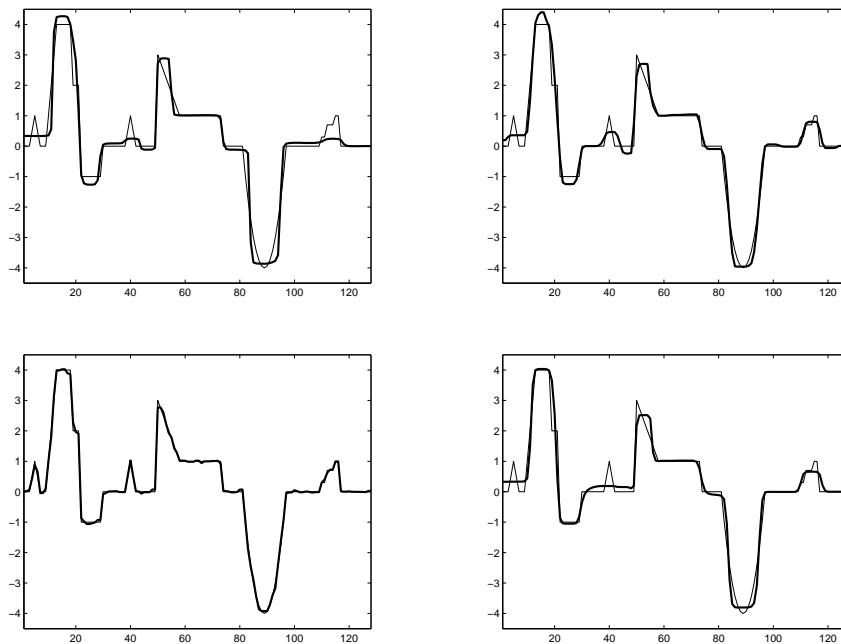


FIGURE 5.3: RTLN solution (top left), Tikhonov ignoring E (top right), Tikhonov solution using true E , E^{true} (bottom left), and $RSTLN$ solution (bottom right).

Recall that \tilde{A} , \tilde{b} , and x_0 are known, and that \tilde{x} , \tilde{E} need to be computed. Further, \tilde{E} is assumed to be Toeplitz and generated by a Toeplitz vector \tilde{t} . In accordance with the approach above, this suggests we ought to solve the following variant of the RSTLN problem, which we call the *modified* RSTLN(MRSTLN) problem:

$$\min_{\tilde{t}, \tilde{x}} \left\{ c \|\tilde{E}(\tilde{t})\|_F^2 + \|(\tilde{A} + \tilde{E}(\tilde{t}))\tilde{x} + \tilde{E}(\tilde{t})x_0 - \tilde{b}\|_2^2 + \lambda^p \|L\tilde{x}\|_p^p \right\}. \quad (5.7)$$

Note that in (5.7) the regularization is only applied to \tilde{x} rather than $x_0 + \tilde{x}$. This choice is based on our limited experience, and appropriate adjustments to the regularization term are the subject of future research.

5.6 Alternating Algorithm

Here we present an algorithm where we first compute the solution of a MRSTLN problem to get E , and then use that E to get a Tikhonov solution applied to the problem $(A + E)x = b$. Then, we use this solution as a starting guess x_0 to again solve the MRSTLN problem, assuming that $E_0 = 0$. This algorithm is summarized in Table 5.1. In the numerical results section, we show some examples and convergence plots. Although we do not have a rigorous analysis that justifies the use of this approach, the results in Section 5.8 show that it is promising and deserves further investigation. We believe that this approach is working because it allows us to effectively constrain the optimization to a subspace where we expect improvement can be made to enhance edge information.

TABLE 5.1: Alternating Algorithm

Input: A, b
Output: E, x

1. $i = 1, x_0 = 0, E_0 = 0$
2. While ($i < \text{Max Iter}$) and ($\frac{\|E_i - E_{i-1}\|_F}{\|E_{i-1}\|_F} > \text{tol}$) and ($\frac{\|x_i - x_{i-1}\|_2}{\|x_{i-1}\|_2} > \text{tol}$)
3. $\tilde{b} = b - Ax_0$
4. $[\tilde{E}, \tilde{x}] = \min_{\tilde{t}, \tilde{x}} \{c\|\tilde{E}(\tilde{t})\|_F^2 + \|(A + \tilde{E}(\tilde{t}))\tilde{x} + \tilde{E}(\tilde{t})x_0 - \tilde{b}\|_2^2 + \lambda^p \|L\tilde{x}\|_p^p\}$
5. $E_i = \tilde{E}$
6. $x_i = \min_x \{\|(A + E)x - b\|_2^2 + \lambda^p \|Lx\|_p^p\}$
7. $i = i + 1, x_0 = x$
8. End while

5.7 Computational Issues

Recall that we need to solve the minimization problem (5.7). We shall follow the same methodology used in [45] to solve the STLS problem. Define the vector t to be the Toeplitz-vector corresponding to E . Then, problem (5.7) can be written as

$$\min_{x,t} \{c\|Dt\|_2^2 + \|Ax + E(x + x_0) - b\|_2^2 + \lambda^p \|Lx\|_p^p\}, \quad (5.8)$$

or as

$$\min_{x,t} \{c\|Dt\|_2^2 + \|Ax + Ft - b\|_2^2 + \lambda^p \|Lx\|_p^p\}, \quad (5.9)$$

where D is the diagonal matrix with entries given by the vector

$$d = (1, \sqrt{2}, \dots, \sqrt{n-1}, \sqrt{n}, \sqrt{n-1}, \dots, \sqrt{2}, 1)$$

and F is a matrix such that $Ft = E(x + x_0)$. Note that F is also Toeplitz but not square. To see that consider the following small example

$$\begin{bmatrix} x_4 & x_3 & x_2 & x_1 & 0 & 0 & 0 \\ 0 & x_4 & x_3 & x_2 & x_1 & 0 & 0 \\ 0 & 0 & x_4 & x_3 & x_2 & x_1 & 0 \\ 0 & 0 & 0 & x_4 & x_3 & x_2 & x_1 \end{bmatrix} \begin{bmatrix} t_{-3} \\ t_{-2} \\ t_{-1} \\ t_0 \\ t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} t_0 & t_{-1} & t_{-2} & t_{-3} \\ t_1 & t_0 & t_{-1} & t_{-2} \\ t_2 & t_1 & t_0 & t_{-1} \\ t_3 & t_2 & t_1 & t_0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}.$$

Then, the Toeplitz-vector that generates F is given by $(0, 0, 0, x_1, x_2, x_3, x_4, 0, 0, 0)$. We apply a Quasi-Newton method to solve this problem where the direction is given by the Least Squares problem

$$\min_{\Delta t, \Delta x} \left\| \begin{bmatrix} F & A + E \\ \sqrt{c}D & 0 \\ 0 & \lambda D_q^{1/2} L \end{bmatrix} \begin{bmatrix} \Delta t \\ \Delta x \end{bmatrix} - \begin{bmatrix} -r \\ \sqrt{c}Dt \\ -\frac{1}{q-1} D_q^{1/2} Lx \end{bmatrix} \right\|_2^2, \quad (5.10)$$

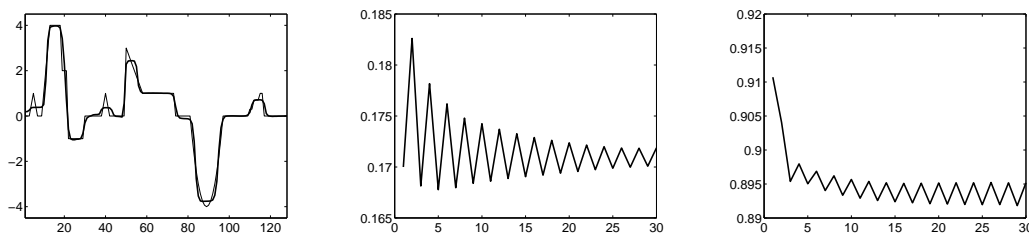
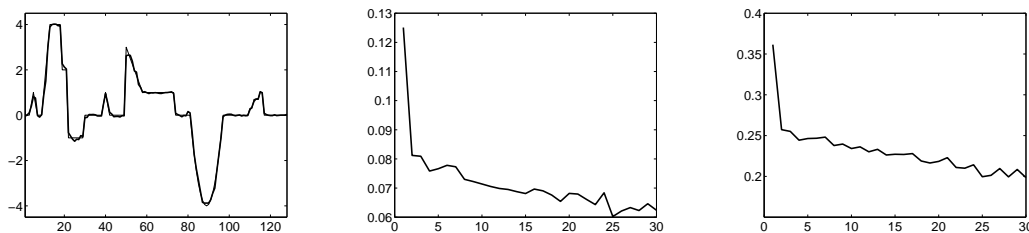
TABLE 5.2: Example 1: Relative errors obtained by applying different methods

	Tikhonov	True E	TLN	RSTLN ($c = 1$)	RSTLN ($c = 0.1$)
$err(x)$	0.1864	0.0401	0.2556	0.2012	0.1379
$err(E)$	1	0	0.9648	0.9106	0.3615

TABLE 5.3: Example 1: Relative errors obtained after 30 iterations of the alternating method for different values of c .

	$c = 1$	$c = 0.6$
$err(x)$.1755	0.0679
$err(E)$.8939	0.2127

the alternating algorithm of Table 5.1. We define 10 logarithmically spaced values of λ from 10^{-3} to 1. The optimal parameters, λ , for lines 5 and 7 in the alternating algorithm are chosen to minimize $\frac{\|x(\lambda) - x^{true}\|_1}{\|x^{true}\|_1}$. We compare convergence of the alternating algorithm for values of $c = 1$ and $c = 0.1$ in the MRSTLN problem of line 5. We conclude that the alternating method gives as better solutions than applying only one iteration (see Tables 5.2 and 5.3 for comparison) and its accuracy is very good with respect to x and E .

FIGURE 5.4: Example 1: Solution of alternating minimization algorithm with $c = 1$ (left), convergence of relative error of x (middle), and convergence of relative error of E (right).FIGURE 5.5: Example 1: Solution of alternating minimization algorithm applied with $c = 0.1$ (left), convergence of relative error of x (middle), and convergence of relative error of E (right).

5.8.2 Example 2

We set up the next example as we did the first one. The matrix A was constructed with parameters $\sigma = 3$ and $\text{band} = 10$. Thus, its condition number is $\kappa(A) = 4.9 \times 10^8$. The perturbation matrix E^{true}

TABLE 5.4: Example 2: Relative errors obtained by applying different methods.

	Tikhonov	True E	TLN	RSTLN ($c = 1$)	RSTLN ($c = 0.6$)
$err(x)$	0.2319	0.0500	0.2588	0.2288	0.2167
$err(E)$	1	0	0.9852	0.9600	0.9071

is a nonsymmetric Toeplitz matrix with same bandwidth as A . Its entries are normally distributed with zero mean, and the standard deviation is chosen such that $\frac{\|E\|_2}{\|A\|_2} = 0.2$. The elements of the noise vector e are normally distributed with zero mean, and the standard deviation is chosen such that $\frac{\|e\|_2}{\|b^{true}\|_2} = 0.01$. The noisy right-hand side of our system is shown in Figure 5.6. The experiments were run in the same way as in the Example 1, except that we also run the alternating algorithm changing the values of c at each iteration. In this example, we see that by using a decreasing c (in this example, we use $c_0=0.6$ and $c = c/1.1$) we get better solutions than with keeping c fixed with value 1 or 0.1 (see Figure 5.7).

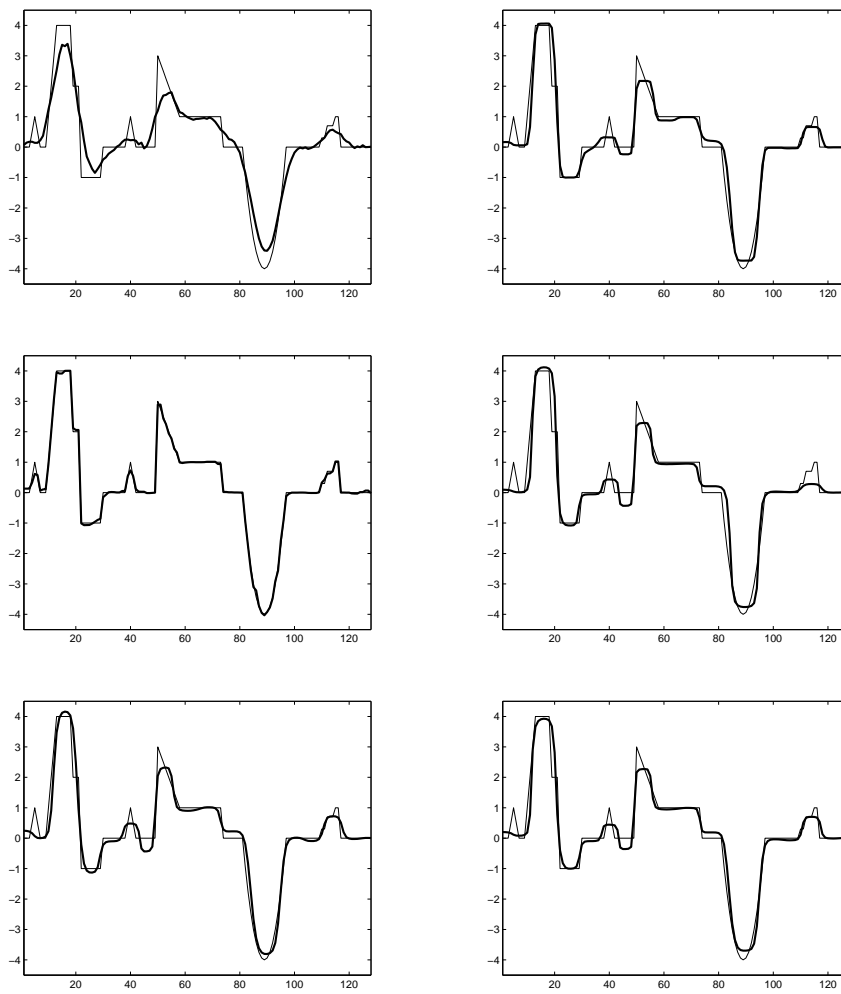


FIGURE 5.6: Example 2: Blurred, noisy right-hand side b (top left), Tikhonov solution ignoring E (top right), Tikhonov solution using true E , E^{true} (middle left), RTLS solution, (middle right), RSTLS solution with $c = 1$ (bottom left), and MRSTLN solution with $c = 0.1$ (bottom right).

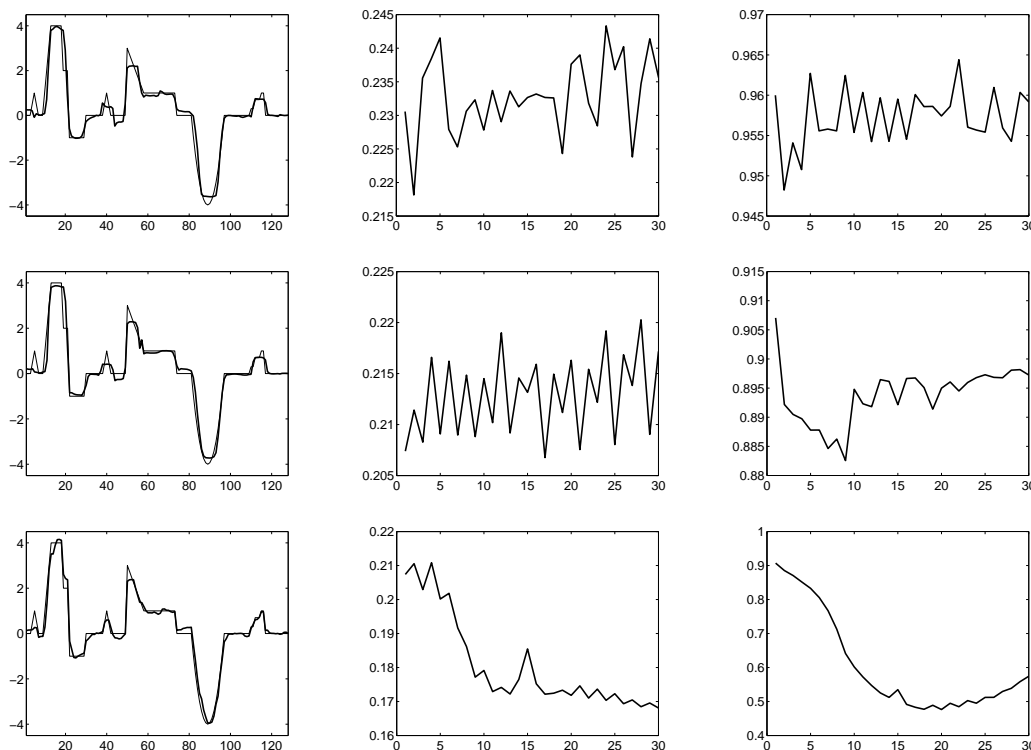


FIGURE 5.7: Example 2: Solution of alternating minimization algorithm with $c = 1$ (top left), convergence of relative error of x (top middle), and convergence of relative error of E (top right). Solution of alternating minimization algorithm with $c = 0.6$ (middle left), convergence of relative error of x (middle middle), and convergence of relative error of E (middle right). Solution of alternating minimization algorithm with decreasing value of c (bottom left), convergence of relative error of x (bottom middle), and convergence of relative error of E (bottom right).

TABLE 5.5: Example 2: Relative errors obtained by applying 30 iterations of the alternating method with different values of c .

	$c = 1$	$c = 0.6$	$c = c/1.1, c_0 = .6$
$err(x)$.2356	0.2171	0.1681
$err(E)$.9592	0.8972	0.5743

5.9 Conclusions

In this chapter, we presented a new model that incorporates a scaling parameter, c , and an initial guess of x , x_0 , into the already known RSTLN problem. Also, we presented an alternating algorithm that uses this new model to obtain better reconstruction not only of x , but also of E . We showed the influence of the scaling parameter c in this algorithm is of great importance. Although we do not have a rigorous analysis that justifies the use of this approach, the results in Section 5.8 show that it is promising and deserves further investigation.

In the next chapter, we present a multilevel method for blind deconvolution problems. We will see that the new MRSTLN problem is the most appropriate choice for the coarse-grid correction.

Chapter 6

Multilevel Methods for Blind Deconvolution

6.1 Introduction

Methods developed previously for solving Tikhonov-Regularized Total Least Norm problems required that the perturbation operator have special structure (e.g. Toeplitz structure, displacement structure) in order to make the minimization problem computationally tractable [45]. Our goal is to gain efficiency by means of a multilevel approach. Therefore, we present an extension of the multilevel method presented in Chapter 3 that solves the modified, structured total least norm (MRSTLN) problem described in Chapter 5. This multilevel method has the ability to recover both the blurring operator and edge information of the solution. It also relies on the Haar wavelet decomposition to move between grids, which has the benefit that, when one enforces a Toeplitz structure on E , this structure is preserved on all levels. Then, to make the problem more computationally tractable, we can use the advantages given not only by a multilevel method, but also by working with Toeplitz matrices. We have found only one article [59] in the literature that applies a multilevel approach to the TLS problem, and this method has only been applied to the TLS problem, not the RSTLN or MRSTLN problems discussed here. There, the authors make use of wavelets, as we do, but develop a very different algorithm: their approach does not include a pre-smoothing step, its prolongation operator is made by padding zeros, and it also assumes all high frequency components to be zero. Consequently, the method presented here has more capabilities.

The structure of this chapter is as follows. First, we will introduce a multilevel method to solve blind deconvolution problems. Then, we will discuss how to solve the coarse-grid and residual corrections. Finally, numerical results are presented to illustrate that this is a promising technique for blind deconvolution problems. Moreover, we illustrate the efficiency of combining this multilevel method with the alternating method presented in Chapter 5.

6.2 Algorithm

Following the discussion from the previous chapter, we restate our blind deconvolution problem described in Chapter 1 as finding x and E such that

$$(A + E)x + Ex_0 = r,$$

where the $m \times m$ matrix A is a known blurring operator, x_0 is an initial guess for x , and $r = b - Ax_0$ is the residual with b being the blurred, noisy signal. This problem can be written in the wavelet domain as

$$(\hat{A} + \hat{E})\hat{x} + \hat{E}\hat{x}_0 = \hat{r}.$$

In other words, if W is the matrix for the 1D Haar matrix transform, then $\hat{A} = W^T A W$, $\hat{E} = W^T E W$, $\hat{x} = W^T x$, $\hat{x}_0 = W^T x_0$, and $\hat{r} = W^T r$. We partition the transformed problem into blocks of size 2^{k-1} to obtain the following problem

$$\begin{bmatrix} \hat{A}_{11} + \hat{E}_{11} & \hat{A}_{12} + \hat{E}_{12} \\ \hat{A}_{21} + \hat{E}_{21} & \hat{A}_{22} + \hat{E}_{22} \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} \hat{E}_{11} & \hat{E}_{12} \\ \hat{E}_{21} & \hat{E}_{22} \end{bmatrix} \begin{bmatrix} \hat{x}_{01} \\ \hat{x}_{02} \end{bmatrix} = \begin{bmatrix} \hat{r}_1 \\ \hat{r}_2 \end{bmatrix}. \quad (6.1)$$

From the first block equation of (6.1), we have

$$(\hat{A}_{11} + \hat{E}_{11})\hat{x}_1 + \hat{E}_{11}\hat{x}_{01} = \hat{r}_1 - (\hat{A}_{12} + \hat{E}_{12})\hat{x}_2 - \hat{E}_{12}\hat{x}_{02}.$$

We make the following remarks:

- \hat{x}_1 is a representation of x on a coarse grid,
- \hat{A}_{11} is still ill-conditioned (see Section 3.2.2),
- \hat{E}_{11} still has random character,
- $\|\hat{A}_{11}\|_2 \geq \|\hat{E}_{11}\|_2$, and
- the term $(\hat{A}_{12} + \hat{E}_{12})\hat{x}_2 - \hat{E}_{12}\hat{x}_{02}$ can be treated as noise.

All of these remarks but the last one are related to important properties of the fine-grid problem that are carried over to coarse-grid problems. We assume that the entries in E are uncorrelated with normal distribution of zero mean and standard deviation s_1 . Then, since W is orthogonal, we have that $\hat{E} = W^T E W$ has also uncorrelated entries with the same normal distribution as E (see Theorem 2.3.10 in [20]). In particular, \hat{E}_{11} also has independent and identically distributed entries with zero mean and standard deviation s_1 .

Recall from Chapter 2, that the first singular value of \hat{A}_{11} , $\hat{\sigma}_1$ is almost the same as the first singular value of A , σ_1 . Then, since $\|A\|_2 = \sigma_1$, $\|A_{11}\|_2 \approx \|A\|_2$. On the other hand, we know that the expected value of the 2-norm of E satisfies that $\mathcal{E}(\|E\|_2) = s_1\sqrt{m}$. We also know that $\mathcal{E}(\|E_{11}\|_2) = s_1\sqrt{\frac{m}{2}}$. Then, we still have $\|\hat{A}_{11}\|_2 \geq \|\hat{E}_{11}\|_2$, which is a relevant assumption discussed in the previous chapter.

Our last remark is related to the fact that we know neither \hat{x}_2 nor \hat{E}_{12} . Thus, we would like to ignore the terms containing them. That is, we want to see if we can consider $(\hat{A}_{12} + \hat{E}_{12})\hat{x}_2 - \hat{E}_{12}\hat{x}_{02}$ to be noise. For that, we make use of the SVD of \hat{A}_{11} and compare the spectral coordinates of $(\hat{A}_{12} + \hat{E}_{12})\hat{x}_2 - \hat{E}_{12}\hat{x}_{02}$ with \hat{r}_1^{true} and $\hat{E}_{11}\hat{x}_1$. Figure 6.1 shows that the spectral coordinates of $(\hat{A}_{12} + \hat{E}_{12})\hat{x}_2 - \hat{E}_{12}\hat{x}_{02}$ are smaller than those of $\hat{E}_{11}\hat{x}_1$ in the subspace related to smooth modes.

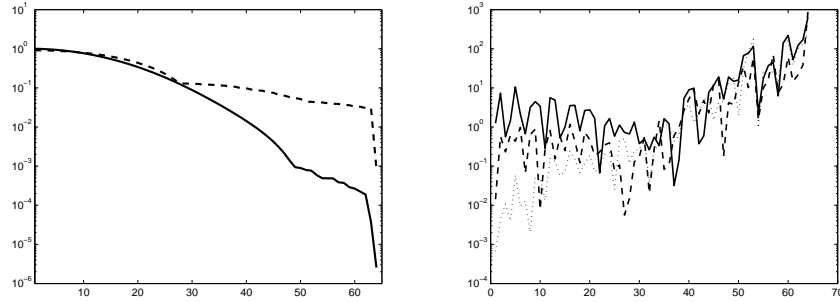


FIGURE 6.1: Singular values of \hat{A}_{11} (solid line) vs singular values of $\hat{A}_{11} + \hat{E}_{11}$ (dashed line) (left) and comparison of coefficients $|u_i^T \hat{E}_{11} \hat{x}_1 / \sigma_i|$ (dashed line), $|u_i^T \hat{r}_1^{true} / \sigma_i|$ (solid line), and $|u_i^T ((\hat{A}_{12} + \hat{E}_{12})\hat{x}_2 - \hat{E}_{12}\hat{x}_{02}) / \sigma_i|$ (dotted line)(right).

Then, the equation

$$(\hat{A}_{11} + \hat{E}_{11})\hat{x}_1 + \hat{E}_{11}\hat{x}_{01} = \hat{r}_1$$

becomes our coarse-grid problem. Furthermore, we obtain its regularized solution by solving the MRTLN problem described in Chapter 5

$$\min_{\hat{x}_1, \hat{E}_{11}} \left\{ c \|\hat{E}_{11}\|_F^2 + \|(\hat{A}_{11} + \hat{E}_{11})\hat{x}_1 + \hat{E}_{11}\hat{x}_{01} - \hat{r}_1\|_2^2 + \lambda^p \|L(\hat{x}_1 + \hat{x}_{01})\|_p^p \right\}, \quad (6.2)$$

where L is a first-derivative operator, and $1 < p \leq 2$. We call this problem the coarse-grid correction problem.

We also need to recover the wavelet coefficients \hat{x}_2 as well as the other three blocks of \hat{E} : \hat{E}_{12} , \hat{E}_{21} , and \hat{E}_{22} . For this part, we define a residual-correction step. Assuming that we have approximations \hat{x}_1^* of \hat{x}_1 and \hat{E}_{11}^* of \hat{E}_{11} from the coarse-grid correction, then Equation (6.1) can be rewritten as

$$\begin{bmatrix} \hat{A}_{12} + \hat{E}_{12} \\ \hat{A}_{22} + \hat{E}_{22} \end{bmatrix} \hat{x}_2 + \begin{bmatrix} \hat{E}_{12} \\ \hat{E}_{22} \end{bmatrix} \hat{x}_{02} + \begin{bmatrix} 0 \\ \hat{E}_{21} \end{bmatrix} (\hat{x}_{01} + \hat{x}_1^*) = \begin{bmatrix} \hat{r}_1 \\ \hat{r}_2 \end{bmatrix} - \begin{bmatrix} \hat{A}_{11} + \hat{E}_{11}^* \\ \hat{A}_{21} \end{bmatrix} \hat{x}_1^* - \begin{bmatrix} \hat{E}_{11}^* \\ 0 \end{bmatrix} \hat{x}_{01}. \quad (6.3)$$

TABLE 6.1: Multilevel V-Cycle: MGM

Input: A^i, b^i, x_0^i
Output: x^i, E^i
1. If $i = n$
2. $[x^i, E^i] = \text{Solve (6.2)}$ (see Section 6.3)
3. Else
4. If $i \neq 0$
5. $[x^i, E^i] = \text{Solve (6.2)}$ (see Section 6.3)
6. End If
7. $r^i = b^i - (A^i + E^i)x^i - E^i x_0^i$
8. $b^{i+1} = W_1^T r^i$
9. $A^i = A^i + E^i$
10. $A^{i+1} = W_1^T A^i W_1$
11. $x_0^i = x_0^i + x^i$
12. $x_0^{i+1} = W_1^T x_0^i$
13. $[\hat{x}_1^{(i+1)}, \hat{E}_{11}^{i+1}] = \text{MGM}(A^{i+1}, b^{i+1}, x_0^{i+1})$
14. $x^i = x^i + W_1 \hat{x}_1^{(i+1)}$
15. $r_{new}^{i+1} = W b^i - W A^i W_1 \hat{x}_1^{(i+1)} - \begin{bmatrix} \hat{E}_{11}^{i+1} \\ 0 \end{bmatrix} (\hat{x}_1^{i+1} + x_0^i)$
16. $[\hat{x}_2^{i+1}, \hat{E}_{12}^{i+1}, \hat{E}_{21}^{i+1}, \hat{E}_{22}^{i+1}] = \text{Solve (6.4)}$ (see Section 6.3)
18. $E_c^i = W \begin{bmatrix} \hat{E}_{11}^{i+1} & \hat{E}_{12}^{i+1} \\ \hat{E}_{21}^{i+1} & \hat{E}_{22}^{i+1} \end{bmatrix} W^T$ (see Section 6.3.2)
19. $x^i = x^i + W_2 \hat{x}_2^{(i+1)}$
20. $E^i = E^i + E_c^i$
21. End If

We denote the right-hand side of Equation (6.3) by \hat{r}_{new} . To solve this problem, we again use the following MRTLN formulation to keep certain smoothness in the solution,

$$\begin{aligned}
\min_{\hat{x}_2, \hat{E}_{12}, \hat{E}_{21}, \hat{E}_{22}} & \quad c \|\hat{E}_{12}\|_F^2 + c \|\hat{E}_{21}\|_F^2 + c \|\hat{E}_{22}\|_F^2 \\
& + \left\| \begin{bmatrix} \hat{A}_{12} + \hat{E}_{12} \\ \hat{A}_{22} + \hat{E}_{22} \end{bmatrix} \hat{x}_2 + \begin{bmatrix} \hat{E}_{12} \\ \hat{E}_{22} \end{bmatrix} \hat{x}_{02} + \begin{bmatrix} 0 \\ \hat{E}_{21} \end{bmatrix} (\hat{x}_{01} + \hat{x}_1^*) - \hat{r}_{new} \right\|_2^2 \\
& + \lambda^p \left\| L(x_0 + W \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}) \right\|_p^p,
\end{aligned} \tag{6.4}$$

where L is a first-derivative operator, and $1 < p \leq 2$ as in Equation (6.2). From a computational perspective, it is very important to note that the regularization is applied at a “finer” scale than the unknown vector \hat{x}_2 for which we are optimizing. Then, our multilevel algorithm is summarized in Table 6.1.

6.3 Computational issues

One nice feature of the Haar decomposition is that it preserves Toeplitz structure between grids. We have shown in Theorem 3.2 that if E is Toeplitz, then \hat{E}_{ij} for $i, j = 1, 2$ are all Toeplitz, and that we can also define the Toeplitz-vector of \hat{E}_{ij} , \hat{e}_{ij} from the Toeplitz-vector of E , \bar{e} . Moreover, the vectors \hat{e}_{ij} for $i, j = 1, 2$ are related.

Corollary 6.1. *Let E be an $m \times m$ matrix with Toeplitz structure, and $m = 2^k$. Then, the $2^{k-1} \times 2^{k-1}$ submatrices \hat{E}_{11} , \hat{E}_{12} , \hat{E}_{21} , and \hat{E}_{22} defined in (6.1) have Toeplitz-vectors that satisfy*

$$\hat{e}_{12} = -\hat{e}_{21} \quad \text{and} \quad \hat{e}_{22} = \hat{e}_{11} - \begin{pmatrix} 2 & 4 & \dots & 4 & 4 \\ 0 & 2 & \dots & 4 & 4 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 2 & 4 \end{pmatrix} \begin{bmatrix} \hat{e}_{21} \\ E_{1,m} \end{bmatrix},$$

where $E_{1,m}$ is the entry 1, m of E .

Proof. We have shown that we can define the Toeplitz-vectors \hat{e}_{ij} of each \hat{E}_{ij} from the Toeplitz-vector \bar{e} of E . From Theorem 3.2, we have the following relations

$$\begin{aligned} \hat{e}_{11} &= t_{11}(1 : 2 : 2m - 3) & \text{where } t_{11} &= (I + Z_m)^2 \bar{e} \\ \hat{e}_{12} &= t_{12}(1 : 2 : 2m - 3) & \text{where } t_{12} &= -(I - Z_m^2) \bar{e} \\ \hat{e}_{21} &= t_{21}(1 : 2 : 2m - 3) & \text{where } t_{21} &= (I - Z_m^2) \bar{e} \\ \hat{e}_{22} &= t_{22}(1 : 2 : 2m - 3) & \text{where } t_{22} &= -(I - Z_m)^2 \bar{e}, \end{aligned}$$

where Z_m is the $(2m - 1) \times (2m - 1)$ “downshift” matrix. From here we can see that $\hat{e}_{21} = -\hat{e}_{12}$. Define $T = 2(I - Z_m^2)^{-1}(I + Z_m^2)$. Then we have that $t_{22} = t_{11} - T t_{21}$. Since \hat{e}_{ij} contains only the first $m - 1$ odd entries of t_{ij} , then we define \tilde{T} by taking only the m odd columns of T and the first $m - 1$ rows of T . We have also the m odd entries of $t_{21} = \begin{bmatrix} \hat{e}_{21} \\ E_{1,m} \end{bmatrix}$. Then, the result follows. \square

This Corollary says that with the knowledge of $E_{1,m}$, \hat{E}_{11} , and \hat{E}_{22} , and the corresponding relations, we can reconstruct the Toeplitz matrix E . Notice that we use just the amount of information needed to define E . That is, to define E we need its corresponding vector \bar{e} of length $2m - 1$. To define \hat{E}_{11} , and \hat{E}_{22} , we need \hat{e}_{11} and \hat{e}_{22} both of length $m - 1$. Then, by knowing \hat{E}_{22} , \hat{E}_{11} , and $E_{1,m}$, we have $m - 1 + m - 1 + 1$ values to reconstruct \hat{E} , and from there we can get $E = W \hat{E} W^T$. Then, we must use this dependence between sub-blocks to formulate the correct minimization problem corresponding to the residual correction step. From now on we assume that A and E are Toeplitz.

Instead of solving problem (6.2), we solve its structured version, the MRSTLN problem

$$\min_{\hat{x}_1, \hat{e}_{11}} \left\{ c \|D \hat{e}_{11}\|_2^2 + \|\hat{A}_{11} \hat{x}_1 + \hat{E}_{11}(\hat{x}_1 + \hat{x}_{01}) - \hat{r}_1\|_2^2 + \lambda^p \|L \hat{x}_1\|_p^p \right\}, \quad (6.5)$$

or its equivalent formulation

$$\min_{\hat{x}_1, \hat{e}_{11}} \left\{ c \|D\hat{e}_{11}\|_2^2 + \|\hat{A}_{11}\hat{x}_1 + F\hat{e}_{11} - \hat{r}_1\|_2^2 + \lambda^p \|L\hat{x}_1\|_p^p \right\}, \quad (6.6)$$

where the vector \hat{e}_{11} is the Toeplitz-vector corresponding to \hat{E}_{11} , D is the diagonal matrix with entries given by the vector

$$d = (1, \sqrt{2}, \dots, \sqrt{\frac{m}{2}} - 1, \sqrt{\frac{m}{2}}, \sqrt{\frac{m}{2}} - 1, \dots, 1),$$

and F is a matrix such that $F\hat{e}_{11} = \hat{E}_{11}(\hat{x}_1 + \hat{x}_{01})$ (see Section 5.7). We apply a Quasi-Newton method to solve this problem where the direction is given by the Least Squares problem

$$\min_{\Delta\hat{e}_{11}, \Delta\hat{x}_1} \left\| \begin{bmatrix} F & \hat{A}_{11} + \hat{E}_{11} \\ \sqrt{c}D & 0 \\ 0 & \lambda D_q^{1/2} L \end{bmatrix} \begin{bmatrix} \Delta\hat{e}_{11} \\ \Delta\hat{x}_1 \end{bmatrix} - \begin{bmatrix} -r \\ \hat{e}_{11} \\ \lambda D_q^{1/2} L \hat{x}_1 \end{bmatrix} \right\|_2^2, \quad (6.7)$$

where $r = \hat{b}_1 - (\hat{A}_{11} + \hat{E}_{11})\hat{x}_1 - \hat{E}_{11}\hat{x}_{01}$ and $D_q = q(q-1)\text{diag}(|\lambda L\hat{x}_1|^{q-2})$. Note that F is a Toeplitz matrix that depends on \hat{x}_1 and \hat{x}_{01} .

Now we want to solve (6.4). From Corollary 6.1, we know that $\hat{e}_{21} = -\hat{e}_{22}$. Then, we can rewrite (6.4) as

$$\begin{aligned} \min_{\hat{x}_2, \hat{e}_{12}, \hat{e}_{22}} & 2c \|D\hat{e}_{12}\|_2^2 + c \|D\hat{e}_{22}\|_2^2 \\ & + \left\| \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix} \hat{x}_2 + \begin{bmatrix} \hat{E}_{12} \\ \hat{E}_{22} \end{bmatrix} (\hat{x}_2 + \hat{x}_{02}) + \begin{bmatrix} 0 \\ -\hat{E}_{12} \end{bmatrix} (\hat{x}_1^* + \hat{x}_{01}) - \hat{r}_{new} \right\|_2^2 \\ & + \left\| \lambda L(x_0 + W^T \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}) \right\|_p^p, \end{aligned} \quad (6.8)$$

or

$$\begin{aligned} \min_{\hat{x}_2, \hat{e}_{12}, \hat{e}_{22}} & 2c \|D\hat{e}_{12}\|_2^2 + c \|D\hat{e}_{22}\|_2^2 + \left\| \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix} \hat{x}_2 + \begin{bmatrix} F_a & 0 \\ F_b & F_a \end{bmatrix} \begin{bmatrix} \hat{e}_{12} \\ \hat{e}_{22} \end{bmatrix} - \hat{r}_{new} \right\|_2^2 \\ & + \lambda^p \left\| L(x_0 + W^T \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}) \right\|_p^p, \end{aligned} \quad (6.9)$$

where $F_a \hat{e}_{12} = \hat{E}_{12}(\hat{x}_2 + \hat{x}_{02})$, and $F_b \hat{e}_{12} = \hat{E}_{12}(-\hat{x}_1^* - \hat{x}_{01})$.

So far, from Corollary 6.1, we have only used the relation $\hat{e}_{12} = -\hat{e}_{21}$. We would like to use all the relations, however, to simplify the minimization subspace of problem (6.8) even more. However, for simplicity we will consider only the case of banded Toeplitz matrices.

6.3.1 Banded Case

Assume that E is Toeplitz and banded. Then, we have that $E_{1,m} = 0$, and therefore, the relation

$$\hat{e}_{22} = \hat{e}_{11} - \begin{pmatrix} 2 & 4 & \dots & 4 & 4 \\ 0 & 2 & \dots & 4 & 4 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 2 & 4 \end{pmatrix} \begin{bmatrix} \hat{e}_{21} \\ E_{1,m} \end{bmatrix}$$

from Corollary 6.1 becomes

$$\hat{e}_{22} = \hat{e}_{11} - \begin{pmatrix} 2 & 4 & \dots & 4 \\ 0 & 2 & \dots & 4 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 2 \end{pmatrix} \hat{e}_{21}.$$

If we define $\tilde{T} = \begin{pmatrix} 2 & 4 & \dots & 4 \\ 0 & 2 & \dots & 4 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 2 \end{pmatrix}$, we have $\hat{e}_{22} = \hat{e}_{11} + \tilde{T}\hat{e}_{12}$. Thus, we can rewrite the minimization problem (6.9)

$$\begin{aligned} \min_{\hat{x}_2, \hat{e}_{12}} \quad & 2c\|D\hat{e}_{12}\|_2^2 + c\|D(\hat{e}_{11} + \tilde{T}\hat{e}_{12})\|_2^2 \\ & + \left\| \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix} \hat{x}_2 + \begin{bmatrix} F_a & 0 \\ F_b & F_a \end{bmatrix} \begin{bmatrix} \hat{e}_{12} \\ \hat{e}_{11} + \tilde{T}\hat{e}_{12} \end{bmatrix} - \hat{r}_{new} \right\|_2^2, \\ & + \lambda^p \left\| L(x_0 + W^T \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}) \right\|_p^p. \end{aligned} \quad (6.10)$$

By algebraic computations, we have that

$$\begin{bmatrix} F_a & 0 \\ F_b & F_a \end{bmatrix} \begin{bmatrix} \hat{e}_{12} \\ \hat{e}_{11} + \tilde{T}\hat{e}_{12} \end{bmatrix} = \begin{bmatrix} F_a \\ F_b \end{bmatrix} \hat{e}_{12} + \begin{bmatrix} 0 \\ F_a \end{bmatrix} (\hat{e}_{11} + \tilde{T}\hat{e}_{12}) = \begin{bmatrix} F_a \\ F_b + F_a\tilde{T} \end{bmatrix} \hat{e}_{12} + \begin{bmatrix} 0 \\ F_a \end{bmatrix} \hat{e}_{11}.$$

Then, we can rewrite problem (6.10) as

$$\begin{aligned} \min_{\hat{x}_2, \hat{e}_{12}} \quad & 2c\|D\hat{e}_{12}\|_2^2 + c\|D(\hat{e}_{11} + \tilde{T}\hat{e}_{12})\|_2^2 \\ & + \left\| \begin{bmatrix} \hat{A}_{12} \\ \hat{A}_{22} \end{bmatrix} \hat{x}_2 + \begin{bmatrix} F_a \\ F_b + F_a\tilde{T} \end{bmatrix} \hat{e}_{12} + \begin{bmatrix} 0 \\ F_a \end{bmatrix} \hat{e}_{11} - \hat{r}_{new} \right\|_2^2, \\ & + \lambda^p \left\| L(x_0 + W^T \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix}) \right\|_p^p, \end{aligned} \quad (6.11)$$

or

$$\min_{\hat{x}_2, \hat{e}_{12}} \{c\|r_1\|_2^2 + \|r_2\|_2^2 + \|r_3\|_2^2 + \|r_4\|_p^p\}. \quad (6.12)$$

We apply a Quasi-Newton method to solve problem (6.11), where the direction is given by the Least Squares problem

$$\min_{\Delta\hat{x}_2, \Delta\hat{e}_{12}} \left\| \begin{bmatrix} F_a & \hat{A}_{12} + \hat{E}_{12} \\ F_b + F_a\tilde{T} & \hat{A}_{22} + \hat{E}_{22} \\ 2\sqrt{c}D & 0 \\ \sqrt{c}D\tilde{T} & 0 \\ 0 & \lambda D_q^{1/2}LW_2 \end{bmatrix} \begin{bmatrix} \Delta\hat{e}_{12} \\ \Delta\hat{x}_2 \end{bmatrix} - \begin{bmatrix} r_3 \\ r_1 \\ r_2 \\ \lambda D_q^{1/2}r_4 \end{bmatrix} \right\|_2^2. \quad (6.13)$$

Notice that F_a and F_b are banded Toeplitz matrices that depend on \hat{x}_2 .

6.3.2 Reconstructing E

Once we have the four matrices \hat{E}_i , we want to get the $m \times m$ Toeplitz matrix E . Using MATLAB notation, define the following vectors

$$\begin{aligned} \tilde{c}_1 &= 0.5(\hat{E}_{11}(:, m/2) - \hat{E}_{12}(:, m/2)) \\ \tilde{c}_2 &= 0.5(\hat{E}_{11}(:, m/2) + \hat{E}_{12}(:, m/2)) \\ \tilde{c}_3 &= 0.5(E_{11}(2 : m/2, 1) - E_{12}(2 : m/2, 1)) \\ \tilde{c}_4 &= 0.5(E_{11}(2 : m/2, 1) + E_{12}(2 : m/2, 1)). \end{aligned}$$

Then, define

$$\begin{aligned} \bar{y}(1 : 2 : m) &= \tilde{c}_1, \\ \bar{y}(2 : 2 : m) &= \tilde{c}_2, \\ \bar{y}(m + 1 : 2 : 2m - 2) &= \tilde{c}_3, \\ \bar{y}(m + 2 : 2 : 2m - 2) &= \tilde{c}_4, \\ \bar{y}(2m - 1) &= \frac{1}{4}(\hat{E}_{11}(1, m/2) + 2\hat{E}_{12}(1, m/2) - \hat{E}_{22}(1, m/2)) \end{aligned}$$

From here, we have that the Toeplitz-vector of E is $e = 2(I + Z)^{-1}\bar{y}$.

In the banded case, the last relation to reconstruct E ,

$$\bar{y}(2m - 1) = \frac{1}{4}(\hat{E}_{11}(1, m/2) - 2\hat{E}_{21}(1, m/2) - \hat{E}_{22}(1, m/2)),$$

is equal to zero since the entries of the block matrices needed are all zeros.

TABLE 6.2: Relative errors obtained by solving STLN with different values of c .

	$c=0$	$c=1$	$c=0.1$
$err(x)$	0.1918	0.2012	0.1174
$err(E)$	1	0.9106	0.3180
λ_{opt}	$\lambda(8)$	$\lambda(6)$	$\lambda(3)$

TABLE 6.3: Relative errors obtained by applying different number of levels.

	1 level	2 levels	3 levels	4 levels
$err(x)$	0.1879	0.2422	0.2415	0.2332
$err(E)$	0.7920	0.9336	0.9266	0.9140
iter	9	5,2	5,2,3,2,2	5,2,2,3,2,2,2
λ_{opt}	$\lambda(6)$	$\lambda(6)$	$\lambda(6)$	$\lambda(6)$

which is a scaled discrete approximation to the first derivative operator. For the solutions shown in Figure 5.3, we define 10 logarithmically spaced values of λ from 10^{-3} to 1. The optimal parameters λ are chosen such that $\frac{\|x(\lambda) - x^{true}\|_1}{\|x^{true}\|_1}$ are minimums. Table 5.2 shows relative errors for optimal values of λ .

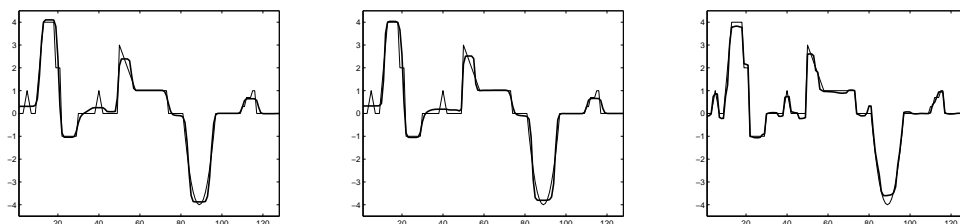
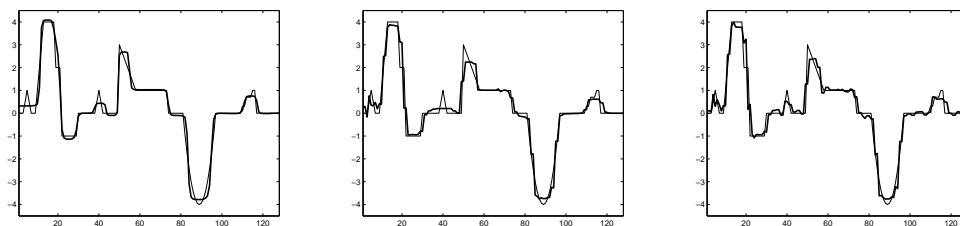
FIGURE 6.3: Solution obtained by ignoring E (left), STLN solution with $c = 1$ (middle), and STLN solution with $c = 0.1$ (right)

FIGURE 6.4: Solutions obtained by multilevel method with different levels: 2-levels (left), 3-levels (middle), and 4-levels (right)

The solution shown in Figure 6.5 is obtained by applying 9 iterations of the alternating method of Table 6.4. The constant c used at all levels and at all iterations is $c = 0.5$. Figure 6.5 also shows the convergence behavior of the alternating method after 30 iterations. We can see that the relative error with respect to E , $err(E)$, is still improving while the relative error with respect to x , $err(x)$, is increasing after 9 iterations. This is due to the fact that we are not using the optimal parameters involved. However, we can see that this solution captures a lot of more details than any other method.

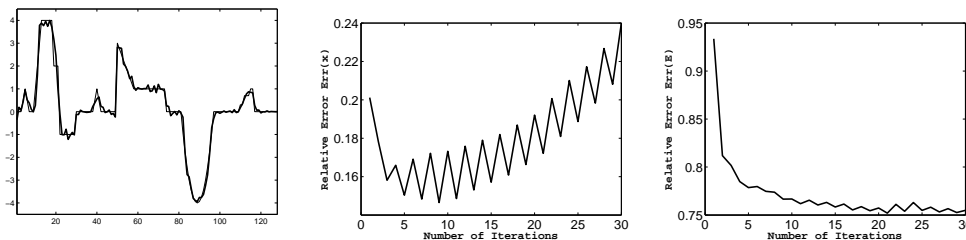


FIGURE 6.5: Solution obtained by applying alternating algorithm (left), convergence behavior with respect to relative error of x (middle), and convergence behavior with respect to relative error of E (right).

TABLE 6.4: Alternating Algorithm with V-cycles

Input: A, b

Output: E, x

1. $i = 1, x_0 = 0, E_0 = 0$
2. While ($i < \text{Max Iter}$) and ($\frac{\|E_i - E_{i-1}\|_F}{\|E_{i-1}\|_F} > \text{tol}$) and ($\frac{\|x_i - x_{i-1}\|_2}{\|x_{i-1}\|_2} > \text{tol}$)
3. $\tilde{b} = b - Ax_0$
4. $[\tilde{E}, \tilde{x}] = \text{MGM}(A, b, x_0)$
5. $E_i = \tilde{E}$
6. $x_i = \min_x \{ \|(A + E)x - b\|_2^2 + \lambda^p \|Lx\|_p^p \}$
7. $i = i + 1, x_0 = x$
8. End while

TABLE 6.5: Relative errors obtained by applying alternating algorithm with a fixed value of $c = 0.5$.

	i=2	i=9	i=30
$err(x)$	0.1581	0.1465	0.2400
$err(E)$	0.8118	0.7666	0.7549

6.5 Conclusions

In this chapter, a new edge-preserving multilevel algorithm for blind deconvolution is presented. The transformation of the problem to the wavelet domain using Haar wavelets leads to a multilevel algorithm that contains an MRSTLN problem on each level. We have shown how to use all the existing connections to obtain efficient solves. Numerical experiments illustrate that this is a promising technique for blind deconvolution problems. Future work includes the extension to 2D image blind deconvolution. In particular, in [13], the authors present an algorithm to solve RSTLS for separable matrices with BTTB structure. If we prove that by using Haar wavelet transforms this structure can be kept between grids, then an extension to the 2D case with BTTB matrices should be straightforward.

Chapter 7

Conclusions and Future Work

In this chapter, we provide a brief review of the main results described in this thesis, and we discuss possible directions for future work.

In this thesis, we analyze three deblurring problems: the signal restoration problem, the image restoration problem, and the blind deconvolution problem. We propose new multilevel methods to solve them, implement these algorithms, and test them.

In Chapter 3, a new edge-preserving multilevel algorithm for signal restoration problems is presented. The transformation of the problem to the wavelet domain using Haar wavelets leads to a multilevel algorithm that preserves

- edge information of the original signal
- characteristics of original blurring operator
- Toeplitz structure.

The use of wavelets to decompose the signal is essential to keep edge information at coarser levels. We also notice that Haar Wavelets pass on some amount of ill-conditioning to coarse-scale blurring operators, and that these always capture the lowest frequency characteristics of the original operator. With these observations in mind, we apply an edge-preserving Tikhonov-like regularization method to solve the coarsened problem at each level. In particular, we discuss how to implement these methods efficiently in the case of Toeplitz matrices. To do so, we show that, in the case of Haar Wavelets, Toeplitz structure is preserved between grids. This leads to intermediate linear system solves inside damped Newton iterations that are structured.

In Chapter 4, we extend the multilevel method of Chapter 3 to the 2D image restoration problem. Since the extension is not straightforward, we first describe two-level methods for a separable deblurring operator. Then, we discuss how to develop their corresponding multilevel methods. Finally, we cover the non-separable case. As in the 1D case, by applying the Haar wavelet transform, the multilevel method preserves the following properties on all levels

- edge information of the original image
- characteristics of original blurring operator
- BTTB structure.

In Chapter 5, we introduced a new Modified, Regularized Total Least Norm approach to solve blind deconvolution problems. This new model incorporates a scaling parameter and an initial guess into the well known Regularized Total Least Norm problem. Also, we present an alternating algorithm that uses this new model to obtain better reconstructions not only of the signal, but also of the blurring operator. By means of numerical examples, we show that the scaling parameter in this algorithm has a great influence on its performance.

Chapter 6 contains an extension of the multilevel method presented in Chapter 3 to blind deconvolution problems. It incorporates the new Modified, Regularized Total Least Norm model introduced in Chapter 5. We also show that the main characteristics of the original problem are also found in the coarse-scale problem. In particular, we show that, when applying orthogonal wavelets, the coarse-scale perturbation matrix has the same statistical properties as the fine-scale matrix. For the implementation of this approach, we focus on problems with Toeplitz matrices. We prove that there are linear relations between sub-blocks that appear in the blurring operator in the wavelet domain. These relations are important in order to formulate the corresponding residual-correction problem efficiently. Moreover, we give details of how to compute solutions of the intermediate linear systems that arise inside the damped Newton iterations.

In all chapters, we present numerical examples that show that these multilevel methods are promising solvers to obtain regularized solutions with edges. Moreover, they are computationally cheaper than standard edge-preserving regularization solvers.

Some future work include:

Parameters

All of the algorithms analyzed in this work have several parameters to fine-tune: for example, regularization parameters, the number of smoothing steps on intermediate grids, or the number of V-cycles. A better understanding of these parameters and incorporation of automatic selection are fundamental for practical purposes.

Discrete Wavelet Transforms

We have only studied the Haar Wavelet transform of Toeplitz matrices. However, there are other structures, such as displacement structure, that are very common not only in deblurring problems but also in other applications where wavelets are a standard tool.

Also, we have only studied the use of orthogonal wavelets. However, there are other types of wavelets such as biorthogonal. For instance, in applications where polar coordinates are used, curvelets are a more

appropriate choice than wavelets. There are several image reconstruction problems where curvelets are applied that could incorporate multilevel ideas to gain computational efficiency.

Extension to 2D and 3D

We have seen that multilevel methods are efficient solvers. Extension of the multilevel methods for signal restoration to the 3D case is an interesting problem, as well as the extensions to 2D and 3D of the multilevel method for blind deconvolution.

Convergence

A natural open problem is a theoretical analysis of convergence of all of these methods.

Bibliography

- [1] A. Aricò and M. Donatelli. A V-cycle Multigrid for multilevel matrix algebras: proof of optimality. *Numerische Mathematik*, 105(4):511–547, 2007.
- [2] A. Aricò, M. Donatelli, and S. Serra Capizzano. V-cycle Optimal Convergence for Certain (Multi-level) Structured Linear Systems. *SIAM Journal on Matrix Analysis and Applications*, 26(1):186–214, 2004.
- [3] A. Beck and A. Ben-Tal. On the solution of the Tikhonov regularization of the total least squares problem. *SIAM J. Optim.*, 17(1):98–118 (electronic), 2006.
- [4] A. Beck, A. Ben-Tal, and C. Kanzow. A fast method for finding the global solution of the regularized structured total least squares problem for image deblurring. *SIAM J. Matrix Anal. Appl.*, 30(1):419–443, 2008.
- [5] M. Belge, M. E. Kilmer, and E. Miller. Wavelet Domain Image Restoration with Adaptive Edge-Preserving Regularization. *IEEE Transactions on Image Processing*, 9(4):598–608, 2000.
- [6] A. Boggess and F. J. Narcowich. *First Course in Wavelets with Fourier Analysis*. Prentice Hall, 2001.
- [7] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial (2nd. Ed.)*. SIAM, Philadelphia, 2000.
- [8] D. Calvetti, B. Lewis, and L. Reichel. On the regularizing properties of the GMRES method. *Numerische Mathematik*, 91:605–625, 2002.
- [9] R. Chan and X. Q. Jin. A family of block preconditioners for block systems. *SIAM Journal on Scientific Computing*, 13, 1992.
- [10] R. R. Coifman and M. V. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Transactions on Informatic Theory*, 38:713–718, 1992.
- [11] M. Donatelli. A Multigrid for image deblurring with Tikhonov regularization. *Numerical Linear Algebra with Applications*, 12:715–729, 2005.
- [12] M. J. Van Fleet. *Discrete Wavelet Transformation: An Elementary Approach with Applications*. Wiley, 2008.

-
- [13] H. Fu and J. Barlow. A regularized structured total least squares algorithm for high-resolution image reconstruction. *Numerical Linear Algebra with Applications*, 391:75–98, 2004.
- [14] P.C. Hansen G. H. Golub and D. P. O’Leary. Tikhonov regularization and total least squares. *SIAM J. Matrix Anal. Appl.*, 21:185–194, 1999.
- [15] V. M. Garcia, L. Acevedo, and A. M. Vidal. Variants of algebraic wavelet-based multigrid methods: Application to shifted linear systems. *Applied Mathematics and Computation*, 202:287–299, 2008.
- [16] G. H. Golub and C. F. Van Loan. An analysis of the total least squares problem. *SIAM J. Numer. Anal.*, 17(6):883–893, 1980.
- [17] G. H. Golub and C. F. Van Loan. *Matrix Computations (3rd. Ed.)*. The Johns Hopkins University Press, Baltimore, Maryland, 1996.
- [18] C. W. Groetsch. *Inverse Problems in Mathematical Sciences*. Vieweg, Wiesbaden, 1993.
- [19] H. Guo and R. A. Renaut. A regularized total least squares algorithm. In *Total least squares and errors-in-variables modeling (Leuven, 2001)*, pages 57–66. Kluwer Acad. Publ., Dordrecht, 2002.
- [20] A. K. Gupta and D. K. Nagar. *Matrix variate distributions*. Chapman & Hall/CRC Monographs and Surveys in Pure and Applied Mathematics. Chapman & Hall/CRC, Boca Raton, FL, 2000.
- [21] W. Hackbusch. *Multigrid methods and applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1985.
- [22] M. Hanke and P. C. Hansen. Regularization methods for large-scale problems. *Surveys Math. Indust.*, 3(4):253–315, 1993.
- [23] M. Hanke and C. R. Vogel. Two-level preconditioners for regularized inverse problems I: Theory. *Numerische Mathematik*, 83:385–402, 1999.
- [24] P. C. Hansen. The discrete Picard condition for discrete ill-posed problems. *BIT*, 30(4), 1990.
- [25] P. C. Hansen. Regularization tools: Matlab package for the analysis and solution of discrete ill-posed problems. *Numerical Algorithms*, 6:1–35, 1994.
- [26] P. C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems*. SIAM, Philadelphia, 1998.
- [27] P. C. Hansen. The L-curve and Its Use in the Numerical Treatment of Inverse Problems. In *Computational Inverse Problems in Electrocardiology*, ed. P. Johnston, *Advances in Computational Bioengineering*, pages 119–142. WIT Press, 2000.
- [28] P. C. Hansen, M. Jacobsen, J.M. Rasmussen, and H. Sørensen. *Methods and Applications of Inversion*, chapter The PP-TSVD algorithm for image restoration problems, pages 171–186. Springer Berlin/Heidelberg, 2000.
- [29] P. C. Hansen, J. G. Nagy, and D. P. O’Leary. *Deblurring Images: Matrices, Spectra and Filtering*. SIAM, Philadelphia, 2006.

-
- [30] S. Van Huffel and J. Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*. SIAM, Philadelphia, 1991.
- [31] B. Kaltenbacher. On the regularizing properties of a full multigrid method for ill-posed problems. *Inverse Problems*, 17:767–788, 2001.
- [32] M. E. Kilmer, P. C. Hansen, and M. I. Español. A projection-based approach to general-form Tikhonov regularization. *SIAM Journal on Scientific Computing*, 29:315–330, 2007.
- [33] M. E. Kilmer and J. G. Nagy. Kronecker product approximations for dense block Toeplitz-plus-Hankel matrices. *Numerical Linear Algebra with Applications*, 14:581–602, 2007.
- [34] M. E. Kilmer and G. W. Stewart. Iterative Regularization and MINRES. *SIAM Journal Matrix Analysis and Applications*, 21:613–628, 1992.
- [35] J. T. King. Multilevel algorithms for ill-posed problems. *Numerische Mathematik*, 61:311–334, 1992.
- [36] F. R. Lin, W. K. Ching, and M. K. Ng. Discrete wavelet transforms for Toeplitz matrices. *Linear Algebra and its Applications*, 370(17):269–285, 2003.
- [37] S. Mallat. *A Wavelet Tour of Signal Processing: The Sparse Way*. Academic Press, third edition, 2008.
- [38] Stephen F. McCormick, editor. *Multigrid methods*, volume 3 of *Frontiers in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1987.
- [39] F. G. Meyer, A. Z. Averbuch, and J. Stromberg. Fast adaptive wavelet packet image compression. *IEEE Transactions on Image Processing*, 9(5):792–800, 2000.
- [40] S. Morigi, L. Reichel, F. Sgallari, and A. Shyshkov. Cascadic Multiresolution Methods for Image Deblurring. *SIAM Journal on Imaging Sciences*, 1:51–74, 2008.
- [41] M. Ng and B. Plemmons. *Blind Deconvolution: Theory and Applications*, chapter Blind Deconvolution and Structured Matrix Computations with Applications to Array Imaging, pages 377–418. CRC Press, 2007.
- [42] C. C. Paige and M. A. Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Transaction on Mathematical Software*, 8:43–71, 1982.
- [43] C. C. Paige and Z. Strakos. Scaled total least squares fundamentals. *Numerische Mathematik*, 91:117–146, 2002.
- [44] F. H. Pereira, S. L. Lopez Verardi, and S. I. Nabeta. A wavelet-based algebraic multigrid preconditioner for sparse linear systems. *Applied Mathematics and Computation*, 182:1098–1107, 2006.
- [45] A. Pruessner and D. P. O’Leary. Blind deconvolution using a regularized structured total least norm algorithm. *SIAM J. Matrix Anal. Appl.*, 24:1018–1037, 2003.
- [46] Å. Björck. *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia, 1996.

-
- [47] B. D. Rao. *Recent Advances in Total Least Squares Techniques and Errors-in-Variables Modelling*, chapter Unified treatment of LS, TLS and truncated SVD methods using a weighted TLS framework. SIAM, Philadelphia, 1997.
- [48] P.C. Hansen R.D. Fierro, G. H. Golub and D. P. O’Leary. Regularization by truncated total least squares. *SIAM J. Sci. Comput.*, 18:1223–1241, 1997.
- [49] L. Reichel and A. Shyshkov. Cascadic multilevel methods for ill-posed problems. *Journal of Computational and Applied Mathematics*. To appear.
- [50] A. Rieder. A wavelet multilevel method for ill-posed problems stabilized by Tikhonov regularization. *Numerische Mathematik*, 75:501–522, 1997.
- [51] J. B. Rosen, H. Park, and J. Glick. Total least norm formulation and solution for structured problems. *SIAM J. Matrix Anal. Appl.*, 17(1):110–126, 1996.
- [52] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [53] Yousef Saad. *Iterative methods for sparse linear systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.
- [54] G. W. Stewart and J. G. Sun. *Matrix Perturbation Theory*. Academic Press, New York, 1990.
- [55] U Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press Inc., San Diego, CA, 2001. With contributions by A. Brandt, P. Oswald and K. Stüben.
- [56] C. R. Vogel. *Computational Methods for Inverse Problems*. SIAM, Philadelphia, 2002.
- [57] G. Wang, J. Zhang, and G. Pan. Solution of Inverse Problems in Image Processing by Wavelet Expansion. *IEEE Transactions on Image Processing*, 4:579–593, 1995.
- [58] W. Zhu, Y. Wang, Y. Deng, Y. Yao, and R. L. Barbour. A Wavelet-Based Multiresolution Regularized Least Squares Reconstruction Approach for Optical Tomography. *IEEE Transactions on Medical Imaging*, 16(2):210–217, 1997.
- [59] W. Zhu, Y. Wang, and J. Zhang. Total least-square reconstruction with wavelets for optical tomography. *Journal of Optical Society of America*, 15(10):2639–2650, 1998.