MULTISCALE AND CURVATURE METHODS FOR THE REGULARIZATION OF THE LINEAR INVERSE PROBLEMS

A Thesis Presented

by

Murat Belge

to

The Department of Electrical and Computer Engineering

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the field of

Electrical Engineering

Northeastern University

Boston, Massachusetts

August 13, 1999

Abstract

In many applications, the recorded data will almost certainly be a degraded version of the original object that is desired, due to the imperfections of physical measurement systems and the particular physical limitations imposed in every application where data are recorded. The situation becomes more complex due to random noise, which is inevitably mixed with the data and may originate from the measurement process, the transmission medium or the recording process. In many practical situations, the degradation can be adequately modeled by a linear system and an additive white Gaussian noise process. The objective is to estimate the original object given the degraded measurements and the matrix describing the forward transformation. Examples of linear inverse problems include image restoration and reconstruction, inverse scattering, seismic analysis, non-destructive testing, etc.

In many cases, the forward transformation acts as a smoothing agent and destroys many high-frequency features in the original object. Mathematically, this means that the system matrix is either ill-conditioned or singular in which case a straightforward inversion is either impossible or results in a solution which is excessively contaminated by the noise. To stabilize the problem, one usually uses a *regularization procedure* where additional information about the original object is incorporated; one may force the computed solution to be smooth, for example. Regularization methods always include a parameter, called the *regularization parameter*, which controls the degree of smoothing or regularization applied to the problem.

If the regularization parameter is too small the solution will be noisy, and if it is too large the solution will be over-smooth. The two basic problems in the regularization of discrete linear inverse problems are the specification of an appropriate prior model that reflects the properties of the original object as closely as possible and the determination of appropriate regularization parameters that produce the closest approximation to the original object under the assumed prior model.

We concentrate our efforts on the solution of the prior specification and the regularization parameter selection problems. In the first part of the thesis, we focus on the image restoration problem. Specifically, we deal with developing multiscale prior models for images to obtain a highly flexible framework for adapting the degree of regularization to the scale and orientation varying features in the image. We demonstrate an efficient half-quadratic algorithm for obtaining the restorations from the observed data.

In the second part, we develop a multi-variate generalization of the conventional L-curve method, the L-hypersurface, for the selection of multiple regularization parameters. The L-curve is one of the simplest and most popular methods for selecting a single regularization parameter. It is based on a plot of the residual norm against the solution norm drawn in a log scale. It has been numerically shown that the corner of the L-curve, which is defined as the point on the L-curve with the maximum curvature, provides a good regularization parameter. We extend the notion of the curvature for plane curves to the notion of Gaussian curvature for hypersurfaces and choose the regularization parameters as those maximizing the Gaussian curvature of the L-hypersurface. Then we deal with the problem of decreasing

the computational effort associated with the practical implementation of the generalized L-curve method.

Acknowledgements

First of all, I would like to thank my advisor, Professor Eric Miller, for his support, encouragement, useful suggestions, and enthusiastic disscussions. Eric has been an exceptional advisor and mentor, both academically and personally. I was very fortunate to have an opportunity to work with him.

Many thanks to Professor Misha Kilmer for her useful suggestions and helpful disscussions. The work on improving the efficiency of the L-hypersurface method as presented in Chapter 5 of this thesis would not be possible without her mathematical expertise. I would also thank Misha for correcting many missing "the's" and serving in my thesis comitte.

In addition, I am grateful to Professors Dana Brooks and Hanoch Lev-Ari for serving in my thesis committee and providing me with valuable feedback on various parts of my thesis on different occasions.

It has been a rewarding and invaluable experience for me to be with talented faculty, students and staff of Electrical and Computer Engineering, and particularly the Digital Signal Processing (CDSP) Center. Special thanks to my friends and peers at Northeastern who have made graduate school life more enjoyable.

Finally, the graduate school took most of its toll on my wife, Aysun, who I love dearly. I dedicate this thesis to her, and thank her for her never-ending patience, understanding, love, and support.

Contents

List of Figures

Chapter 1

Introduction

In many applications in the applied sciences and engineering, the need arises to find a good approximation to an unknown object given a collection of its linearly transformed and noisy measurements. Such inverse problems arise in a variety of fields such as image processing [65, 70, 71], medical imaging [15, 16, 27, 63], computer assisted tomography (CAT, PET) [13,64], non-destructive testing, inverse scattering [17,60], seismic analysis [18], inversion of Laplace transforms, etc. [5, 8, 43, 44, 51]. Of particular interest in this thesis are the image restoration and reconstruction problems for which the goal is to recover an image given its blurred and noise corrupted measurements [2, 52, 65, 70].

A number of well-recognized challenges exist which complicate the process of generating a solution to the underlying inverse problem. Most of the problems encountered have to do with the loss of information associated with the forward transformation which usually acts as a smoothing agent and wipes out the high frequency information in the original object.

This makes the problem ill-posed in the sense that small perturbations in the observed data can result in large, non-physical artifacts in the recovered image [2, 70]. Such instability is typically addressed through the use of a *regularization procedure* that introduces *a priori* information about the original object into the inversion process. The prior information underlying the most commonly used regularization schemes is that the object is basically smooth [2]. While the resulting restorations are less sensitive to noise, it is well known that the smoothness assumption impedes the accurate recovery of important features, especially edges [2, 4, 65, 70]. Regularization methods always include a parameter, called the *regularization parameter*, which controls the degree of smoothing or regularization applied to the problem [56]. Two basic difficulties in finding regularized solutions of inverse problems are the specification of an appropriate prior model that reflects the properties of the original object as closely as possible and the determination of appropriate regularization parameters that produce the closest approximation to the original object under the assumed prior model.

The research presented in this thesis spans these two basic problems related to the solution of discrete linear inverse problems: in the first part we concentrate on the image restoration problem. Specifically, we will deal with developing multiscale prior models for images to obtain a highly flexible framework for adapting the degree of regularization to the scale and orientation varying features in the image. We demonstrate an efficient half-quadratic algorithm [46] for obtaining the restorations from the observed data.

The multiscale image restoration/reconstruction scheme developed in the first part of this thesis requires the selection of multiple regularization parameters and leads to a generalized

L-curve framework, the discussion of which is the subject of the second part. The L-curve is one of the simplest and most popular methods for selecting a single regularization parameter [56, 58, 86]. The method is based on a plot of the size of the solution (measured by an appropriate norm) against the size of the data mismatch for all valid regularization parameters. Intuitive justifications and numerical experiments indicate that the so-called "corner" of the L-curve gives a regularization parameter which provides acceptable estimates in terms of closeness to the original object. We first develop a multi-dimensional extension of the conventional L-curve method to select multiple regularization parameters simultaneously. Then we deal with the problem of decreasing the computational effort associated with the practical implementation of the generalized L-curve method.

1.1 An Overview of Discrete Linear Inverse Problems

Discrete linear inverse problems are a special case of inverse problems in which the observations, $\mathbf{g} \in \mathcal{R}^m$, and the original object, $\mathbf{f} \in \mathcal{R}^n$, are related through a set of linear equations characterized by the ill-conditioned or singular system matrix $\mathbf{H} \in \mathcal{R}^{m \times n}$ often obtained by discretizing a continuous integral equation. In this thesis, we are not particularly interested in the discretization process but rather we will assume that we have a set of noisy linear equations $\mathbf{g} = \mathbf{H}\mathbf{f} + \mathbf{e}$ for which an approximation $\mathbf{f}^*$ to the original object $\mathbf{f}$ is desired. By doing so we will avoid the discussion of regularization techniques phrased in terms of functional analytic language, geared towards infinite-dimensional problems. Regularization can be discussed using elementary linear algebra which is far more accessible to many people

with diverse backgrounds. The linear algebra setting is also much closer to the way in which regularization methods are implemented in practice.

The major difficulty associated with solving discrete linear inverse problems is that the systems are ill-conditioned and sometimes under-determined. Often there is a severe loss of information from $\mathbf{f}$ to $\mathbf{g}$. In particular, if the system is under-determined, $\mathbf{H}^{-1}$ is obviously not well defined. Moreover, even if $\mathbf{H}$ were invertible, since the matrix $\mathbf{H}$ is ill-conditioned there is little control over the propagation of measurement errors from the data to the solution. These observations can be made more precise by introducing operator theoretic terms, but the basic problem is clear: given $\mathbf{g}$ and $\mathbf{H}$, the solution space typically contains many objects, $\mathbf{f}_1$, $\mathbf{f}_2$, with transformed values, $\mathbf{Hf}_1$, $\mathbf{Hf}_2$, close to $\mathbf{g}$ can be very different.

A practical way of coping with the uniqueness and the stability problems is to use a *regularization* procedure [8, 91] which introduces *a priori* information about the original object into the inversion process and helps stabilize the problem. For instance, one may assume that the original object is smooth. Typically the prior information is mathematically expressed in terms of a parametric model. The subset of parameters that are used to control the degree of smoothing or regularization applied to the problem are termed the *regularization parameters* [8, 44, 56]. A small regularization parameter is likely to result in a noisy solution while a large regularization parameter may over-smooth the computed solution. Sophisticated regularization schemes may require the use of several regularization parameters which should be estimated from the data. Methods are available to perform this task [26, 30, 53, 58, 79, 95, 99].

Of particular interest in this thesis are the image restoration and reconstruction problems in which the objective is to estimate a 2-D array of pixels representing a true scene from its linearly transformed and noise corrupted measurements. The distinction between the restoration and reconstruction problems lies in the space in which the observed data is represented. In image restoration problems the data are merely a blurred version of the original image while in image reconstruction problems the data are represented in an entirely different domain than the original image (an example is the Radon transform domain in CAT applications). In image reconstruction problems the system matrix $\mathbf{H}$ is not a degradation matrix but rather a description of the physics of a known measurement process which facilitates the observation of an otherwise unobservable image. A major issue in solving the image reconstruction and restoration (IRR) problems is the computational complexity associated with the solution of large dense sets of linear equations. For a typical 256×256 image, the size of the system matrix is $65,536 \times 65,536$ which makes the problem prohibitively expensive to solve unless $\mathbf{H}$ and the regularization operators are structured to allow some kind of a diagonalization. In IRR problems, computational efficiency is a great virtue and a main motivation of some of the developments presented in this thesis. The second important problem in IRR is the fact that the ultimate authority that will assess the success of a particular IRR algorithm is the human eye, which prefers sharp restorations/reconstructions (tolerating noise in the vicinity of edges) over smooth ones. There are examples in the literature (see for example [2]) showing images which are very close in terms of quantitative measures (MSE, for example, which measures the l_2 norm of the difference between the original and

restored images) but far apart in terms of visual quality. Traditionally, the prior information underlying the most commonly used regularization schemes is that the image is basically smooth [2]. While the resulting restorations are less sensitive to noise, it is well known that the smoothness assumption impedes the accurate recovery of important features, especially edges.

In response to this problem, there has recently been considerable work recently in the formulation of "edge-preserving" regularization methods that result in less smoothing to the areas with large intensity changes in the reconstructed image. These methods necessarily require non-quadratic regularization functions and therefore result in nonlinear image restoration algorithms. Along these lines, *Geman* and *Yang* [46] introduced the concept of "half quadratic regularization" which addresses the nonlinear optimization problem that results from using such functions. Later, *Charbonnier et. al.* [25] built upon the results of this work by providing the conditions for edge preserving regularization functions. They showed that any edge-preserving regularization function can be transformed, through a set of auxiliary variables, to a quadratic function and gave a deterministic algorithm for optimization. Another recent advance in this area is the Total Variation (TV) based image restoration algorithms [10, 11, 69, 98]. In this approach, images are modeled as functions of Bounded Variation [98] which need not be continuous. Therefore, formations of edges are encouraged and the restorations obtained by the TV-based algorithms look sharper than those obtained by conventional techniques, especially if the exact image is piecewise continuous.

With the introduction of the wavelet decomposition and the multiresolution analysis [75],

a new class of IRR schemes, where the original inverse problem is expressed and solved in another domain through some kind of multiscale decomposition, have arisen. It has been recognized empirically that the wavelet decomposition has certain advantages in terms of modeling [74, 75]. Meyer [76] has shown that the wavelet domain is a natural basis for the representation of objects belonging to the Besov Spaces [39, 76, 96]. Besov spaces typically contain objects with discontinuities (edges in 2-D). In terms of statistical language, subbands of wavelet decompositions of "edgy" objects follow generalized Gaussian (GG) distributions. Using the GG distribution for modeling subbands of images has been apparently first reported in [75] in the context of image compression. Later, Donoho and Johnstone [37] introduced what they call *wavelet shrinkage* for smoothing noisy data (de-noising). Wavelet shrinkage takes the wavelet coefficients of the noisy image and shrinks those whose absolute values are smaller than a threshold towards zero. Clearly, the logic behind such an approach is that the smaller wavelet coefficients are due to noise and larger ones carry important features, specifically edges, in the image and hence must be preserved. In a series of papers [33–36, 38] Donoho and Johnstone show that wavelet shrinkage leads to near-optimal noise removal properties when the images are modeled as members of several Besov spaces. Moulin *et. al.* [80] and Simoncelli [88] considered similar de-noising schemes by modeling the subbands of the wavelet decomposition of images stochastically as GG distributed.

Aside from these developments, there is a large body of work on wavelet-based, statistical regularization methods. These methods have largely concentrated on the use of multi-scale smoothness priors [4,77,78,92,100]. While the issue of edge preservation is considered [4,100],

it is based on the processing of the output of an edge detector applied to the noisy data in order to alter the degree of regularization in a multiscale smoothness constraint. Our research in this thesis is different from these methods in that the edge-preservation is directly built into the prior model. Our IRR scheme is motivated by the wavelet shrinkage estimators of Donoho and Johnstone.

1.2 Contributions

In this section, we review the contributions of this thesis to the solution of linear inverse problems and to the regularization parameter selection problem.

1.2.1 Multiscale Image Restoration

In chapter 3 of this thesis we consider a statistically based, wavelet-domain approach to edge-enhanced image restoration. Along the lines of [19, 23, 87], we employ a stochastic interpretation of the regularization process. Specifically, we regard the image as a realization of a random field for which the wavelet coefficients are independently distributed according to generalized Gaussian (GG) distribution laws [12]. This model is motivated by two factors. First, recent work [19] suggests that these models, which have heavier tails than a straight Gaussian distribution, provide accurate descriptions of the statistical distribution of wavelet coefficients in image data. Second, in addition to being a basis for $l^2(\mathcal{R})$ (i.e. square integrable functions defined on the real line), wavelets also are unconditional bases for more exotic function spaces whose members include functions with sharp discontinuities and thus

serve as natural function spaces in which to analyze images [23, 31, 76]. Because the norms in these Besov spaces are nothing more than weighted l_p, $1 \leq p \leq 2$, norms of the wavelet coefficients, it is shown easily that deterministic regularization with a Besov norm constraint is equivalent to the specification of an appropriately parameterized GG wavelet prior model.

We make use of GG wavelet priors in a number of ways. We show that their use in an image restoration problem does in fact significantly improve the quality of edge information relative to more common smoothness priors. We also provide an efficient algorithm for solving the convex, non-linear optimization problem defining the reconstruction. By appropriately structuring the weighting pattern on the wavelet l_p norm, we demonstrate that these models provide an easy and flexible framework for adaptively determining the appropriate level of regularization as a function of the underlying structure in the image; in particular, scale-to-scale or orientation based features. We verify the performance of this restoration scheme on a variety of images comparing the results both to smoothness constrained methods and the TV restorations.

1.2.2 Regularization Parameter Selection

The multiscale image restoration algorithm introduced in chapter 3 leads to a multidimensional generalization of the conventional L-curve method developed in [56] for choosing a single regularization parameter. In chapter 4, we introduce the L-hypersurface method for choosing multiple regularization parameters simultaneously. The notion of curvature

which is used to find a single regularization parameter is extended to the notion of Gaussian curvature. It is shown through numerical examples that the Gaussian curvature of the L-hypersurface as a function of regularization parameters provides valuable information regarding the behavior of the mean square error surface. In particular we show that the regions where Gaussian curvature reaches a local maxima contain acceptable regularization parameters in terms of minimizing the mean square error between the original and reconstructed objects. We provide details regarding the implementation of the L-hypersurface method in typical multiply constrained least squares problems with arbitrary regularization functionals. Furthermore, we combine the L-hypersurface method with the multiscale IRR algorithm to obtain a highly flexible algorithm which determines the appropriate level of regularization as a function of the underlying structure in the image; in particular, scale-to-scale or orientation based features. To reduce the computational complexity associated with computing the Gaussian curvature of the L-hypersurface method, we provide an approximation of the Gaussian curvature through discrete derivatives.

In many cases, evaluating points on the (Gaussian) curvature of the (generalized) L-curve is computationally very demanding and one would prefer using a standard optimization strategy instead of exhaustive search to locate the regularization parameters that correspond to the (generalized) corner. However, the (Gaussian) curvature function possesses many extrema and therefore poses a difficult optimization problem. In chapter 5, we consider an alternative approach for approximating the maximum curvature point which consists of replacing the Gaussian curvature by a surrogate function which is far easier to optimize. We

prove that in the one parameter case, minimization of this surrogate function is essentially equivalent to the maximization of the curvature function. We demonstrate through numerical examples that, although there is little performance loss as compared to the maximization of the Gaussian curvature, the computational burden is orders of magnitude smaller.

1.3 Thesis Organization

The rest of the thesis is organized as follows. Chapter 2 contains background material for linear inverse problems and in particular regularized image restoration and reconstruction. The method of regularization is described in detail. In addition, common regularization parameter selection methods are explained in this chapter. In chapter 3, a multiscale image restoration algorithm for edge-enhanced image restoration will be given. In chapter 4, we introduce a multi-variate generalization of the conventional L-curve method for choosing multiple regularization parameters. In chapter 5, we explore the issue of decreasing the computational complexity of the L-hypersurface method. Finally, in chapter 6, we summarize the results and contributions of this thesis, and indicate future research directions.

Chapter 2

Background

In the first part of this section, a brief overview of the image restoration problem and wavelet theory will be presented. In the second part, common regularization parameter selection methods and, in particular, the L-curve method [56, 58] will be reviewed. Since the conventional L-curve method forms the basis for our generalized L-curve scheme, we review the theory here, before we introduce our algorithms in Section 4.

2.1 Image Restoration Background

In many applications recorded images represent a degraded version of the original scene. For example, the images of extraterrestrial objects observed by ground based telescopes are distorted by atmospheric turbulence [70] while motion of a camera can result in an undesired blur in a recorded image. Despite the different origins, these two cases along with others from a variety of fields, share a common structure where the exact image undergoes

a forward transformation and is corrupted by observation noise [29]. The source of this noise is the disturbance caused by the random fluctuations in the imaging system and the environment. The goal of image restoration is to recover the original image from these degraded measurements.

A grey-scale image, f, can be considered as a collection of pixels obtained by digitizing a continuous scene. The image is indexed by (i, j), $1 \leq i, j \leq 2^J$ (the image size is chosen as a power of 2 for convenience), and the intensity at the position (i, j) is denoted by $f(i, j)$. In image reconstruction and restoration problems, the objective is to estimate the image $f(i, j)$ from its degraded measurements. Mathematically, such a scenario can be adequately represented by the following linear formulation

$$g(k, l) = \sum_{k'} \sum_{l'} H(k, l, k', l') f(k', l') + e(k, l) \tag{2.1}$$

where g contains the sampled values of our degraded image, H is a known operator representing the linear degradation and e is the disturbance which we assume to be white Gaussian noise with variance σ_e^2 and mean zero. The model equation in (2.1) can be represented more compactly as a linear matrix equation by forming the vectors $\mathbf{g}$, $\mathbf{f}$ and $\mathbf{e}$ from the lexicographically ordered elements of the two-dimensional arrays $g(.,.)$, $f(.,.)$ and $e(.,.)$, respectively. In this way we obtain

$$\mathbf{g} = \mathbf{H}\mathbf{f} + \mathbf{e} \tag{2.2}$$

where the vectors $\mathbf{g}$, $\mathbf{f}$ and $\mathbf{e}$ represent, respectively, the lexicographically ordered degraded image, the original image, and the disturbance. The known square matrix $\mathbf{H}$ represents the linear distortion. $\mathbf{H}$ can have very large dimensions ($65,536 \times 65,536$ for a 256×256 image)

and is typically ill-conditioned.

2.1.1 Regularized Image Restoration and Reconstruction

One of the approaches to obtain an estimate of the original image from (2.2) is to minimize the following

$$J(\mathbf{f}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 \tag{2.3}$$

which results in the pseudo-inverse estimate $\mathbf{f}^\dagger$

$$\mathbf{f}^\dagger = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{g} \tag{2.4}$$

where superscript T denotes the transpose of a matrix or a vector. We now assume that we know a singular value decomposition (SVD) of $\mathbf{H}$ [49],

$$\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \quad \mathbf{U},\mathbf{V} \text{ orthogonal}, \mathbf{\Sigma} = \text{diag}(\sigma_1,\ldots,\sigma_n), \quad \sigma_1 \geq \ldots \geq \sigma_n \geq 0\ . \tag{2.5}$$

For $\mathbf{H} \in \mathcal{R}^{m\times n}$ with $m \geq n$, $\mathbf{U}$ is a square, orthonormal $m \times m$ matrix, $\mathbf{V}$ a square, orthonormal $n \times n$ matrix, $\mathbf{\Sigma} = \text{diag}(\sigma_1,\ldots,\sigma_n)$ a rectangular, diagonal $m \times n$ matrix with diagonal entries $\mathbf{\Sigma}_{i,i} = \sigma_i$. The least squares solution corresponding to (2.3) is given by

$$\mathbf{f}^\dagger = \mathbf{V}\mathbf{\Sigma}^\dagger\mathbf{U}^T\mathbf{g} = \sum_{i=1,\sigma_i\neq 0}^{n} \frac{1}{\sigma_i}\left(\mathbf{u}_i^T\mathbf{g}_{true} + \mathbf{u}_i^T\mathbf{e}\right)\mathbf{v}_i, \tag{2.6}$$

where $\mathbf{g}_{true} = \mathbf{H}\mathbf{f}$ is the noiseless data, the ith columns $\mathbf{v}_i, \mathbf{u}_i$ of $\mathbf{V},\mathbf{U}$ are the singular vectors corresponding to the ith singular value σ_i, and

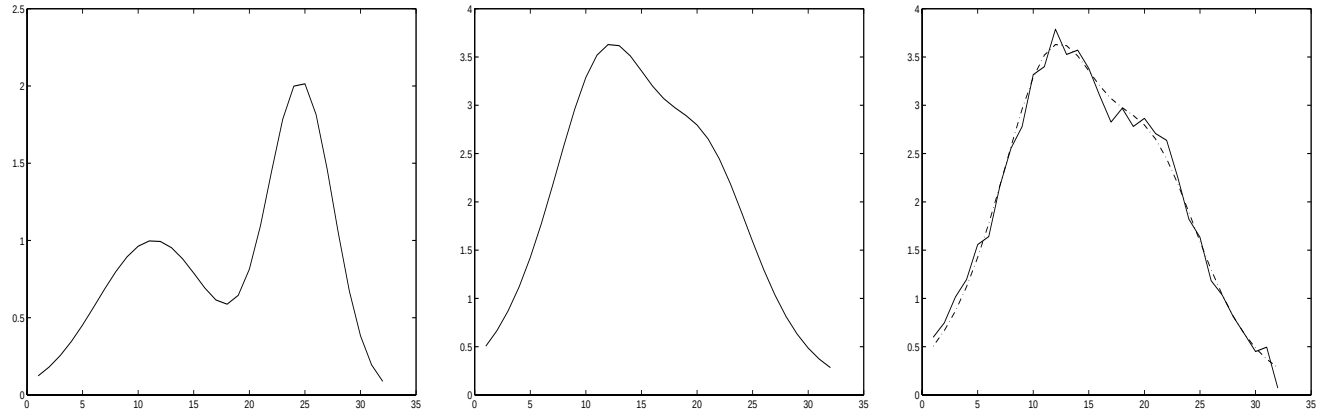
$$\mathbf{\Sigma}^\dagger = \text{diag}\left(\sigma_i^\dagger\right), \quad \sigma_i^\dagger = \begin{cases} \frac{1}{\sigma_i} & \text{if } \sigma_i \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.7}$$

We assume that $\mathbf{g}_{true}$ satisfies the discrete Picard condition [55]; that is the generalized Fourier coefficients $|\mathbf{u}_i^T \mathbf{g}_{true}|$ of g_{true} decay on the average faster than the singular values σ_i. As a result of the white noise assumption, the generalized Fourier coefficients of the noise $|\mathbf{u}_i^T \mathbf{e}|$ are roughly constant for all i with expected value

$$E\left\{|\mathbf{u}_i^T \mathbf{e}|\right\} = \frac{\sigma_e}{n^{1/2}}, \quad i = 1, \ldots, m \tag{2.8}$$

As a consequence, the generalized Fourier coefficients of the perturbed right hand side $|\mathbf{u}_i^T \mathbf{g}|$ level off at approximately $\frac{\sigma_e}{m^{1/2}}$ even if the noiseless right hand side $\mathbf{g}_{true}$ satisfies the discrete Picard condition because these generalized Fourier coefficients are dominated by $|\mathbf{u}_i^T \mathbf{e}|$ for large i. Ill-conditioned matrices are characterized by the presence of very small singular values σ_i. It is clear from the representation (2.6) that the least-squares solution $\mathbf{f}^\dagger$ is dominated by the $|\mathbf{u}_i^T \mathbf{e}|$ corresponding to the small σ_i since the generalized Fourier coefficients of the noise level off according to (2.8) while $|\mathbf{u}_i^T \mathbf{g}_{true}|$ is insignificant compared to the noise because of the discrete Picard condition. Therefore, the minimum norm least squares solution is useless for problems with tiny but nonzero singular values.

In order to demonstrate the adverse effects of the noise and the ill-conditioning of the system matrix on the least-squares solution $\mathbf{f}^\dagger$, we generated a test problem where the 32×32 system of linear equations, $\mathbf{g}_{true} = \mathbf{H}\mathbf{f}$, was obtained by invoking the `shaw(32)` command in Hansen's regularization toolbox [57] in MATLAB. The exact right hand side is modified by adding zero mean Gaussian distributed noise, $\mathbf{e}$, to $\mathbf{g}_{true}$ so that SNR $= \frac{\text{variance}(\mathbf{g}_{true})}{\sigma_e^2} = 10^2$ (we will sometimes express the SNR in dB in which case SNR is defined as $10 \log\left(\frac{\text{variance}(\mathbf{g}_{true})}{\sigma_e^2}\right)$). Figure 2.1 (a)-(c) show the original object, the data, and the



(a) (b) (c)

Figure 2.1: (a) The original object $\mathbf{f}$. (b) The noiseless data $\mathbf{g}_{true}$. (c) Noisy data (solid line) $\mathbf{g}$ super-imposed on $\mathbf{g}_{true}$ (broken line). $SNR = 10^2$.

perturbed data, respectively. We then computed the least-squares solutions $\mathbf{f}^{\dagger}$ corresponding to the exact $\mathbf{g}_{true}$ and noisy right hand sides $\mathbf{g}$ as displayed in Fig. 2.2 (a) and (b). As can seen in Fig. 2.2 (a), most of the generalized Fourier coefficients of the unperturbed problem $|\mathbf{u}_i^T \mathbf{g}_{true}|$ satisfy the discrete Picard condition although eventually, for large i, both the singular values and the Fourier coefficients become dominated by rounding errors. For the noisy problem in Fig. 2.2 (b) we see that the Fourier coefficients for the noisy right hand side become dominated by the perturbation errors for i much smaller than it is in Fig. 2.2 (a). We also see from Fig. 2.2 (b) that the Fourier coefficients of the perturbed system still satisfy the discrete Picard condition for small i. To have an acceptable solution we should dampen out the components for which the perturbation dominates and leave the rest of the components for small i intact. This objective can be achieved through the use of a *regularization procedure* [8,9,44,91] where a unique and stable estimate $\mathbf{f}^*$ is sought by incorporating prior information on the original object. This has the effect of replacing the original ill-conditioned

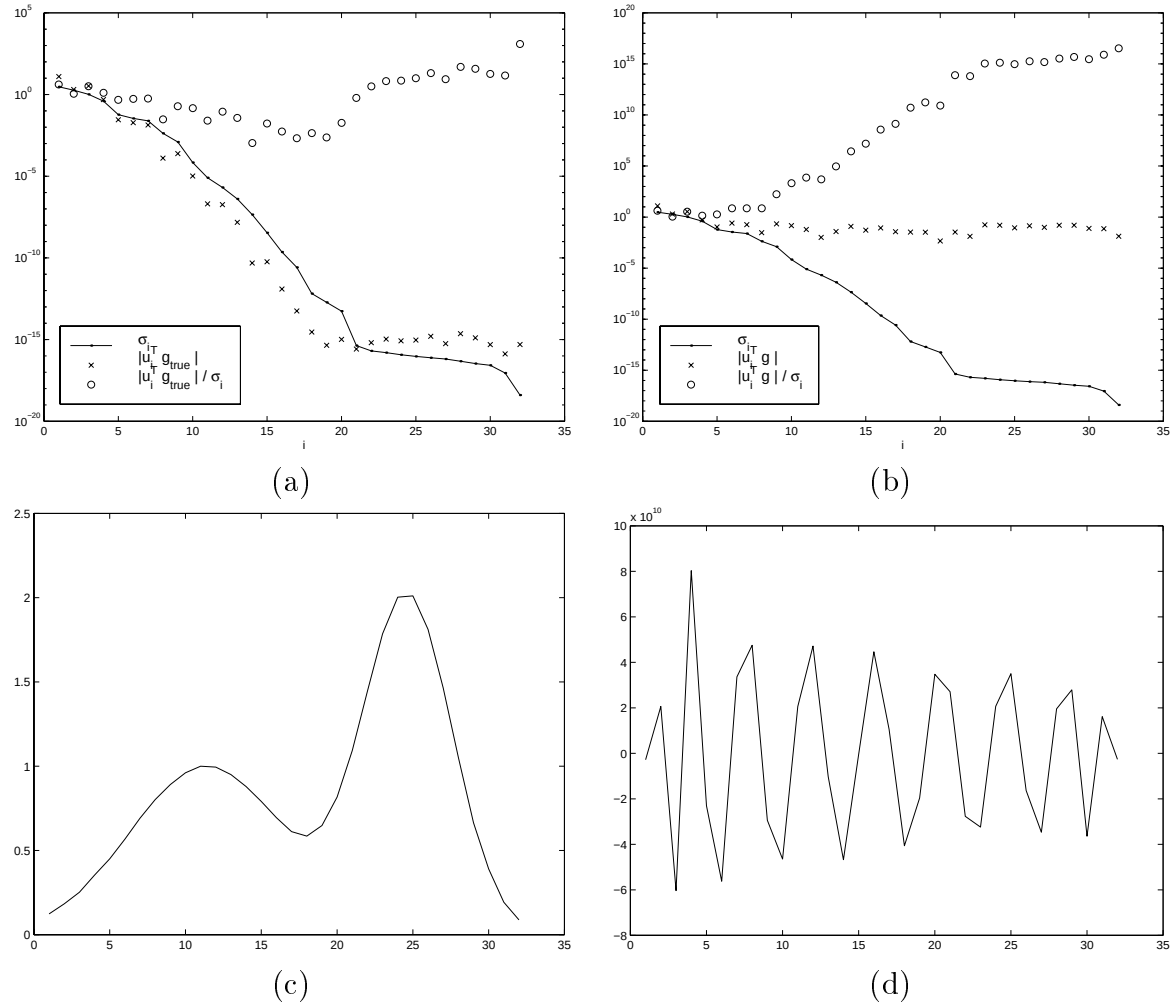


Figure 2.2: (a) Picard plot for exact right hand side (r.h.s.) $\mathbf{g}_{true}$. (b) Picard plot for noisy r.h.s. $\mathbf{g}$. (c) Least squares solution $\mathbf{f}^{\dagger}$ obtained from the exact right hand side $\mathbf{g}_{true}$. (d) Least squares solution $\mathbf{f}^{\dagger}$ obtained from the noisy right hand side $\mathbf{g}$.

problem with a well-conditioned one whose solution approximates that of the original.

2.1.2 Tikhonov Regularization

The simplest and the most well-known regularization method is that of Tikhonov [91] which consists of replacing the problem (2.3) with the problem of finding a solution to the following

$$\min_{\mathbf{f}} \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \lambda \|\mathbf{R}\mathbf{f}\|_2^2 \ , \tag{2.9}$$

where $\lambda > 0$ is a *regularization parameter* and $\mathbf{R}$ is a *regularization operator*, usually chosen to be the identity matrix or a discrete approximation of a derivative operator. For example, for a 1-D object $\mathbf{f}$, $\mathbf{R}$ can be $\mathbf{D}^{(1)}$ or $\mathbf{D}^{(2)}$ where

$$\mathbf{D}^{(1)} = \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{bmatrix} \in \mathcal{R}^{(n-1)\times n} \tag{2.10}$$

and

$$\mathbf{D}^{(2)} = \begin{bmatrix} 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix} \in \mathcal{R}^{(n-2)\times n} \tag{2.11}$$

are approximations to the first and second derivative operators.

The solution of (2.9) is

$$\mathbf{f}^*(\lambda) = \left(\mathbf{H}^T\mathbf{H} + \lambda \mathbf{R}^T\mathbf{R}\right)^{-1} \mathbf{H}^T\mathbf{g} \ . \tag{2.12}$$

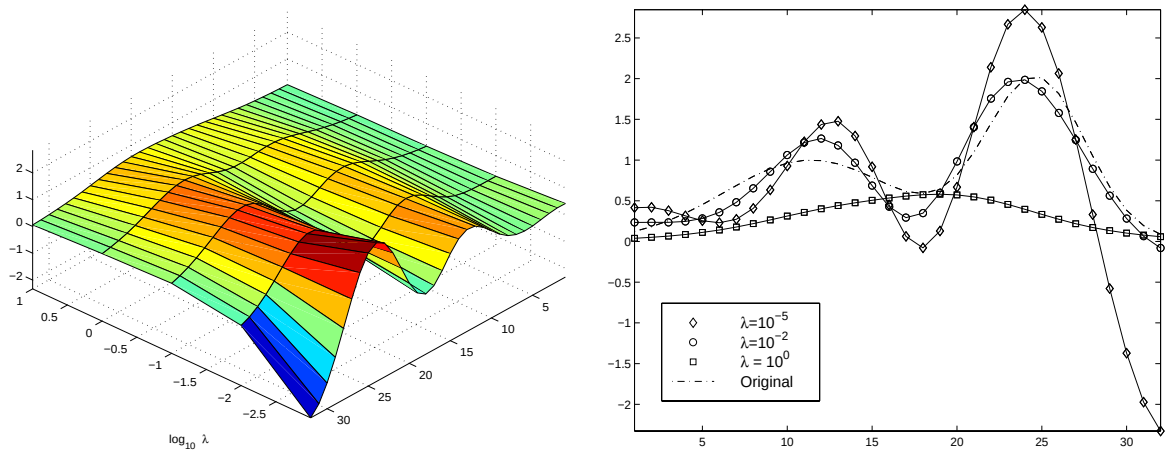
For the time being let us assume $\mathbf{R} = \mathbf{I}$ and write (2.12) in terms of the SVD of $\mathbf{H}$

$$\mathbf{f}^*(\lambda) = \sum_{i=1}^{n} \left(\frac{\sigma_i}{\sigma_i^2 + \lambda} \mathbf{u}_i^T\mathbf{g} + \frac{\sigma_i}{\sigma_i^2 + \lambda} \mathbf{u}_i^T\mathbf{e} \right) \mathbf{v}_i \ . \tag{2.13}$$

In light of the SVD analysis of the least-squares solution carried out in the previous paragraph, it is easy to see that the purpose of the regularization is to dampen or filter out the contributions to the solution coming from the small singular values. In (2.13) the contribution of the noise components $\mathbf{u}_i^T\mathbf{e}$ due to singular values $\sigma_i < \lambda$ is much less than it is in (2.6). Therefore $\mathbf{f}^*(\lambda)$ contains less distortion than $\mathbf{f}^\dagger$ and can be closer to the original noise-free solution $\mathbf{f}$. As the regularization parameter λ increases, the effect of the noise in the regularized solution becomes less evident. Note, however, that as λ increases we move further away from the original problem and may run the risk of having an over regularized solution.

In Fig. 2.3 (a) and (b) we display the regularized solutions corresponding to the `shaw(32)` problem. We used the the Tikhonov method with identity as the regularization operator. In Fig. 2.3 (a) we display the regularized solutions as a function of the regularization parameter and Fig. 2.3 (b) we plot the regularized solutions for $\lambda = 10^{-5}, 10^{-2}, 10^{0}$. Note that the regularized solutions become more oscillatory as the regularization parameter decreases and the regularized solutions corresponding to the smaller values of λ clearly show the effects of noise. On the other hand for large λ the solution becomes smoother and can no longer follow the variations present in the original object. From Fig. 2.3 (b) it is seen that among the cases illustrated, the regularized solution is closest to the original object for $\lambda = 10^{-2} = \frac{1}{\mathrm{SNR}}$.

If $\mathbf{R} \neq \mathbf{I}$ the solution $\mathbf{f}^*(\lambda)$ can no longer be expressed in terms of the SVD of $\mathbf{H}$. In such a case, one must use the Generalized Singular Value Decomposition (GSVD) of the matrix pair $(\mathbf{H}, \mathbf{R})$ [49, 59]. Assuming $\mathbf{H} \in \mathcal{R}^{m\times n}$ and $\mathbf{R} \in \mathcal{R}^{p\times n}$ and $m \geq n \geq p$, the GSVD of



(a) (b)

Figure 2.3: (a) Variation of the regularized solution with respect to the regularization parameter. (b) Selected regularized solutions and the original object.

$(\mathbf{H}, \mathbf{R})$ is given by

$$\mathbf{H} = \tilde{\mathbf{U}} \begin{bmatrix} \tilde{\mathbf{\Sigma}} & 0 \\ 0 & \mathbf{I}_{n-p} \end{bmatrix} \mathbf{X}^{-1}, \quad \mathbf{R} = \tilde{\mathbf{V}} \begin{bmatrix} \mathbf{B} & 0 \end{bmatrix} \mathbf{X}^{-1}, \tag{2.14}$$

where columns of $\tilde{\mathbf{U}} \in \mathcal{R}^{m \times n}$ and $\tilde{\mathbf{V}} \in \mathcal{R}^{p \times p}$ are orthonormal, $\mathbf{X} \in \mathcal{R}^{n \times n}$ is nonsingular and $\tilde{\mathbf{\Sigma}}$ and $\mathbf{B}$ are $p \times p$ diagonal matrices such that

$$\tilde{\mathbf{\Sigma}} = \mathrm{diag}(\tilde{\sigma_1}, \ldots, \tilde{\sigma_p}), \qquad \mathbf{B} = \mathrm{diag}(\mu_1, \ldots, \mu_p). \tag{2.15}$$

Furthermore, the diagonal entries of $\tilde{\mathbf{\Sigma}}$ and $\mathbf{B}$ are non-negative and ordered such that

$$0 \leq \tilde{\sigma_1} \leq \ldots \leq \tilde{\sigma_p} \leq 1, \quad 1 \geq \mu_1 \geq \ldots \geq \mu_p \geq 0, \tag{2.16}$$

and they are normalized such that

$$\tilde{\sigma_i}^2 + \mu_i^2 = 1, \quad i = 1, \ldots, p\ . \tag{2.17}$$

The generalized singular values of $(\mathbf{H}, \mathbf{R})$ are defined as the ratios

$$\gamma_i = \frac{\tilde{\sigma}_i}{\mu_i}, \quad \gamma_1 \geq \ldots \geq \gamma_p \geq 0 \quad i = 1, \ldots, p. \tag{2.18}$$

Note however that the matrices $\tilde{\mathbf{U}}$, $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{\Sigma}}$ are different from the matrices appearing in the ordinary SVD of the matrix $\mathbf{H}$. Using the GSVD of $(\mathbf{H}, \mathbf{R})$, the regularized solution can be written as

$$\mathbf{f}^*(\lambda) = \sum_{i=1}^{n} \frac{\gamma_i^2}{\gamma_i^2 + \lambda} \frac{\tilde{\mathbf{u}}_i^T \mathbf{g}}{\tilde{\sigma}_i} \mathbf{x}_i \ , \tag{2.19}$$

where $\mathbf{x}_i$ is the ith column of $\mathbf{X}$. One noteworthy difference between (2.19) and (2.13) is that in the latter the solution space is spanned by the $\mathbf{v}_i$ while in the former the solution is a linear combination of the $\mathbf{x}_i$. The basic limitation of the Tikhonov regularization with identity is that while the left singular vectors $[\mathbf{v}_i]_{i=1,\ldots,n}$ form a good basis for the representation of the column space $\mathbf{H}$, they may not be suitable for recovering the original object $\mathbf{f}$. Thus if the same operator $\mathbf{H}$ occurs in two different inverse problems, the basis functions for the representation of the solution will be the same if $\mathbf{R} = \mathbf{I}$, even though the type of object to be recovered may be quite different in the two problems. Tikhonov regularization with an operator $\mathbf{R}$ other than the identity provides an effective means of changing the basis functions for the representation of $\mathbf{f}$. Unfortunately, there is little publicized work that provides guidance for choosing the regularization operator $\mathbf{R}$. It has been argued that an $\mathbf{R}$ matrix which leaves the vector $[1, \ldots, 1]^T$ unregularized is a superior choice over $\mathbf{I}$ [82].

Finally, it is useful to note that (2.12) is equivalent to the following problem

$$\min_{\mathbf{f}} \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 \ \text{ subject to } \ \|\mathbf{R}\mathbf{f}\|_2^2 < C \tag{2.20}$$

for the appropriate choice of the constant C. From this perspective, the Tikhonov method can be seen as a modification of the original least squares problem which restricts the solution to lie in a ball of radius C in a Hilbert space $\mathcal{P}$ with the norm $||\mathbf{f}||_{\mathcal{P}} = ||\mathbf{Rf}||_2$. This corresponds to incorporating prior information on the original object $\mathbf{f}$ that the object lies in $\mathcal{P}$ with a known bound on its norm. As we will see in the next section, the method of regularization can also be expressed in the statistical language as a Bayesian estimation technique [7]. It has been pointed out by several authors that many regularization schemes developed in the context of functional analysis have corresponding interpretations in terms of Bayesian estimation where a specific prior distribution is assumed on the original object [29, 50, 93, 94]. In the rest of this thesis, we will prefer using the Bayesian interpretation of the regularization theory in developing our IRR algorithms but we will also indicate the relation of the Bayesian estimation technique under consideration to its functional analytic counterpart where appropriate.

2.1.3 Statistical Interpretation of the Regularization

In the Bayesian image restoration method of interest here the prior information is quantified by specifying a probability density on $\mathbf{f}$. Then the prior information is combined with the information contained in $\mathbf{g}$ to produce an estimate of the unknown image. We assume here a linear, additive Gaussian noise model so that the probability density for $\mathbf{g}$ given $\mathbf{f}$ is

$$P(\mathbf{g}|\mathbf{f}, \sigma_e) = \frac{1}{(2\pi\sigma_e^2)^{m/2}} \exp\left\{-\frac{1}{2\sigma_e^2}\|\mathbf{g} - \mathbf{Hf}\|_2^2\right\}, \tag{2.21}$$

where σ_e^2 is the variance of the noise. If it so happens that the probability distribution for $\mathbf{f}$, which is dependent on a set of parameters $\boldsymbol{\theta}$, is in the form

$$P(\mathbf{f}|\boldsymbol{\theta}) \propto \exp\{-\Phi(\mathbf{f}, \boldsymbol{\theta})\} \ , \tag{2.22}$$

then by Bayes's rule, the Maximum a Posteriori (MAP) estimate, $\mathbf{f}^*$, is obtained by minimizing the following log posterior density function with respect to $\mathbf{f}$ [3, 7]

$$J(\mathbf{f}, \sigma_e, \boldsymbol{\theta}) = \frac{1}{2\sigma_e^2}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \Phi(\mathbf{f}, \boldsymbol{\theta}). \tag{2.23}$$

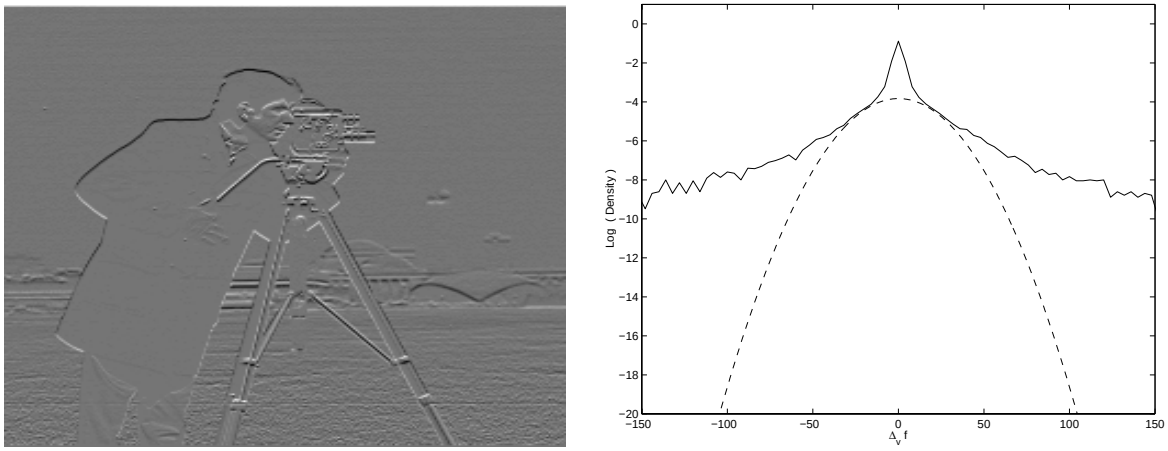
The function $\Phi(\mathbf{f}, \boldsymbol{\theta})$, called the *energy* function in the context of Bayesian estimation, is the energy attributed to the image $\mathbf{f}$, and $\boldsymbol{\theta}$ is the vector of possibly unknown model parameters. We give low energy to the images which coincide with our prior conceptions and high energy to those which do not. Thus, if our prior belief about the true image is that it is smooth, then the energy is a measure of the *roughness*.

One of the most commonly used energy functions has the form

$$\Phi(\mathbf{f}, \theta) = \frac{1}{2\theta^2}\|\mathbf{R}\mathbf{f}\|_2^2 \ , \tag{2.24}$$

where $\mathbf{R}$ is a differential operator. In stochastic terms, this choice implies that $\mathbf{R}\mathbf{f}$ is a vector of zero mean independent identically distributed Gaussian random variables with variance θ^2. Substituting $\Phi(\mathbf{f}, \theta)$ in (2.24) into (2.23) we obtain the cost function for the constrained least squares (CLS) restoration

$$J(\mathbf{f}, \sigma, \theta) = \frac{1}{2\sigma_e^2}\|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \frac{1}{2\theta^2}\|\mathbf{R}\mathbf{f}\|_2^2 \ , \tag{2.25}$$



(a) (b)

Figure 2.4: (a) Vertical difference image of the Cameraman. (b) Distribution of the vertical differences, $\nabla_v \mathbf{f}$, of the Cameraman image. Broken line depicts a Gaussian distribution with the same mean and variance.

which yields the estimate

$$\mathbf{f}^*(\lambda) = \left(\mathbf{H}^T\mathbf{H} + \lambda \mathbf{R}^T\mathbf{R}\right)^{-1} \mathbf{H}^T\mathbf{g}\,, \tag{2.26}$$

where $\lambda = \frac{\sigma_\epsilon^2}{\theta^2}$ is the regularization parameter. Note that (2.26) is exactly the same as (2.12), thereby establishing the connection between statistical and the functional analytic interpretations of the regularization. The criterion (2.25) contains two terms; the first, $\|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2$, expresses the fidelity to the available data $\mathbf{g}$ and the second, $\|\mathbf{R}\mathbf{f}\|_2^2$, the smoothness of the estimate. Therefore, the regularization parameter λ represents the trade-off between fidelity to the data and smoothness of the estimate $\mathbf{f}^*(\lambda)$.

Looking at (2.23), we can immediately identify two issues of primary importance associated with the MAP estimation procedure. The first issue is the determination of an appropriate prior model, $\Phi(\mathbf{f}, \boldsymbol{\theta})$. The CLS image restoration scheme outlined above assumes that the gradient of the image follows a Gaussian distribution. This assumption, however, is not

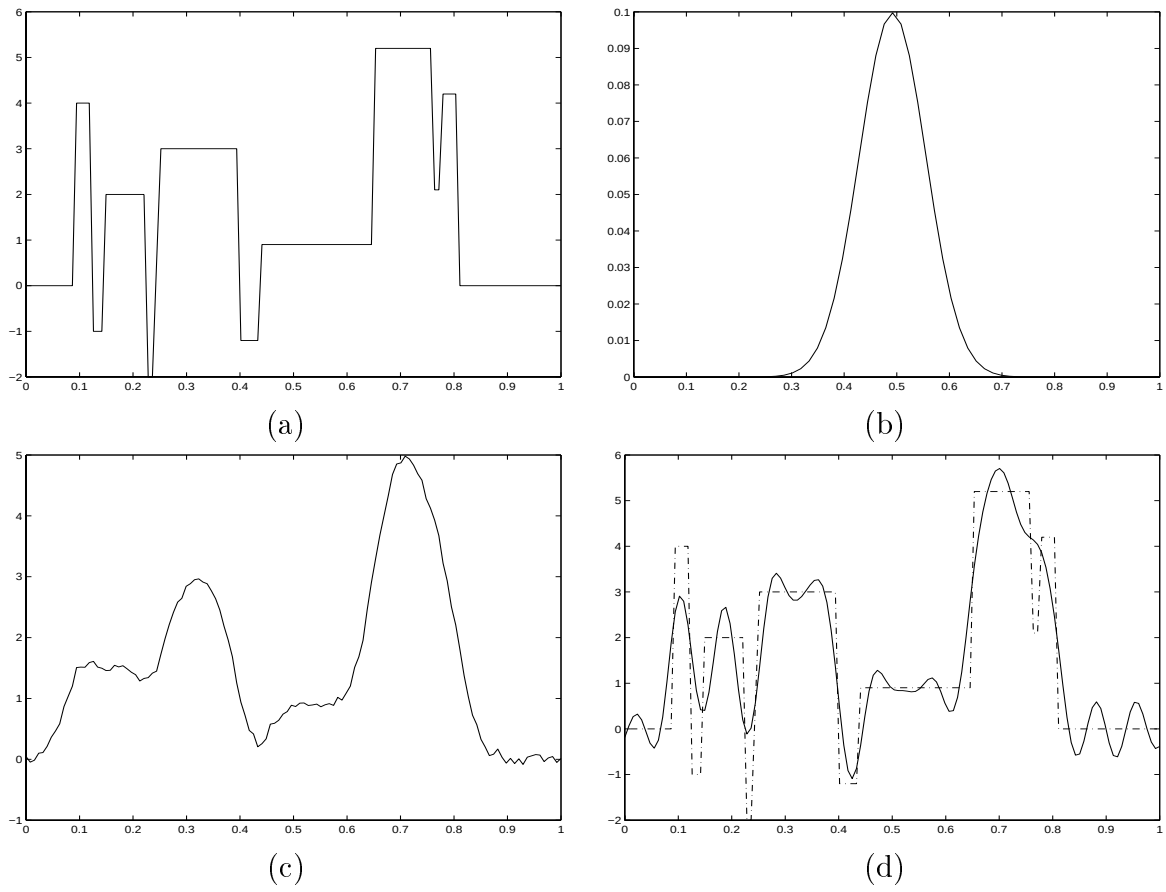


Figure 2.5: (a) Original Blocks image. (b) Degradation kernel. (c) Degraded Blocks. (d) The CLS restoration with $\mathbf{R} = \mathbf{D}^{(1)}$.

true in general since images typically consist of smooth areas separated by occasional edges. This empirical observation implies that the gradient of the image should follow a distribution which is concentrated around zero due to the contribution of smooth areas with heavy-tails representing the contribution of the edges. Figure 2.4(b) shows the histogram of the vertical differences of the Cameraman image. For comparison, the mean and the variance of the vertical difference image were computed, and a Gaussian pdf with the same mean and variance was plotted on top of the histogram of the vertical difference image. As observed from

Fig. 2.4(b), the distribution of the vertical difference image is notably non-Gaussian. The CLS image restoration scheme has been criticized on the grounds that it applies too much penalty to the edges in the image and produces an over-smooth restoration. An example of this phenomena is provided in Fig. 2.5. We generated a test problem by degrading the 128 point *Blocks* signal (Fig. 2.5 (a)) extracted from the Donoho's Wavelab Toolbox [20] in MATLAB. The blurring kernel was a Gaussian function $h(t) = \frac{1}{\sqrt{\pi\sigma_h^2}} \exp\left\{-\frac{t^2}{2\sigma_h^2}\right\}$ with $\sigma_h = 4.0$ as seen in Fig. 2.5 (b). We will use this test problem frequently in subsequent sections and refer to the problem as simply *Blocks*. We added zero mean white Gaussian noise to the degraded Blocks so that SNR $= 30dB$. The degraded noisy signal is shown in Fig. 2.5 (c). We restored the degraded signal by CLS using $\mathbf{R} = \mathbf{D}^{(1)}$ and the best regularization parameter minimizing the l_2 norm of the error between the original and restored signals. The resulting restoration is shown in Fig. 2.5 (d). As observed, the edge information present in the original signal is completely lost and edges are replaced by smooth transitions. Therefore, it is of great interest to develop so-called *edge-preserving* prior models for the purposes of restoration.

After specifying an appropriate prior, the second important issue is the determination of the regularization parameter(s) which would yield a good compromise between the fidelity to the data and smoothness of the solution. Note that in the CLS restoration scheme the regularization parameter is defined as $\lambda = \frac{\sigma_\epsilon^2}{\theta^2}$. Therefore, the MAP estimation procedure described here provides an answer to the regularization parameter selection problem. If the

σ_e^2 and θ^2 are known we immediately obtain the regularization parameter upon substitution. However, such prior information is rarely available. Therefore, there arises a need for techniques for determining the regularization parameter(s) which do not assume any side information on signal or noise variance.

2.1.4 Half Quadratic Regularization

The regularization approach in (2.23) can be trivially extended to include multiple constraints. The reason for using multiple constraints is the desire to incorporate more *a priori* information on the object to be restored/reconstructed in the hope of obtaining a better solution. In some cases, multiple constraints are necessary to obtain a useful solution (see [15], for example). We extend the regularization scheme in (2.23) in the following way:

$$\min_{\mathbf{f}} \left\{ \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \sum_{i=1}^{M} \lambda_i \Phi_i(\mathbf{R}_i\mathbf{f}) \right\}, \quad \mathbf{R}_i \in \mathcal{R}^{m\times n} \ , \tag{2.27}$$

where $\mathbf{R}_i$, $i = 1, \ldots, M$ are regularization operators and λ_i are the corresponding regularization parameters, $\Phi_i(\mathbf{R}_i\mathbf{f}) = \sum_{j=1}^{m} \phi_i([\mathbf{R}_i\mathbf{f}]_j)$ with $\phi_i(t)$ being a function such as $\phi_i(t) = t^2$ and the notation $[\mathbf{R}_i\mathbf{f}]_j$ denotes the jth element of the vector $\mathbf{R}_i\mathbf{f}$. In order to find first order conditions that must be satisfied by the solution $\mathbf{f}^*(\boldsymbol{\lambda})$, we take the gradient of (2.27) with respect to $\mathbf{f}$ and set the result equal to zero to get

$$\mathbf{K}_{\mathbf{f}^*}\mathbf{f}^* = \mathbf{H}^T\mathbf{g}, \tag{2.28}$$

where

$$\mathbf{K}_{\mathbf{f}^*} = \mathbf{H}^T\mathbf{H} + \frac{1}{2}\sum_{i=1}^{M} \lambda_i \mathbf{R}_i^T \mathrm{diag}_{k=1,\ldots,m}\left[\frac{\phi_i'([\mathbf{R}_i\mathbf{f}^*]_k)}{[\mathbf{R}_i\mathbf{f}^*]_k}\right]\mathbf{R}_i \ . \tag{2.29}$$

Except for a few special cases, minimizing (2.27) for arbitrary ϕ_i is a difficult task because (2.28) is nonlinear in $\mathbf{f}^*$. Note that the formula (2.28) defines $\mathbf{f}^*$ only implicitly since $\mathbf{f}^*$ appears in the expression for $\mathbf{K}_{\mathbf{f}^*}$ on the left hand side of (2.28. In order simplify the minimization task, Geman and Yang [46] introduced the concept *half-quadratic* regularization. In their own words, "The basic idea is to introduce a new objective function which, although defined over an extended domain, has the same minimum in $\mathbf{f}^*$ and can be manipulated with linear algebraic methods" . The method simply consists of solving (2.28) iteratively starting from an initial solution vector $\mathbf{f}^0$

$$\mathbf{K}_{\mathbf{f}^j}\mathbf{f}^{j+1} = \mathbf{H}^T\mathbf{g}, \tag{2.30}$$

where $\mathbf{f}^j$ is the solution obtained at the jth iteration and

$$\mathbf{K}_{\mathbf{f}^j} = \mathbf{H}^T\mathbf{H} + \frac{1}{2}\sum_{i=1}^{M}\lambda_i\mathbf{R}_i^T \mathrm{diag}_{k=1,\ldots,m}\left[\frac{\phi_i'([\mathbf{R}_i\mathbf{f}^j]_k)}{[\mathbf{R}_i\mathbf{f}^j]_k}\right]\mathbf{R}_i\ . \tag{2.31}$$

(2.30) is iterated until $\frac{\|\mathbf{f}^{j+1}-\mathbf{f}^j\|}{\|\mathbf{f}^j\|} < tol$, where tol is a small positive constant (we use $tol = 10^{-5}$ in our examples). Note that when $\mathbf{f}^j$ is known, (2.30) is the solution to the following quadratic problem

$$\min_f \|\mathbf{g}-\mathbf{H}\mathbf{f}\|_2^2 + \sum_{i=1}^{M}\lambda_i\|[\mathbf{Q}_i^{(j)}]^{1/2}\mathbf{R}_i\mathbf{f}\|_2^2 \tag{2.32}$$

where $\mathbf{Q}_i^{(j)} = \mathrm{diag}_{k=1,\ldots,m}\left[\frac{\phi_i'([\mathbf{R}_i\mathbf{f}^j]_k)}{[\mathbf{R}_i\mathbf{f}^j]_k}\right]$. It is proven in [25] that if $\phi_i(t)$, $i = 1,\ldots,M$ satisfy the following conditions, (2.27) has a unique solution $\mathbf{f}^*$ and the iterates $\mathbf{f}^j$ produced by the half-quadratic algorithm in (2.30) converges to $\mathbf{f}^*$ as $k \to \infty$:

1. $\phi_i(t)$ is convex.

2. $\phi_i(t) \geq 0$, $\forall t$ with $\phi_i(0) = 0$.

3. $\phi_i(t) = \phi_i(-t)$.

4. $\phi_i(t)$ is continuously differentiable.

5. $\lim_{t \to 0^+} \frac{\phi_i'(t)}{t} = C$, $0 < C < \infty$.

6. $\phi_i'(t) \geq 0$, $\forall t$.

7. $\phi_i'''(0) = 0$.

8. $\phi_i^{(4)}(0)$ exits.

The last two items are technical assumptions used in the convergence proof. If the condition that $\phi_i(t)$ is convex is relaxed, the half-quadratic algorithm still converges but probably computes a local minimum of (2.27) [25].

2.2 Common Parameter Selection Methods

The problem of choosing a suitable regularization parameter which would yield an estimate as close to the original object as possible has received considerable attention in the past and is still an open problem. There are methods which rely on prior information such as a bound on the (semi)norm of the signal or the noise level. A well known example is the Morozov's discrepancy principle [79]. On the other hand, methods such as Generalized Cross Validation (GCV) [26] and the L-curve [56, 58] do not require any side information. We note that all of the methods mentioned above deal with choosing a *single* parameter.

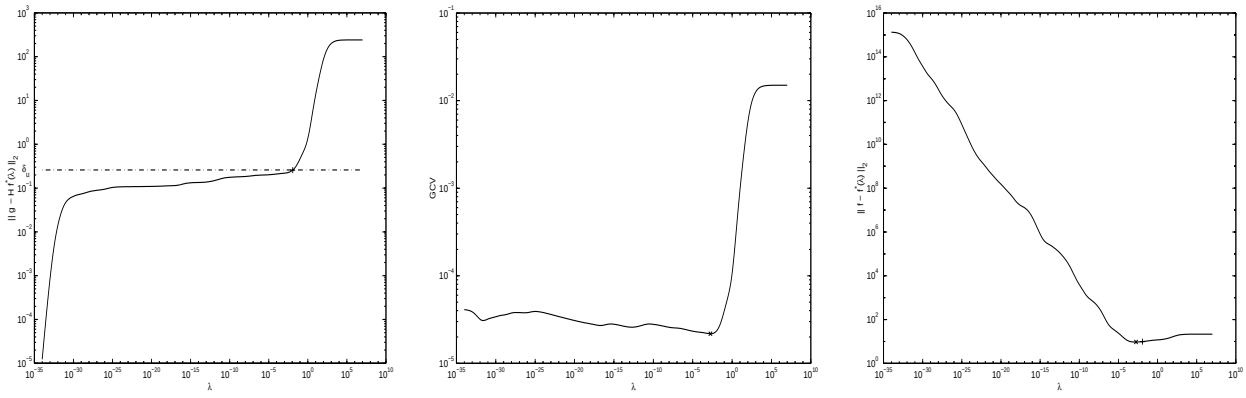


Figure 2.6: (a) Illustration of the discrepancy principle. '+' indicates the intersection of $\|\mathbf{g} - Hf^*(\lambda)\|$ with $\delta_{\mathbf{u}}$. (b) GCV function. 'x' indicates the minimum of the GCV function. (c) Corresponding estimation error plot.

2.2.1 The Discrepancy Principle

One method that has attained a widespread interest is the discrepancy principle, usually attributed to Morozov [79]. The idea is simply to select the regularization parameter λ so that the residual norm is equal to an *a priori* bound $\delta_{\mathbf{u}}$ for the norm of the errors $\mathbf{u}$ in the right hand side, i.e.,

$$\|\mathbf{g} - \mathbf{H}\mathbf{f}^*(\lambda)\| = \delta_{\mathbf{u}}, \quad \text{where } \|\mathbf{u}\| \leq \delta_{\mathbf{u}} \tag{2.33}$$

Methods for solving (2.33) with respect to λ can be found in [24]. Note that if the exact system is consistent, then as $\delta_{\mathbf{u}} \to 0$, $\mathbf{f}(\lambda) \to \mathbf{f}$. Figure 2.6(a) illustrates the discrepancy principle on the *Blocks* problem.

2.2.2 Generalized Cross Validation

The Generalized Cross Validation (GCV) [26, 99] method does not require any prior information on the noise variance or the norm of the object. GCV is based on statistical considerations. That is, a good value of the regularization parameter should predict missing data values. More precisely, the construction of cross validation criterion is argued as follows:

1. Construct an estimate $\mathbf{f}^*(\lambda;\ i)$ based on (2.25), but excluding the ith data point from the first term.

2. Evaluate the corresponding prediction error $p_i(\lambda) = g_i - [\mathbf{H}\mathbf{f}^*(\lambda;\ i)]_i$.

3. Repeat i) and ii) for $i = 1, \ldots, n$ and compute the cross validation function

$$C(\lambda) = \sum_{i=1}^{n} p_i^2(\lambda)\ . \tag{2.34}$$

4. Repeat i) - iii) for a variety of values for λ and choose λ_{CV} to minimize the cross validation function.

Golub *et. al.* [48] introduced the generalized the cross validation argument and proposed minimizing a generalized version of the ordinary cross validation function, the GCV function $G(\lambda)$. $C(\lambda)$ and $G(\lambda)$ do not differ greatly but the former has more pleasing mathematical properties. The formula of $G(\lambda)$ is given by

$$G(\lambda) = \frac{\|\mathbf{g} - \mathbf{H}\mathbf{f}^*(\lambda)\|_2^2}{\mathrm{trace}(\mathbf{I} - \mathbf{H}\mathbf{L}(\lambda))}\ , \tag{2.35}$$

where $\mathbf{L}(\lambda)$ is the matrix which maps the data to the regularized solution, i.e. $\mathbf{f}^*(\lambda) = \mathbf{L}(\lambda)\mathbf{g}$. In Tikhonov regularization, for example, $\mathbf{L}(\lambda)$ is given by

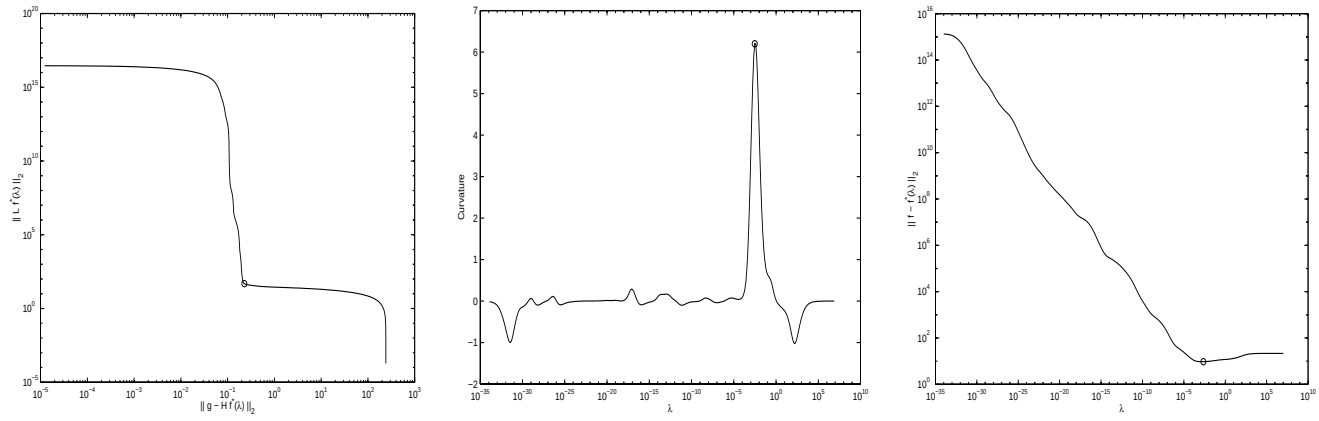
$$\mathbf{L}(\lambda) = (\mathbf{H}^T\mathbf{H} + \lambda\mathbf{R}^T\mathbf{R})^{-1}\mathbf{H}^T \ . \tag{2.36}$$

Figure 2.6(b) shows the GCV function for the *Blocks* experiment. The GCV method has proven its usefulness in numerous applications and has desirable theoretical properties [72]. However, two difficulties are associated with this method: the minimum of the GCV function is often very flat and, therefore, difficult to locate numerically, and the method may fail to compute the correct λ when the errors are highly correlated [56].

2.2.3 The L-curve Method

A convenient tool for choosing a single regularization parameter which does not require any side information is the L-curve method first introduced by Lawson and Hanson [73] and popularized by Hansen [56, 58] *et. al.*. The L-curve is simply a logarithmic plot of residual norm $\|\mathbf{g} - \mathbf{H}\mathbf{f}^*(\lambda)\|_2^2$ (the first term on the right hand side (r.h.s.) of (2.23)) versus the log of the reconstruction (semi)norm (second term on the r.h.s. of (2.23)) for a set of admissible regularization parameters. In this way, the L-curve displays the compromise between the minimization of these two quantities. It has been argued and numerically shown that the so called "corner" of the L-curve, defined as the point with maximum curvature, corresponds to a point where regularization and perturbation errors are balanced [56].

In Fig. 2.7 we plotted a typical L-curve along with the error between the original and restored signals for a range of regularization parameters. The experiment for which the



(a) (b) (c)

Figure 2.7: (a) A typical L-curve, (b) its curvature and (c) the error between original and restored signals. Circle indicates the point with maximum curvature. $\mathbf{f}^*(\alpha)$ is the estimate of $\mathbf{f}$ using α as the regularization parameter.

L-curve was computed was the *Blocks* problem solved with Tikhonov regularization with identity [91]. The circle indicates the corner of the L-curve defined to be the point on the curve $(z(\lambda), x(\lambda)) = (\log ||\mathbf{g} - \mathbf{H}\mathbf{f}^*(\lambda)||_2, \log ||\mathbf{R}\mathbf{f}||_2)$ with maximum curvature $\kappa(\lambda)$ [32, 81]

$$\kappa(\lambda) = \frac{z'x'' - x'z''}{\left[(x')^2 + (z')^2\right]^{3/2}} \tag{2.37}$$

where differentiation is with respect to λ. The part of the L-curve to the left of the corner contains a region where the regularization parameter is getting smaller and the error between the original and reconstructed signals is dominated by the perturbation errors. The solution (semi)norm is very sensitive to small changes in the regularization parameter indicating a noisy solution. On the other hand, the part of the L-curve to the right of the corner is a region where the regularization parameter is gradually increasing and the residual norm is the most sensitive to the changes in the regularization parameter. In this region, the

restored signal is excessively smooth. These two regions are clearly separated by the corner point, hence the corner corresponds to a point where regularization and perturbation errors are approximately balanced. Supporting this observation, the corner of the L-curve in this example indicates a point which is very close to the point where the error between the original and restored signals is a minimum (Fig. 2.7(c)).

The points to the left and right of the corner where the L-curve bands towards the abscissa axis are called the *false corners* or *knees* of the L-curve. The regularization parameter λ is either too small (at the knee to the left of the corner) or too large (at the knee to the right of the corner). Observe from Fig. 2.7(b) that the curvature function has local minima at the false corners.

2.3 Orthonormal Wavelet Transform

In this thesis, we adopt a transform domain approach to the image restoration problem. Before introducing our image model, we briefly review the wavelet theory [28, 75, 76]. The fundamental idea behind the discrete wavelet transform is to decompose a signal into a sequence of increasingly "coarser" representations while at the same time retaining the information lost in moving from a fine scale to a coarser scale. Following the literature on orthonormal wavelets, elements of the 1-D signal $\mathbf{y} = [y_1, y_2, \ldots, y_n]^T$ are called the finest scale scaling coefficients and are denoted by $\mathbf{y}_J^{(0)}$ with $n = 2^J$. Beginning with $\mathbf{y}_J^{(0)}$, a lower resolution representation for $\mathbf{y}_J^{(0)}$ is obtained by first passing $\mathbf{y}_J^{(0)}$ through a low pass filter l and then decimating the output by a factor of two. Thus, $\mathbf{y}_{J-1}^{(0)}$ is coarser than $\mathbf{y}_J^{(0)}$ in

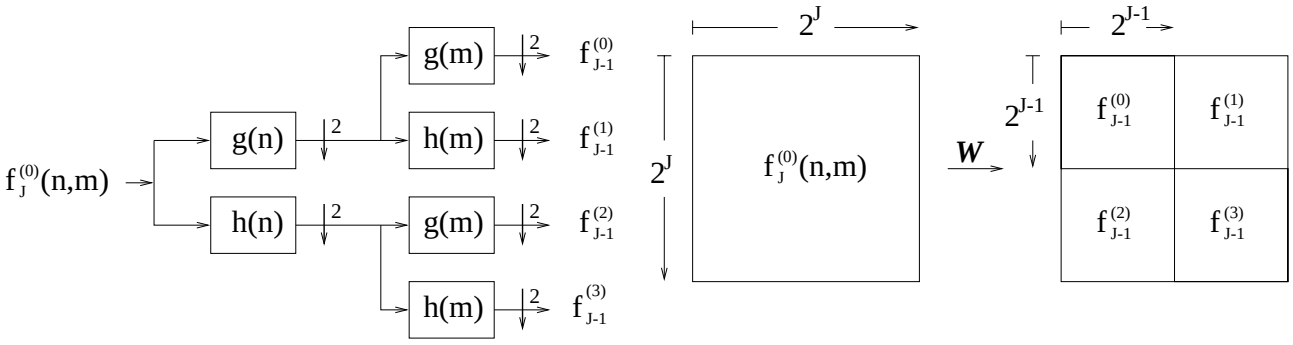


Figure 2.8: Wavelet decomposition of an image.

that the filtering/down-sampling operation removes the high frequency structure from the original signal and $\mathbf{y}_{J-1}^{(0)}$ is half as long as $\mathbf{y}_{J}^{(0)}$. The detail lost in moving from $\mathbf{y}_{J}^{(0)}$ to $\mathbf{y}_{J-1}^{(0)}$ is separately extracted by a high pass filter h followed by down sampling by two. The resulting vector of wavelet coefficients is denoted by $\mathbf{y}_{J-1}$ The filtering/down sampling operation can be repeated recursively on the scaling coefficients to obtain a multi-level wavelet decomposition of $\mathbf{y}$

$$\hat{\mathbf{y}} = [\mathbf{y}_{j_0}^{(0)T} | \mathbf{y}_{j_0}^{T} | \dots | \mathbf{y}_{J-1}^{T}]^T \ , \tag{2.38}$$

where j_0 is the coarsest scale at which $\mathbf{y}$ is represented and $\mathbf{y}_{j_0}^{(0)}$ denotes the vector of scaling coefficients at scale j_0 and the vectors $\mathbf{y}_j$, $j = j_0, \dots, J-1$, contain the wavelet coefficients at different scales. In effect, the wavelet transform can be represented as an operator taking the discrete signal $\mathbf{y}$ into its wavelet transform domain representation through matrix multiplication $\hat{\mathbf{y}} = \mathcal{W}\mathbf{y}$. Since the transform is orthonormal it is self inverting, i.e $\mathbf{y} = \mathcal{W}^T\hat{\mathbf{y}}$.

Figure 2.9 schematically depicts the three level wavelet decomposition of an image. As observed, the wavelet decomposition produces three scales consisting of the wavelet coefficients

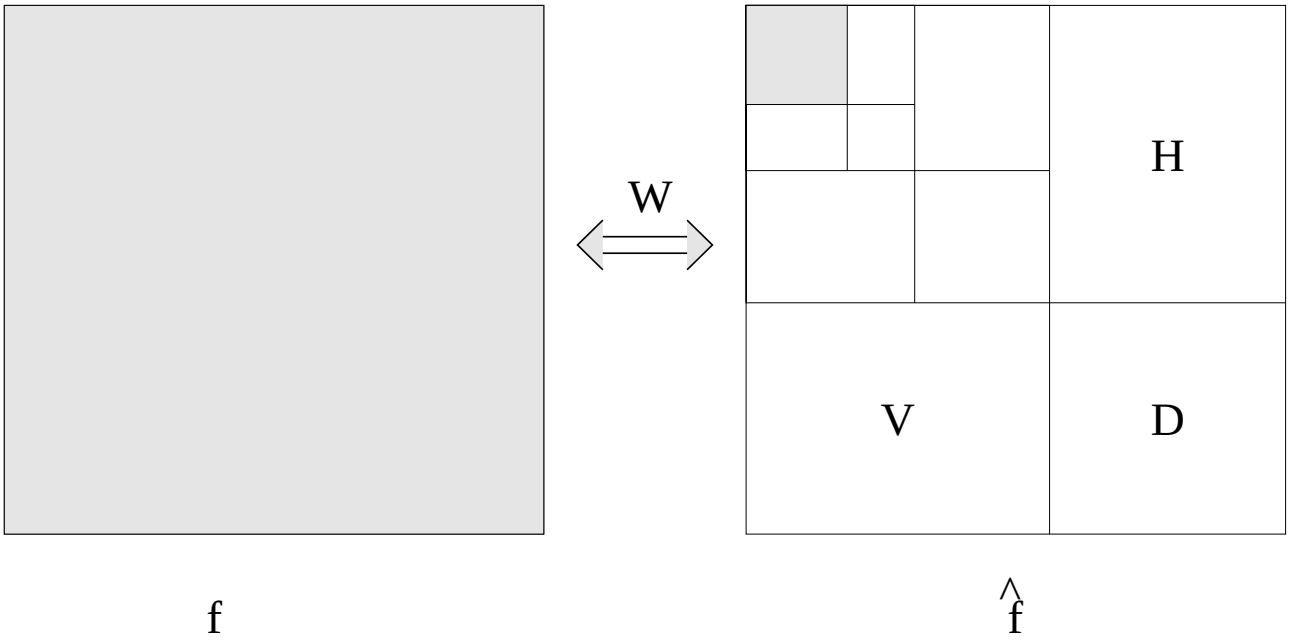


Figure 2.9: 3-level wavelet decomposition of an image.

representing the horizontal (indicated with H), vertical (indicated with V), and diagonal (indicated with D) details in the image at each scale. On the upper left corner we have the scaling coefficients which represent a low resolution rendition of the original image f.

2.3.1 Wavelet Representation of Image Restoration Problem

It is possible to obtain the wavelet transform of higher dimensional signals through a separable representation. If l and h are the discrete low-pass and high-pass filters associated with a particular 1-D wavelet transform then the discrete high pass filters $\{h(n)l(m),\ l(n)h(m),\ h(n)h(m)\}$ together with the low pass filter $l^{(0)}(n,m) = l(n)l(m)$ can be used to form the wavelet decomposition of $f(n,m)$. This decomposition can be implemented by 1-D filtering of rows and columns of images. In Fig. 2.8, we have schematically illustrated a 1-level

wavelet decomposition of an image $f(n,m)$ with $f_J^{(0)}(n,m)$ denoting the finest scale scaling coefficients. The 1-level wavelet decomposition of the image $f_J^{(0)}(n,m)$ produces four sub-images of size $2^{J-1} \times 2^{J-1}$, $f_{J-1}^{(k)}, \quad k = 0, \ldots, 3$. $f_{J-1}^{(0)}$ represents the scaling coefficients at scale $J-1$ and $f_{J-1}^{(k)}, \quad k = 1, \ldots, 3$ are the wavelet coefficients at scale $J-1$ corresponding to the vertical, horizontal and diagonal orientations in the image plane. Multi-level wavelet decompositions of the image $f(n,m)$ can be obtained by applying the 1-level wavelet decomposition scheme, outlined above, recursively to the scaling coefficients $f_{J-1}^{(0)}(n,m)$. We will use $\mathbf{f}_j^{(i)}$ to denote the vector of wavelet(scaling) coefficients obtained by lexicographically ordering the elements of the 2-D array $f_j^{(i)}(m,n)$ and $\hat{\mathbf{f}}$ to denote a lexicographically ordered version of all wavelet coefficients $\hat{f}(n,m)$.

With the conventions above, we can represent the problem in (2.2) in the wavelet domain as

$$\begin{aligned} \mathcal{W}\mathbf{g} &= \left(\mathcal{W}\mathbf{H}\mathcal{W}^T\right)\mathcal{W}\mathbf{f} + \mathcal{W}\mathbf{e} \\ \hat{\mathbf{g}} &= \hat{\mathbf{H}}\hat{\mathbf{f}} + \hat{\mathbf{e}}, \end{aligned} \tag{2.39}$$

where $\mathcal{W}$ is the 2-D wavelet transform matrix, $\hat{\mathbf{g}}$, $\hat{\mathbf{f}}$ and $\hat{\mathbf{e}}$ are the vectors holding the scaling and wavelet coefficients of the data, the original image, and the disturbance, $\hat{\mathbf{H}}$ is the wavelet domain representation of our linear degradation operator $\mathbf{H}$, and $\mathcal{W}^T\mathcal{W} = \mathbf{I}$ follows from the orthogonality of the wavelet transform. Note that since the wavelet transform is orthonormal $\hat{\mathbf{e}}$ is again Gaussian with zero mean and variance σ_e^2.

Chapter 3

A Multiscale Image Restoration Algorithm

In this section, we consider a statistically based, wavelet-domain approach to edge-enhanced image restoration in which we employ a stochastic interpretation of the regularization process [19,23,87]. We note that almost all of the work to date on wavelet-based, statistical regularization methods has concentrated on the use of multi-scale smoothness priors [4,77,78,100]. While Wang *et. al.* did consider issues of edge preservation in [100], their method was based on the processing of the output of an edge detector applied to the noisy data to alter the degree of regularization in a multiscale smoothness constraint. As described below and in subsequent sections, our approach is significantly different as the edge preservation is built directly into the regularization scheme itself.

Specifically, we regard the image as a realization of a random field for which the wavelet

coefficients are independently distributed according to generalized Gaussian (GG) distribution laws. This model is motivated by two factors. First, recent work [19, 74, 75, 87] suggests that these models, which have heavier tails than a standard Gaussian distribution, provide accurate descriptions of the statistical distribution of wavelet coefficients in image data. Second, in addition to being a basis for $l^2(R)$, wavelets also are unconditional bases for more exotic function spaces whose members include functions with sharp discontinuities and thus serve as natural function spaces in which to analyze images [23, 31, 76, 96]. Because the norms in these Besov spaces are nothing more than weighted l_p, $0 < p$, norms of the wavelet coefficients, it is shown easily that deterministic regularization with a Besov norm constraint is equivalent to the specification of an appropriately parameterized GG wavelet prior model. From this perspective, our work can be viewed as an extension of the research done mostly in the area of image denoising.

Specifically, the wavelet domain image model of interest in this paper and the resulting nonlinear restoration algorithm are related to the large body of work originating from the *wavelet shrinkage estimators* first proposed by Donoho and Johnstone [39]. In a series of papers [33, 35–37], Donoho and Johnstone have shown that wavelet shrinkage estimators achieve near optimal estimation performance when the unknown signal belongs to Besov spaces. Later, several authors contributed to the advancements in the area. The notion of Besov regularization has been introduced by Amato and Vuza [96] and Chambolle-DeVore-Lee-Lucier [23] and the resulting theory was interpreted in a function space setting. On the other hand, Simoncelli and Adelson [19] developed a similar denoising scheme, which

they called *Bayesian wavelet coring*, by stochastically modeling the image subbands. GG modeling of image subands has later been analyzed by Moulin and Liu [80] and the connection between the Besov Spaces and GG priors has been established.

In this work, we make use of GG wavelet priors in a number of ways. We show that their use in an image restoration problem does in fact significantly improve the quality of edge information relative to more common smoothness priors. Inspired by the "lagged diffusivity" fixed point iteration proposed by Vogel and Oman [98] for the solution of the TV problem, we also provide an efficient algorithm for solving the non-linear optimization problem defining the restoration. By appropriately structuring the weighting pattern on the wavelet l_p norm, we demonstrate that these models provide an easy and flexible framework for adaptively determining the appropriate level of regularization as a function of the underlying structure in the image; in particular, we consider scale-to-scale or orientation based features. This adaptation is achieved through a data-driven choice of a vector of hyperparameters governing the prior model. For this task, we introduce and make use of a multi-variate generalization of the L-curve method developed in [56] for choosing a single hyperparameter. We verify the performance of this restoration scheme on a variety of images, comparing the results both to smoothness constrained methods and the TV restorations.

3.1 A Multiscale Prior Model for Images

A key component of our image restoration algorithm is the use of a multiscale stochastic prior model for f. To motivate the particular choice of prior model used here, we consider

the wavelet coefficients of a typical image at a particular resolution. Wavelet coefficients are obtained by differentiation-like operations. Since the spatial structure of many images typically consists of smooth areas dispersed with occasional edges, the distribution of wavelet coefficients should be sharply peaked around zero, due to the contribution of smooth areas, and have broad tails representing the contribution of the edges [19].

Following the work in [19, 87] on image coding and denoising, we model the distribution of wavelet coefficients of images by a Generalized-Gaussian (GG) density [74, 75]

$$P\left(f_j^{(i)}(m,n)|p,\kappa_j^{(i)}\right) \propto \exp\left\{-\frac{1}{p}\left|\frac{f_j^{(i)}(m,n)}{\kappa_j^{(i)}}\right|^p\right\}, \tag{3.1}$$

where $f_j^{(i)}(m,n)$ is the wavelet coefficient at scale j, orientation i and the position (m,n), $1 \leq m,n \leq 2^j$, $0 \leq p \leq 2$ is a parameter which determines the tail behavior of the density function and $\kappa_j^{(i)}$ is a *scale parameter* similar to the standard deviation of a Gaussian density. We will refer to the zero mean density in (3.1) as $GG(0,\kappa_j^{(i)},p)$. For $p = 1$ we have the Laplacian density and for $p = 2$ we have the familiar Gaussian density. The tails of the GG distribution become increasingly heavy as p approaches zero. We assume that the mean of the image is subtracted from the image and that the scaling coefficients $f_{j_0}^{(0)}(m,n)$, are i.i.d. $GG(0,\kappa_{j_0}^{(0)},p)$.

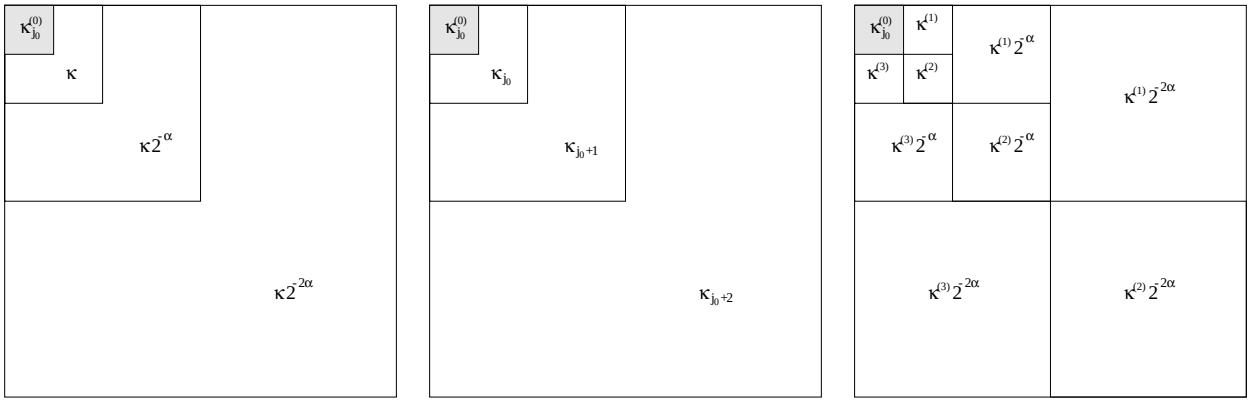
The specification of one κ parameter for each scale and orientation results in an image model far too complex to be of use in a restoration procedure. Nonetheless, the structure of the model in (3.1) coupled with the specification of the problem in the wavelet domain does suggest a variety of simplifications which are of use for the restoration problem. In this work, we consider the following three models:

1. **Model 1:** The scaling coefficients $f_{j_0}^{(0)}(m,n)$, are i. i. d. with $GG(0,\kappa_{j_0}^{(0)},p)$ and the wavelet coefficients are i. i. d. with exponentially decreasing variances, i.e. $f_j^{(i)}(m,n) \sim GG(0,\kappa 2^{-\alpha(j-j_0)},p)$, $i=1,2,3$, $j_0 \leq j \leq J-1$ with j_0 the coarsest scale, κ the scale parameter corresponding to j_0 and $\alpha \geq 0$ (Fig. 3.1(a)). The rationale behind this model is that it is equivalent to a deterministic modeling of the image as a member of a Besov space [76].

2. **Model 2:** The scaling coefficients $f_{j_0}^{(0)}(m,n)$, are i. i. d. with $GG(0,\kappa_{j_0}^{(0)},p)$ and the wavelet coefficients at a particular scale are i. i. d. with $GG(0,\kappa_j,p)$, $j=j_0,\ldots,J-1$ (Fig. 3.1(b)). This model is useful in cases where the variance of the wavelet coefficients at different scales cannot be well-approximated by a simple exponential law.

3. **Model 3:** The scaling coefficients $f_{j_0}^{(0)}(m,n)$, are i. i. d. with $GG(0,\kappa_{j_0}^{(0)},p)$ and the wavelet coefficients at different orientations (horizontal, vertical or diagonal) are distributed with $GG(0,\kappa^{(i)}2^{-\alpha(j-j_0)},p)$, $i=1,2,3$, $j_0 \leq j \leq J-1$ (Fig. 3.1(c)). Such a model is most suitable for images with significantly different characteristics in different orientations as might arise in geophysical restoration problems involving layered structures.

We make several observations regarding these models. First, they are indeed of low dimensionality. In addition to the α and p parameters, Model 1 is characterized by two κ coefficients: one for the coarsest scale scaling coefficients and one multiplying the exponential for the wavelet coefficients. There are a total of $J-j_0+1$ κ's for Model 2 and four κ values required to characterize Model 3. In subsequent sections, we shall see that the



(a) (b) (c)

Figure 3.1: (a) Model 1 (b) Model 2 (c) Model 3 prior models.

number of regularization parameters to be determined in the restoration algorithm is equal to the number of κ's characterizing the prior model being used. Moreover, an appropriate on-line choice of the model parameters provides a mechanism for adapting the level of regularization in an image to the underlying scale-to-scale structure (Models 1 and 2) or to orientation-dependent structure (Model 3). While the above three models certainly do not represent an exhaustive enumeration of all possible multiscale regularization approaches, as seen in Section 4.4.2, they do provide a strong indication as to the utility of this type of modeling technique for image restoration.

3.1.1 Hyperparameter Estimation

In this section, we comment on the estimation of the hyper-parameters, p, α, and $\kappa_j^{(i)}$. In a restoration algorithm, these parameters could be estimated from the data by assigning appropriate priors to each and maximizing the resulting log-likelihood function with respect to

the hyper-parameters and the image. However, such an approach presents many computational difficulties and unnecessarily complicates the problem. Instead, we choose to simplify the problem by fixing the p and α *a priori*. Generally, the performance of the regularizer is impacted to a greater extent by the on-line identification of the κ parameters [40] (or as explained in subsequent sections, quantities closely related to κ) so we concentrate our efforts on identifying good choices of κ.

The issue of selecting an appropriate p is extensively discussed in Section 3.21 . As for the selection of α, we propose using a fixed *a priori* choice obtained from the empirical study of a number of images. We computed the α values for set of images seen in (Fig. 3.2) by calculating the slope of the line fitted to the logarithm of the p-norm of the wavelet coefficients at different scales. According to our findings, for most images representing natural scenes the α value which produced the best fit to the image data under the Model 1 scheme (for $p = 1$) fell between 0.6 and 1.6 with mean $\hat{\alpha} \approx 1.2$ and variance $E\{(\alpha - \hat{\alpha})^2\} \approx 0.11$ (see Fig. 3.2). In the absence of the knowledge about the exact value of α for a specific image, $\hat{\alpha}$ can be used as a substitute. We evaluated the effects of varying the α value on a number of restoration problems and saw that the results were relatively insensitive to variations of α in the range $[0.6, 1.6]$ suggested by the observations. That is, the performance of multiscale restoration was robust against misspecifications of the α. This aspect of our regularization scheme can also be observed from the numerical examples supplied in Section 4.4. The fact that the same α is used for Model 1 restorations in all cases irrespective of the exact image considered, yet that the restored images were remarkably similar to Model 2 restorations,

clearly illustrates this point.

In [41], Dufour and Miller investigate the degradation in estimator performance for $p = 2.0$ as the α is varied around the optimal value. The results in that work indicate that the performance is quite insensitive to variations in α. In particular, their results show that 30% error in the specification of α results in 10% degradation in MSE. This result was obtained for $p = 2.0$. The work of Moulin and Liu [80] indicates that the GG distribution family becomes more robust to misspecifications of the model parameters as p decreases, hence we expect even lower degradation for $p \leq 1.0$.

Figure 3.2: A set of images and the corresponding α values for $p = 1.0$. Mean is $\hat{\alpha} = 1.1685$.

To conclude, use of a fixed *a priori* estimate of α is justified on the grounds that $\alpha = 1.2$ (for $p = 1.0$) represents a good value for most natural scenes, and performance of the Model

1 restoration scheme introduced in our paper is robust to variations in α around the optimal value as both our results and previous research [41, 80] suggest.

3.2 A Multiscale Image Restoration Algorithm

The MAP estimate of the wavelet coefficients of the original image is found by maximizing the log-likelihood function in (2.23). Substituting the prior probability density developed in Section 3 into (2.23), the MAP estimate of $\hat{\mathbf{f}}$ is seen to be the minimum of the following cost function with respect to $\hat{\mathbf{f}}$ (assuming for the time being that the $\lambda_j^{(i)}$ are known)

$$J(\hat{\mathbf{f}}, \boldsymbol{\lambda}) = \|\hat{\mathbf{g}} - \hat{\mathbf{H}}\hat{\mathbf{f}}\|_2^2 + \lambda_{j_0}^{(0)} \|\mathbf{f}_{j_0}^{(0)}\|_p^p + \sum_{j=j_0}^{J-1} \sum_{i=1}^{3} \lambda_j^{(i)} \|\mathbf{f}_j^{(i)}\|_p^p \,, \tag{3.2}$$

where $\lambda_j^{(i)} = \frac{2\sigma^2}{p\left(\kappa_j^{(i)}\right)^p}$ are weighting parameters and $\boldsymbol{\lambda} = [\lambda_{j_0}^{(0)}, \lambda_{j_0}^{(1)}, \ldots, \lambda_{J-1}^{(3)}]^T$. The formulation in (3.2) easily accommodates the Model 1-3 regularization schemes discussed in Section 3 by defining the appropriate relationships for $\lambda_j^{(i)}$. For example, putting $\lambda_{j_0}^{(0)} = \lambda_1$ and $\lambda_j^{(i)} = \lambda_2 2^{\alpha(j-j_0)}$ results in the Model 1 regularization scheme while assigning a different λ_j to each scale in the wavelet domain without regarding the orientation we obtain the Model 2 regularization scheme. Suppose that $J(\hat{\mathbf{f}}, \boldsymbol{\lambda})$ has a minimum in $\hat{\mathbf{f}}$. Then at a stationary point $\hat{\mathbf{f}}^*$, the gradient of $J(\hat{\mathbf{f}}, \boldsymbol{\lambda})$ must vanish. Unfortunately, the l_p norm terms appearing in (3.2) are not differentiable for $p \leq 1$. Hence, we propose the following *smooth* approximation to the l_p norm, raised to the power p, as in [98]:

$$\|\mathbf{x}\|_p^p \approx \sum_i \left(\left(|x_i|^2 + \beta \right)^{p/2} - \beta^{p/2} \right) \,, \tag{3.3}$$

where $\beta \geq 0$ is a stabilization constant and x_i denotes the i-th element of the vector $\mathbf{x}$. Substituting (3.3) into (3.2) and taking the gradient of the cost function we arrive at the following equation for $\hat{\mathbf{f}}^*$:

$$\mathbf{D}^* = \operatorname{diag}\left[\frac{\lambda_i}{(|\hat{f}_i^*|^2 + \beta)^{1-p/2}}\right]_{i=1}^{N^2}, \tag{3.4}$$

$$\left(\hat{\mathbf{H}}^T\hat{\mathbf{H}} + \frac{p}{2}\mathbf{D}^*\right)\hat{\mathbf{f}}^* = \hat{\mathbf{H}}^T\hat{\mathbf{g}}, \tag{3.5}$$

where $\hat{f}_i^*$ is the i-th element of $\hat{\mathbf{f}}^*$ and λ_i is the associated regularization parameter. The above equation gives the first order conditions that must be satisfied by $\hat{\mathbf{f}}^*$. By direct analogy with the lagged diffusivity method of Vogel and Oman [98], we can develop a fixed point iteration to solve for $\hat{\mathbf{f}}^*$. Starting with an initial point $\hat{\mathbf{f}}^0$, we solve the following equation for $\hat{\mathbf{f}}^{k+1}$:

$$\left(\hat{\mathbf{H}}^T\hat{\mathbf{H}} + \frac{p}{2}\mathbf{D}^k\right)\hat{\mathbf{f}}^{k+1} = \hat{\mathbf{H}}^T\hat{\mathbf{g}}, \tag{3.6}$$

where $\mathbf{D}^k$ is obtained by replacing $\hat{f}_i^*$ by $\hat{f}_i^k$ in (3.4). The iteration is terminated whenever $\frac{\|\hat{\mathbf{f}}^{k+1}-\hat{\mathbf{f}}^k\|}{\|\hat{\mathbf{f}}^k\|} < \gamma$, with γ being a small positive constant ($\gamma = 10^{-5}$ is used in our experiments). The fixed point iteration in (3.6) is a special case of the "half quadratic regularization" scheme introduced by Geman *et. al.* [46] and the ARTUR scheme due to Charbonnier *et. al.* [25]. Adopting the notation in [25] we define the following function

$$\phi(t) = \left(t^2 + \beta\right)^{\frac{p}{2}} - \beta^{\frac{p}{2}}. \tag{3.7}$$

The approximated cost function can be expressed in terms of the function $\phi(t)$. Furthermore, $\phi(t)$ satisfies the conditions 1-8 presented in Section 2.1.4 so that the restoration algorithm is convergent in the sense that the sequence $J(\hat{\mathbf{f}}^k, \boldsymbol{\lambda})$ is convergent and that $\hat{\mathbf{f}}^{k+1} - \hat{\mathbf{f}}^k \rightarrow_{k\rightarrow+\infty} 0$

pointwise. In the special case where $\phi(t)$ is convex (which occurs if $p \geq 1$) and $\hat{\mathbf{H}}$ is full-rank, the iterates $\hat{\mathbf{f}}^k$ converge and the computed solution is the unique minimum of (2.23). However, when $p < 1$, $\phi(t)$ is not convex and the algorithm computes a local minimum of (3.2) [25].

The iterative algorithm in (3.6) requires the solution of a very large linear matrix equation. Note that the matrix appearing on the right hand side of (3.6) is symmetric and positive definite. Therefore, the conjugate gradient (CG) algorithm [49] can be conveniently used to compute the solution $\hat{\mathbf{f}}^{k+1}$ in (3.6) at each step. In this way, the algorithm given in (3.6) is doubly iterative in that an outer iteration is used to update the solution $\hat{\mathbf{f}}^k$ and an inner iteration is used to solve the system of equations in (3.6) by the CG method. The special structure of the matrices $\hat{\mathbf{H}}$ and $\mathbf{D}^k$ could be used to decrease the computational cost substantially. The first matrix, $\hat{\mathbf{H}}$, is merely the wavelet domain representation of our degradation operator. If the kernel is convolutional, it has been shown by Zervakis *et. al.* [102] that this matrix can be diagonalized by a special Fourier transform matrix by invoking the circulant assumption. On the other hand, the second matrix $\mathbf{D}$ is diagonal in the wavelet domain. Therefore, the vector matrix multiplications required for the implementation of the CG algorithm can be computed in an efficient way by going back and forth between the wavelet and the Fourier transform domains. In this case, the cost of multiplying a vector with the matrix $\hat{\mathbf{H}}^T\hat{\mathbf{H}} + \frac{p}{2}\mathbf{D}^k$ is dominated by the cost of the FFT which is $O(N^2 \log N)$, where $N = 2^{2J}$ is the number of pixels in the image.

We note that the iterative algorithm in (3.6) can be efficient even in the case where $\hat{\mathbf{H}}$

is not convolutional since the wavelet domain representation of a wide range of operators is sparse [39]. In those cases, standard techniques for sparse matrices can be used to reduce computational complexity.

3.2.1 Selection of GG family; The Shape Parameter p

The possibility that multiple local minima of (2.23) may exist for $p < 1$ presents an interesting trade-off. From a computational viewpoint, it is highly desirable to use $p \geq 1$, since in this case the cost function is convex and global convergence is guaranteed. However, in empirical studies of the wavelet coefficients of images it has been shown that the GG model for the distribution of the wavelet coefficients usually corresponds to $p < 1$ and a typically recognized value is $p = 0.7$ [74].

There are basically two arguments against using $p < 1$:

1. $p < 1$ leads to a non-convex optimization task.

2. We have observed through numerous experiments that there is no practical difference between the restored images corresponding to the $p = 1$ and the best p-value (in terms of the model fit to the original image) cases (see Section 4.4 on how to compute p).

The item 1. above is obviously an important consideration for any image restoration algorithm. In general, if the optimization task is non-convex, one has to find a way to determine the global minimum using, for example, simulated annealing or a continuation method. Since the computational effort required by a global search is very high, such an approach can only be justified on the grounds that the restored images are significantly better when $p < 1$,

which brings us to item 2. above. We would like to point out that use of GG priors with $0 < p \leq 2$ was analyzed in the context of image denoising by Moulin and Liu in [80]. In this work, numerical experiments performed on the *barbara* and *peppers* images have shown that there was only 15% difference in the MSE values (MSE $= \|f - \hat{f}\|_2^2/N^2$, N =number of pixels and f and $\hat{f}$ are the original and estimated images, respectively) over the range $0.5 < p \leq 1$. In particular, it was observed that $p = 0.7$ minimized the MSE for both images. For $p = 0.7$ and $p = 1.0$, the respective MSE values were 33.64 and 33.68 for the *barbara* image and 30.00 and 30.53 for the *peppers* image, a minor difference.

In numerous experiments we performed on typical indoor and outdoor scenes as well as the artificial datasets provided in the Donoho's Wavelab toolbox, we observed no significant improvement in the restored images when using the best p-value, which was typically between 0.5 and 1.0, as compared to $p = 1$. Furthermore, we have also seen that $p = 1$ consistently gave the better results, both in terms of visual satisfaction and the MSE values, than the entire range $1 < p \leq 2$. Our observations obtained through empirical study of various images coupled with the results in Moulin and Liu's work suggest that the benefits of using $p < 1$ as opposed to $p = 1$ are so small that we would rather use $p = 1.0$ for which the resulting optimization problem is convex.

Note that for $p < 1$, the fixed point iteration in (8-9) still works but it is not guaranteed to provide the global minimum. In order to approximate the global minimum, we describe a very simple continuation method (in Section 4.4) where the solution for $p < 1$ is obtained by supplying the solution for $p = 1.0$ (which is unique) as the initial approximation to the

half-quadratic algorithm in equation 3.6. Since the optimization problem is continuous in p, we expect the solution obtained by this process to be an approximation to the global minimum. There is still no guarantee that the resulting solution is the global optimum, nevertheless we obtained good results using this method. The first example in Section 4.4 shows that only modest improvement can be gained by taking $p < 1$. In this example, we computed the p-value that produces the best fit to the high-frequency subbands of the *mandrill* image, which was found to be $p = 0.7280$, and then computed a restoration by using the continuation method described above. Clearly, this is not a realistic situation since we used the original image for estimating the p-value. The results reveal that there is almost no difference in the visual quality of the restoration for $p < 1$ as compared to $p = 1.0$. This example, together with earlier results in [80], suggests that $p = 1.0$ can be conveniently used as an *a priori* estimate. Note that we do not claim that $p = 1.0$ should always be used. In fact, there may be cases where a reliable prior estimate of p is available and that it is known that using this value will make a big difference in the quality of the restoration as compared to using $p = 1.0$. Rather, we are saying that in the absence of such information, $p = 1.0$ should be used since it was observed empirically that $p = 1.0$ provides good restorations for typical natural scenes and global convergence is guaranteed.

Finally, we would like to point out that Moulin's work indicates that adapting the p-value to each scale and orientation in the wavelet domain does not improve the restoration results. On the contrary, Moulin found out that using a scale varying p results in degraded performance over the simpler estimator using a fixed p for all scales [80].

3.2.2 Selection Stabilization Constant β

The role of the parameter β is two-fold. First it controls how close the approximation in (3.3) is to the original l_p norm. Using a relatively small β provides better restoration of edges in the image since a smaller β value provides better approximation to the l_p norm. Second, it essentially determines the convergence speed of the algorithm. While we do not intend to carry out a numerical analysis of the fixed point iteration in (3.6), the basic reason is that for $\beta = 0$, $\phi(t)$ in (3.7) is not differentiable at $t = 0$ and instability in the numerical computations may arise. If β is relatively large, the algorithm is fast, and the convergence speed deteriorates as β gets smaller. Therefore, β should be set so as to achieve a compromise between the convergence speed and the edge preservation. Based on our experience on natural scenes, we found that restorations obtained for $\beta \approx 1$ were visually indistinguishable from the restorations obtained for $\beta \ll 1$.

3.2.3 Numerical Examples

In this section we demonstrate the performance of the proposed multiscale restoration algorithm on a simple deconvolution problem. The "1-D image" under consideration is the "Blocks" [37] sequence as seen in Fig. 3.3(a). This image was degraded by a Gaussian convolutional kernel, $h(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{x^2}{2\sigma^2}\}$, with $\sigma = 4.0$. Zero mean white noise was added to the degraded image to set the SNR at 30dB. The degraded image obtained in this way is shown in Fig 3.3(b). We computed two restorations corresponding to the the CLS

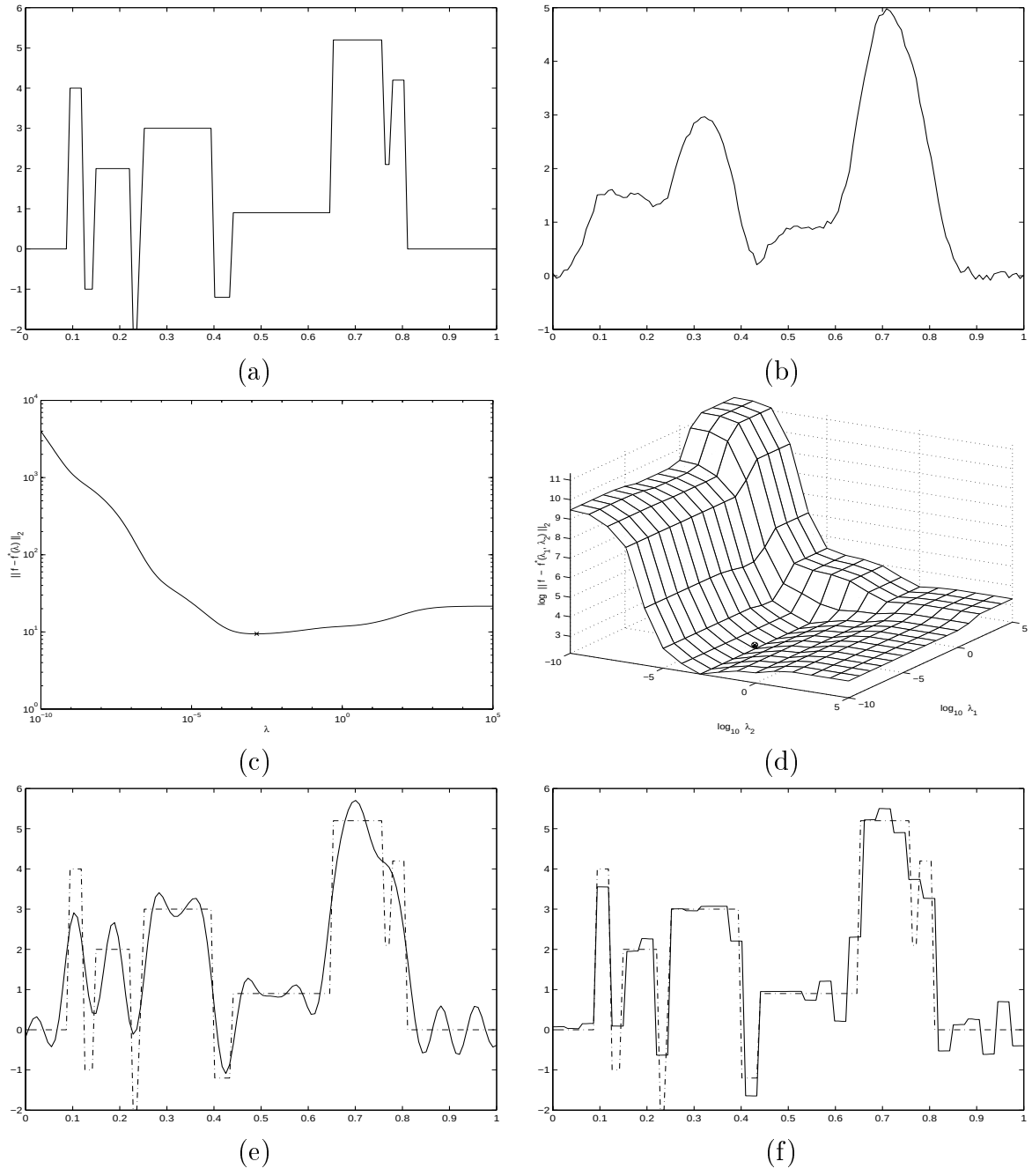


Figure 3.3: (a) Original Blocks image. (b) Degraded Blocks. (c) Restoration error versus regularization parameter for the CLS algorithm. (d) Restoration error versus regularization parameters for the multiscale algorithm. In this plot, λ_1 regularizes the scaling coefficients while λ_2 is the regularization parameter for wavelet coefficients (e) CLS restoration. (f) Multiscale restoration with $p = 1.0$.

algorithm with a differential regularizer and our multiscale algorithm with Model 1 regularization scheme. In order to evaluate the potential performance improvement of our multiscale algorithm over the CLS algorithm, we chose the required regularization parameters by minimizing the restoration error defined as, $\|\mathbf{f} - \mathbf{f}^*(\lambda)\|_2^2$, with respect to the regularization parameters. Figure 3.3(c) shows the variation of the restoration error with the regularization parameter λ for CLS. In Fig. 3.3(e) we display the CLS restoration obtained by using the "best" regularization parameter whose location is indicated in Fig. 3.3(c) with a cross. For our multiscale algorithm, we employed a 5-level wavelet decomposition with the Haar basis. In accordance with the Model 1 regularization scheme, we have two regularization parameters λ_1 and λ_2 which determine the amount of regularization applied to the scaling and the wavelet coefficients, respectively. The parameters of our multiscale algorithm were: $p = 1.0$, $\beta = 0.01$ and, $\alpha = 1.2$. The regularization parameters for the scaling and the wavelet coefficients were obtained by minimizing the restoration error with respect to both parameters. The variation of the restoration error with respect to the regularization parameters is displayed in Fig. 3.3(d). In this plot, λ_1 regularizes the scaling coefficients while λ_2 is the regularization parameter for the wavelet coefficients. Figure 3.3(f) shows the restoration obtained by our multiscale algorithm.

As observed from Fig. 3.3(e), the CLS algorithm produces an over-smooth restoration and completely fails to recover the edge information in the image. On the other hand the multiscale algorithm does a much better job in preserving edges in the image while suppressing the high frequency oscillations in flat areas caused by the presence of noise.

This example shows that our proposed multiscale image restoration algorithm is capable of producing much better results than the conventional CLS algorithm if the image under consideration contains edges. However, the results in this example were obtained under the assumption that the best parameters (in the sense of minimizing the restoration error) were available. In a practical situation, these parameters are typically unknown and therefore should be estimated from the available data. In order to accomplish this task, in the next section we develop a multi-dimensional extension of the conventional L-curve method to select multiple regularization parameters required by our multiscale algorithm. Then we will combine the generalized L-curve method with the multiscale image restoration algorithm introduced here to obtain a highly flexible image restoration scheme which automatically adjusts the degree of regularization according to the scale or orientation varying structure of the underlying image. Examples will be provided in Section 4.4 which compare and contrast the performance of our algorithm to that of the classical and more recent edge-preserving image restoration algorithms in practical situations.

Chapter 4

A Multi-dimensional Extension of the L-curve Method: The L-hypersurface

As described in the previous chapters, our multiscale IRR algorithm requires the determination of several regularization parameters which are essential for good performance. In the literature, there are a number of methods developed for one-parameter regularization schemes including the discrepancy principle [79], generalized cross validation [26], the L-curve method [56], the minimum bound method [47, 83, 84] e.t.c. [61, 62, 90, 101]. One can even use the so-called experimental method, in which the regularization parameter is tuned by the user, which was jokingly described in a conference [89] by James Nagy[1] as, "turning the focus ring of your photography machine which comes to sharp focus when you turn it just the right amount". It is easy to see the complications of a two-parameter regularization

[1]Professor Nagy is a member of faculty in the Department of Mathematics at Emory University.

scheme by extending James Nagy's analogy by visualizing a more complicated photography machine that has two focus rings to be adjusted. Such a photography machine would require significantly more user input and time and therefore it would be nice if an "auto-focus" mechanism could be developed. In search of such a mechanism we decided to use a multi-variate generalization of one of the known methods in the literature. The extension of methods developed for a single parameter to choose multiple regularization parameters is often a non-trivial task. In this thesis, we introduce the L-hypersurface method for use in linear inverse problems that require multiple regularization parameters. The L-hypersurface is based on the classical L-curve method. It is an M-dimensional (M is the number of regularization parameters) function of the regularization parameters and provides significant information about the likelihood of a particular parameter set being optimum.

4.1 Why the L-curve?

Our first requirement for an appropriate parameter selection method is that it not depend on any side information such as the signal norm, $\|\mathbf{f}\|$, or the noise variance σ_e^2. This immediately rules out many possibilities, leaving the L-curve and the GCV method as the most prominent candidates.

GCV has a number of desirable properties. The solution estimates converge to the true solution as the error norm goes to zero (in a continuous setting) or the number of data samples goes to infinity (in a semi-discrete setting). GCV locates the best parameter by

minimizing the GCV function which is reproduced below for convenience:

$$G(\lambda) = \frac{\|\mathbf{g} - \mathbf{H}\mathbf{f}^*(\lambda)\|_2^2}{\text{trace}(\mathbf{I} - \mathbf{H}\mathbf{L}(\lambda))} , \tag{4.1}$$

where $\mathbf{L}(\lambda)$ is the matrix which maps the data to the regularized solution, i.e. $\mathbf{f}^*(\lambda) = \mathbf{L}(\lambda)\mathbf{g}$. A multi-dimensional extension of the GCV method is conceptually straightforward as the method can be trivially extended by considering a two-parameter GCV function. However, the GCV method is not suitable for our multiscale regularization algorithm because computing the term in the denominator of the GCV function in (4.1) is computationally a formidable task. The diagonal entries of the matrix $\mathbf{L}(\lambda)$ in (4.1) can only be computed from the knowledge of the wavelet domain representation of the entire matrix $\mathbf{H}$. For a 256×256 image $\mathbf{H}$ is an $65,536 \times 65,536$ matrix and demands enormous storage requirements.

Although only a few theoretical properties of the L-curve method are known (see for example Hansen [56] and Reginska [86]) the method has found considerable interest due to its simplicity, and has been applied successfully in a variety of different applications. On the other hand, the L-curve method has been criticized on the grounds that it fails to find the right regularization parameter [54, 97]. More specifically, Vogel [97] is concerned with the limiting process of a semi-discrete model problem

$$g_i = \mathbf{A}_n f + \eta_n, \quad i = 1, \ldots, n , \tag{4.2}$$

where $\mathbf{A}_n$ is an operator from a Hilbert space $\mathcal{H}$ to $\mathcal{R}^n$, η is the discrete white noise and $f \in \mathcal{H}$. In this setting, a regularization parameter selection method is said to be convergent

if it yields a parameter $\lambda(n)$ for which

$$E\left\{||f_{\lambda(n),n} - f||^2\right\} \to 0 \quad \text{as} \quad n \to \infty \ , \tag{4.3}$$

where E denotes the mathematical expectation operator. He showed that, under some specific assumptions on f and the decay of the singular values of $\mathbf{A}_n$, the regularization parameter obtained by the L-curve method, $\lambda_{L-curve}(n)$, does not go to zero as $n \to \infty$ and hence (4.3) cannot be satisfied.

Similarly, Hanke [54] has considered a continuous problem

$$y^\delta = Kx, \quad K \ : \ \mathcal{X} \to \mathcal{Y}, \tag{4.4}$$

where $\mathcal{X}$ and $\mathcal{Y}$ are Hilbert spaces and $y^\delta \in \mathcal{Y}$ is the given data, possibly contaminated by noise of magnitude $\delta > 0$. If λ^δ is the regularization parameter obtained by a regularization parameter selection method, the following must be satisfied for convergence:

$$x^\delta_{\lambda^\delta} \to K^\dagger y \quad \text{as} \quad y^\delta \to y \ , \tag{4.5}$$

where $K^\dagger$ is a pseudoinverse of K and y is the noiseless data. Again, Hanke has shown that the regularization parameter obtained by the L-curve method fails to satisfy (4.5) for "smooth" x. Hanke concluded that L-curve might yield a parameter which is likely to be too small as $\delta \to 0$.

It is not clear whether the convergence issue has any importance in a practical situation where the noise level is significantly non-zero and the number of data samples is fixed. Furthermore, Vogel and Hanke's results apply only under very specific assumptions on the

signal/operator, limiting their scope. Based on our experience and the results regarding the application of the L-curve method to practical instances of inverse problems we can only say that the method is indeed useful. As we will see in the next few sections, a multi-dimensional extension of the conventional L-curve can be developed and implemented for the multiscale IRR algorithm.

4.2 The L-hypersurface

In this section we will develop a multi-dimensional extension of the conventional L-curve method using a generic multiple-parameter regularization scheme that includes many popular regularization schemes as its special cases. Before proceeding we note that the idea of a multi-variate L-curve was first proposed by Brooks [15] *et. al.* in the context of the inverse problem of electrocardiography. Facing the problem of choosing two or more parameters simultaneously, Brooks [15] *et. al.* tried to extend the idea of the L-curve by drawing the residual norm against two side constraint norms. They named the resulting plot the "L-surface" (the L-hypersurface in our terminology).

In order to extend the L-curve, we consider the following multiply constrained regularization approach:

$$\min_{\mathbf{f}} \left\{ \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \sum_{i=1}^{M} \lambda_i \Phi_i(\mathbf{R}_i\mathbf{f}) \right\}, \quad \mathbf{R}_i \in \mathcal{R}^{m\times n} \quad , \tag{4.6}$$

where $\mathbf{R}_i$, $i = 1, \ldots, M$ are regularization operators and λ_i are the corresponding regularization parameters, $\Phi_i(\mathbf{R}_i\mathbf{f}) = \sum_{j=1}^{m} \phi_i([\mathbf{R}_i\mathbf{f}]_j)$ and the notation $[\mathbf{R}_i\mathbf{f}]_j$ denotes the jth element

of the vector $\mathbf{R}_i\mathbf{f}$. We assume that $\phi_i(t)$ is a continuously differentiable, convex, non-negative ($\phi_i(t) \geq 0, \ \forall t$), even function which satisfies the following conditions [25]:

i. $\phi'(t) \geq 0, \ \forall t \geq 0,$

ii. $\lim_{t \to 0^+} \frac{\phi_i'(t)}{t} = C, \ \ 0 < C < \infty,$

where prime denotes differentiation. These two conditions on $\phi_i(t)$ ensure that there is a unique solution to the minimization problem in (4.6) and that a half-quadratic algorithm can be found to compute the minimum of (4.6).

The formulation in (4.6) includes many popular regularization techniques as its special cases. For example, by taking, $\phi_i(t) = t^2, \ \ i = 1, .., M$, and $\mathbf{R}_i$ as a discrete approximation to the ith order differentiation, we obtain the conventional Tikhonov method with the Sobolev norm as a constraint. The multiscale IRR algorithm developed in Chapter 3 is also a special case of (4.6). In this case, the quantities of interest $(\mathbf{g}, \mathbf{H}, \mathbf{f}, \mathbf{n})$ represent the wavelet decomposition of the related quantities, $\phi_i(t) = (|t|^2 + \epsilon)^{p/2}, i = 1, \ldots, M$ with $1 \leq p \leq 2$ and ϵ a small positive constant, and $\mathbf{R}_i, \ \ i = 1, .., M$ are operators extracting the desired portions of the wavelet coefficients.

By taking the gradient of (4.6) with respect to $\mathbf{f}$ and setting the result equal to zero we obtain the following first order condition that must be satisfied by the solution $\mathbf{f}^*$:

$$\mathbf{K}_{\mathbf{f}^*}\mathbf{f}^* = \mathbf{H}^T\mathbf{g}, \tag{4.7}$$

where $\mathbf{K}_{\mathbf{f}^*} = \mathbf{H}^T\mathbf{H} + \frac{1}{2}\sum_{i=1}^{M} \lambda_i \mathbf{R}_i^T \mathrm{diag}_{k=1,\ldots,m}\left[\frac{\phi_i'([\mathbf{R}_i\mathbf{f}^*]_k)}{[\mathbf{R}_i\mathbf{f}^*]_k}\right]\mathbf{R}_i$. Note that $\mathbf{f}^*$ can be approximated by a half-quadratic iterative algorithm (Section 2.1.4) .

To construct the L-hypersurface, we first introduce the following quantities

$$\mathbf{f}^*(\boldsymbol{\lambda}) = \arg\min_{\mathbf{f}} J(\mathbf{f}, \boldsymbol{\lambda}) \tag{4.8}$$

$$z(\boldsymbol{\lambda}) = \|\mathbf{g} - \mathbf{H}\mathbf{f}^*(\boldsymbol{\lambda})\|_2^2, \tag{4.9}$$

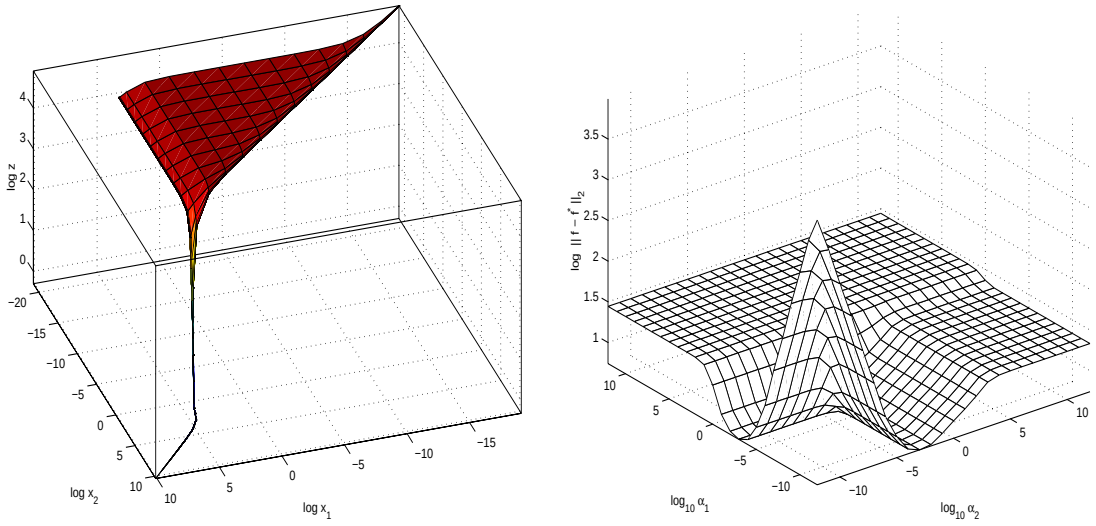
$$x_j(\boldsymbol{\lambda}) = \|\mathbf{R}_j\mathbf{f}^*(\boldsymbol{\lambda})\|_p^p, \quad j = 1, \ldots, M. \tag{4.10}$$

With the above definitions, the L-hypersurface is defined as a subset of $\mathcal{R}^{M+1}$ associated with the map $\mathcal{S}(\boldsymbol{\lambda}) : \mathcal{R}_+^M \rightarrow \mathcal{R}^{M+1}$ such that

$$\mathcal{S}(\boldsymbol{\lambda}) = (\bar{x}_1(\boldsymbol{\lambda}), \ldots, \bar{x}_M(\boldsymbol{\lambda}), \bar{z}(\boldsymbol{\lambda})), \tag{4.11}$$

where $\bar{x}_i = \log x_i, \quad i = 1, \ldots, M$ and $\bar{z} = \log z$. In simple terms, the L-hypersurface is a plot of the residual norm as a function of the constraint norms drawn in log scale for a range of regularization parameters.

Figure 4.1(a) and (b) show a typical L-hypersurface and the corresponding estimation error surface, $\|\mathbf{f} - \mathbf{f}^*(\boldsymbol{\alpha})\|_2$, for `baart(100)` problem from Hansen's Regularization Toolbox. In Fig. 4.1(a), $x_1 = \|\mathbf{f}^*(\boldsymbol{\alpha})\|_2$ and $x_2 = \|\mathbf{D}^{(1)}\mathbf{f}^*(\boldsymbol{\alpha})\|_2$ where $\mathbf{D}^{(1)}\mathbf{f} = [f_2 - f_1, f_3 - f_1, \ldots, f_n - f_{n-1}]^T$, f_i representing the ith element of the vector $\mathbf{f}$. Observing Fig. 4.1(a) we realize that the L-hypersurface, for this example, has an odd shape which is getting narrower at the bottom and almost degenerating to a 1-D curve in $\mathcal{R}^3$. This phenomena considerably complicates the visualization and the interpretation of the behavior of the L-hypersurface. Note also that we are unable to establish a direct relationship between the L-hypersurface in Fig. 4.1(a) and the estimation error surface in Fig. 4.1(b). Considering these difficulties we propose extending the notion of curvature for plane curves to the notion of Gaussian



(a) (b)

Figure 4.1: (a) A typical L-surface, (b) corresponding estimation error surface.

curvature for surfaces in the hope of finding a better way of visualizing and analyzing the L-hypersurface.

4.3 The Gaussian Curvature of the L-hypersurface

In a one-parameter regularization scheme, the corner of the L-curve is defined as the point with the maximum curvature on the L-curve. The curvature at a point q of a plane curve is the rate with which the angle between the tangent to the curve at q and the x-axis changes with respect to the arclength [32, 81] (arclength $\frac{ds}{dt} = \|\boldsymbol{\beta}'(t)\|_2$ where $\boldsymbol{\beta}(t) = [x(t), y(t)]$ is the curve parameterized by $t \in \mathcal{R}$). The curvature can be viewed as a simple, geometrically meaningful transformation of the L-curve which emphasizes the turning points, which for our application are practically the only interesting points. Our goal in using the Gaussian

curvature of the L-hypersurface is to simplify the interpretation and to reveal important regions of the L-hypersurface where good regularization parameters are located. In the next section, we describe the geometrical meaning and the mathematics of the Gaussian curvature.

4.3.1 Gaussian Curvature: Geometrical Interpretation and the Math

The Gaussian curvature is a scalar quantity which measures how much and in what way a surface is warped in the vicinity of a point. Before going forward with the definition we note that the Gaussian curvature exists only if the surface under consideration is *regular*. Roughly speaking, a regular surface is obtained by taking pieces of a plane, deforming them, and rearranging them in such a way that the resulting figure has no sharp points, edges, or self intersections so that there is a unique definitive tangent plane at points of the figure [32,81]. Mathematically, to be regular, a surface has to be differentiable with no self-intersections and the columns of the following $(M+1)\times M$ matrix of the linear map $d\mathcal{S}$ must be independent:

$$d\mathcal{S} = \begin{bmatrix} \frac{\partial \bar{x}_1}{\partial \lambda_1} & \cdots & \frac{\partial \bar{x}_1}{\partial \lambda_M} \\ \vdots & \ddots & \vdots \\ \frac{\partial \bar{x}_M}{\partial \lambda_1} & \cdots & \frac{\partial \bar{x}_M}{\partial \lambda_M} \\ \frac{\partial \bar{z}}{\partial \lambda_1} & \cdots & \frac{\partial \bar{z}}{\partial \lambda_M} \end{bmatrix}. \tag{4.12}$$

Thus, each of the minors of order $M-1$ of $d\mathcal{S}$ must have a determinant different than zero.

The regularity condition guarantees that for every point q on $\mathcal{S}$ we can associate a unique plane that is tangent to the surface at q. The tangent plane at q, which we denote by T_q,

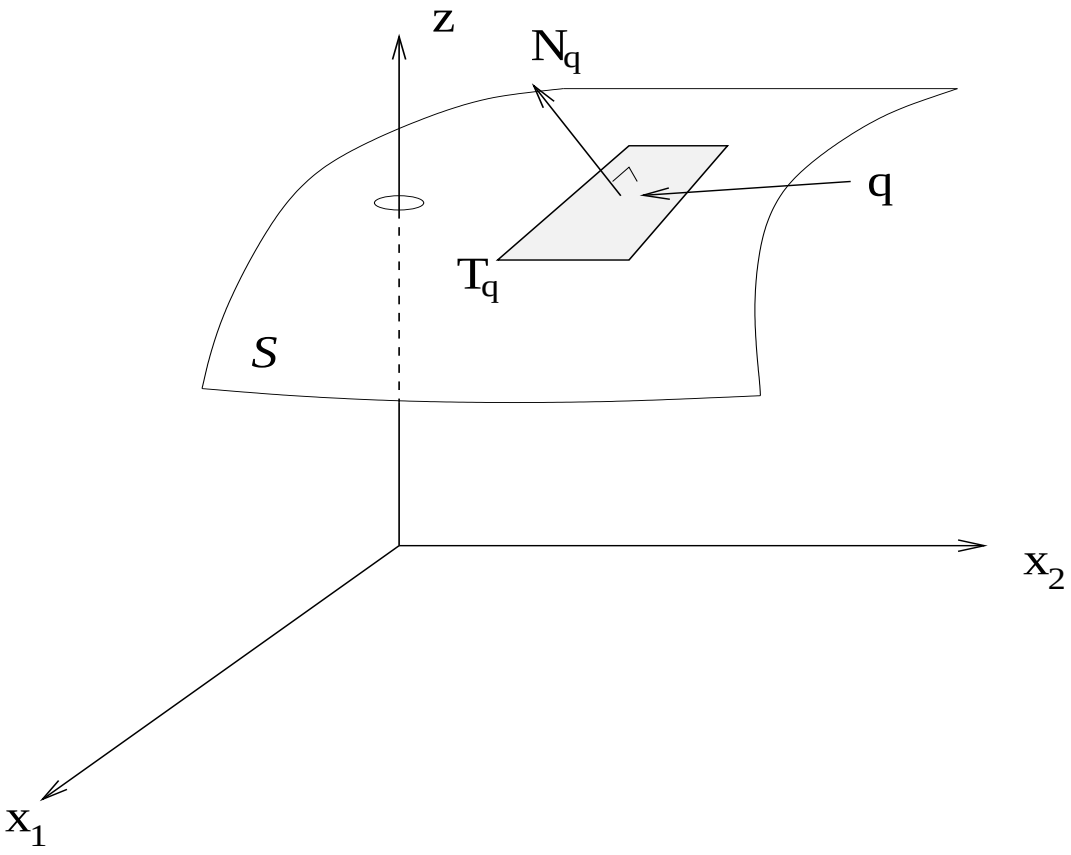


Figure 4.2: Tangent plane T_q and the normal vector N_q to a surface $\mathcal{S}$ at point q.

is spanned by the columns of the matrix $d\mathcal{S}$ in (4.12). There are two unit vectors that are normal to the tangent plane T_q at q; each of them is called the unit normal vector at q and is denoted by N_q. They can be obtained by finding the vector lying in the null-space of $d\mathcal{S}^T : M \times (M+1)$ in (4.12). Let's define a curve $\mathbf{s}(t) = \mathcal{S}(\boldsymbol{\lambda}[t])$, $\mathbf{s}(0) = q$ with $\boldsymbol{\lambda}(t) = [\lambda_1(t), \ldots, \lambda_M(t)]$ and $t \in (-\epsilon, \epsilon)$ is a parameter. The normal to $\mathcal{S}$ along $\mathbf{s}(t)$ is $N(\mathbf{s}[t]) = N_q(t)$. Then

$$\frac{dN[\mathbf{s}(t)]}{dt} = N_q'(0) = dN(\mathbf{s}'[0]) = \mathbf{A}_q \mathbf{s}(t) \ : \ T_q \to T_q \tag{4.13}$$

is a linear map and measures how $N(t)$ pulls away from N_q in a neighborhood of q. In one dimension the matrix $\mathbf{A}_q$ characterizing the linear map at the point q is a scalar and is called the curvature. In higher dimensions, the determinant of this linear map $det(\mathbf{A}_q)$ is called the Gaussian curvature. Working out the math it can be shown that for the L-hypersurface the Gaussian curvature at a point $\boldsymbol{\lambda}$ is given by [45]

$$\kappa(\boldsymbol{\lambda}) = \frac{(-1)^M}{w^{M+1}}|\mathbf{P}|\,, \tag{4.14}$$

where $w^2 = 1 + \sum_{i=1}^{M}(\frac{\partial \bar{z}}{\partial \bar{x}_i})^2$ and $\mathbf{P}_{i,j} = \frac{\partial^2 \bar{z}}{\partial \bar{x}_i \partial \bar{x}_j}$. We can infer the shape of the surface in the vicinity of the point q by examining the sign of $\kappa(\boldsymbol{\lambda})$. In $\mathcal{R}^3$ (i.e. a two-parameter L-hypersurface) a point of a surface is called

i. Elliptic if $det(\mathbf{A}_q) > 0$ (see Fig. 4.3).

ii. Hyperbolic if if $det(\mathbf{A}_q) < 0$.

iii. Parabolic if $det(\mathbf{A}_q) = 0$, with $\mathbf{A}_q \neq \mathbf{0}$.

iv. Planar if $\mathbf{A}_q = \mathbf{0}$.

At an elliptic point the Gaussian curvature is positive. The surface looks like a bowl (or inverted bowl) and all curves passing through q have their normal vectors pointing towards the same side of the tangent plane. At a hyperbolic point, the Gaussian curvature is negative and therefore there are curves through q whose normal vectors at q point toward any of the sides of the tangent plane. The surface looks like a saddle. At a parabolic point the surface is planar in one direction: an example is a point on a cylinder. Finally, at a planar

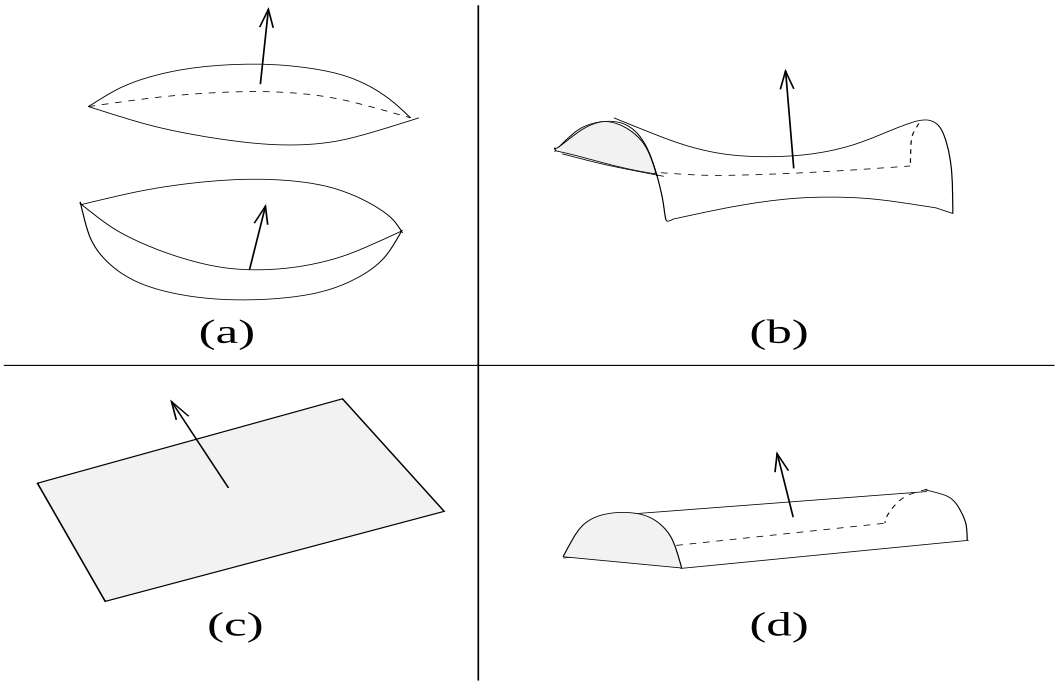
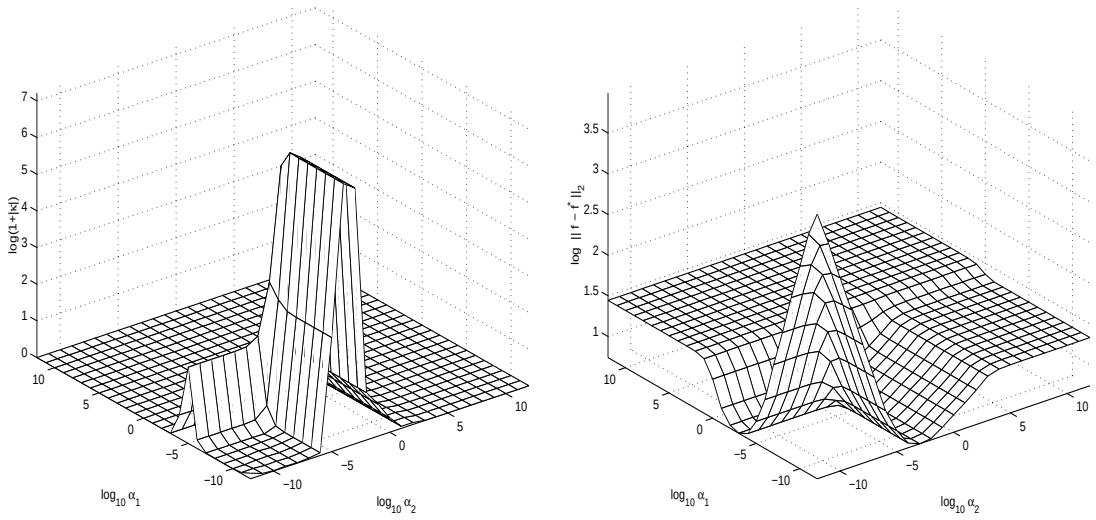


Figure 4.3: The shape of a surface in the vicinity of (a) Elliptic (b) Hyperbolic (c) Planar (d) Parabolic points.

point the surface looks like a plane. A moment's thought reveals that in the vicinity of a good regularization parameter vector $\boldsymbol{\alpha}$, the L-hypersurface has to be elliptic. That is, the curvature must be greater than zero. Negative curvatures are not interesting because at a negative curvature point cutting the surface by the $z - x_1$ plane produces a convex/concave curve indicating that λ_1 is in the vicinity of the correct corner while cutting the surface by the $z - x_2$ plane produces a concave/convex curve which signals that λ_2 is close to one of the false corners or the knees.

Figure 4.4 (a) shows the Gaussian curvature of the L-hypersurface in Fig. 4.1 (a) and Fig. 4.4 (b) is the corresponding estimation error, $\|\mathbf{f} - \mathbf{f}^*(\boldsymbol{\lambda})\|_2$, surface. We immediately



(a) (b)

Figure 4.4: (a) Gaussian curvature of the L-surface in Fig. 4.1 (b) corresponding estimation error surface.

observe that the Gaussian curvature plot presents a remarkably clear picture. The points on the curvature plot where the curvature achieves a local maximum seems to track the local minimum of the estimation error surface. Good regularization parameters can be located very easily by inspection of Fig. 4.4 (a). We note that the form of the Gaussian curvature surface is not specific to the particular problem at hand but is quite general. Similar curvature plots can be obtained for different regularization schemes/problems.

4.3.2 Gaussian Curvature: Computational Details

The Gaussian curvature of the L-hypersurface can be computed in a straightforward but tedious way from (4.14). One important thing to notice in (4.14) is that the formula (4.14) requires $\frac{\partial \bar{z}}{\partial \bar{x}_i}, \quad i = 1, \ldots, M$ and $\frac{\partial^2 \bar{z}}{\partial \bar{x}_i \partial \bar{x}_j}, \quad i, j = 1, \ldots, M$ which are usually unavailable. On

the other hand $\frac{\partial \bar{z}}{\partial \lambda_i}$ and $\frac{\partial^2 \bar{z}}{\partial \lambda_i \partial \lambda_j}$ can be easily obtained and can be used to compute the partial derivatives of $\bar{z}$ with respect to $\bar{x}_i$ through a linear transformation. Details are provided in Appendix A. From the formulas in Appendix A, it can be easily seen that the following quantities are needed to compute the numerical value of the Gaussian curvature at a point $\boldsymbol{\lambda}$:

$$\bar{z}, \bar{x}_i, \frac{\partial \bar{z}}{\partial \lambda_j}, \frac{\partial \bar{x}_i}{\partial \lambda_j}, \frac{\partial^2 \bar{z}}{\partial \lambda_j \partial \lambda_k}, \frac{\partial^2 \bar{x}_i}{\partial \lambda_j \partial \lambda_k}; \quad i, j, k = 1, \ldots, M. \tag{4.15}$$

The quantities $\bar{z}$ and $\bar{x}_i$ can be computed from the knowledge of $\mathbf{f}^*$ alone. On the other hand, each of the remaining partials requires the solution of a matrix-vector equation which is of the same size as the original inverse problem. Thus, for each point $\boldsymbol{\lambda}$ on the curvature surface, in addition to computing $\mathbf{f}^*$, we need to solve $M + \frac{M(M+1)}{2}$ linear equations. Considering that the curvature must be estimated for many regularization parameter vectors $\boldsymbol{\alpha}$ to obtain a reliable estimate of the best parameters we realize that the L-hypersurface method adds a significant computational burden to the already complicated inverse problem. Therefore, it is of great interest to develop methods that reduces the computational cost of the L-hypersurface method.

4.4 Numerical Experiments

In this section, we will evaluate the performance of the L-hypersurface method by

- comparing the efficiency of the L-hypersurface method to that of the multi-parameter GCV [85]

- implementing the L-hypersurface method in conjunction with the multiscale IRR algorithm presented in the previous Chapter.

Results, summarized below, indicate that the L-hypersurface method offers at least comparable, if not better, performance as compared to the GCV and it works quite well with the multiscale IRR algorithm in practical situations.

4.4.1 Numerical Results: Efficiency of the L-hypersurface Method

In this section, the performance of the L-hypersurface method for choosing two regularization parameters, $\boldsymbol{\lambda} = [\lambda_1, \lambda_2]$, will be evaluated and compared with the GCV method. The measure of performance we employ is the the *classical efficiency* E [30]

$$E = \frac{\|\mathbf{f}^*(\boldsymbol{\lambda}_{best}) - \mathbf{f}\|_2}{\|\mathbf{f}^*(\boldsymbol{\lambda}_L) - \mathbf{f}\|_2}, \tag{4.16}$$

where $\boldsymbol{\lambda}_L$ is obtained by either the L-hypersurface or the GCV method and the vector of best regularization parameters $\boldsymbol{\lambda}_{best}$ are those minimizing the l_2 norm of the error between the original and the estimated objects, $\|\mathbf{f}^*(\boldsymbol{\lambda}) - \mathbf{f}\|_2$. Since $\|\mathbf{f}^*(\boldsymbol{\lambda}_{best}) - \mathbf{f}\|_2 \leq \|\mathbf{f}^*(\boldsymbol{\lambda}_L) - \mathbf{f}\|_2$ by definition of λ_{best}, the efficiency values are spread over an interval from zero to one, with a value close to one indicating good performance and a value close to zero indicating poor performance. The global performance of the L-hypersurface and GCV methods were evaluated by observing the values of E obtained over 100 experiments which differed only by simulated noise. These 100 efficiency values were then used to create a histogram by partitioning the range $[0, 1]$ into 10 intervals of equal length. We believe that such a histogram is a good summary of the performance of a parameter selection method.

The performance of both methods were evaluated in a signal deconvolution experiment where the signal of interest is the 128-point "Blocks" sequence extracted from Donoho's Wavelab software package [37]. It was degraded by a Gaussian convolutional kernel of variance $\sigma_h = 4.0$. Zero mean white Gaussian noise was added to the degraded signal to obtain SNR's of 40dB and 100dB. The inversion scenario in (4.6) was implemented with $\mathbf{R}_1 = \mathbf{I}$, $\mathbf{R}_2 = \mathbf{D}^{(1)}$ and $\phi_i(t) = t^2$, $i = 1, 2$ where $\mathbf{D}^{(1)}$ is a matrix representing the discrete approximation to differentiation.

Figure 4.5(a)-(b) and (c) display the L-hypersurface, the GCV function and the MSE surface obtained for a particular noise realization for the case of SNR = 40dB, respectively. Note that the flat appearance of the GCV function contrasts heavily with the nicely defined contours of the L-hypersurface in Fig. 4.5 (a). Notice that the ridges of the L-hypersurface in Fig. 4.5 (a) correlate well with the local minimum of the MSE surface in Fig. 4.5 (c). Such behavior is not specific to this experiment but rather general. We performed numerous experiments by changing the signal, degradation or noise level and obtained exactly the same results.

In Fig. 4.5 (d) and (e), we plot the histograms of E values obtained on 100 experiments for the L-hypersurface and the GCV methods for SNR = 40dB case, respectively. At each experiment, $\boldsymbol{\lambda}$ maximizing the curvature of the L-hypersurface and the $\boldsymbol{\lambda}$ minimizing the GCV function were found by exhaustively searching an interval $\boldsymbol{\lambda} \in [10^{-8}, 10^{0}] \times [10^{-8}, 10^{0}]$. From Fig. 4.5 (d), we observe that a large number of efficiency values for the GCV method lie in the close vicinity of one but the histogram indicates that these good efficiency values were

accompanied by a rather long tail in the distribution extending well into the territory where the performance is quite poor. As a conclusion we can say that although GCV is quite good in general it has a tendency to to produce unacceptable performance occasionally. Examining Fig. 4.5 (e) reveals that the L-hypersurface method is considerably more robust than GCV and all of the observed values of the efficiency fall in the range $0.9 \sim 1.0$.

Figure 4.5 (f) and (g) display the respective histogram of efficiency values for the L-hypersurface and the GCV methods for SNR 100dB. As observed from Fig. 4.5 (g), the L-hypersurface, in this case, performs almost perfectly in spite of the results in the literature saying that the performance gets worse as SNR increases. Finally, GCV performs almost the same as it did for SNR 40dB case.

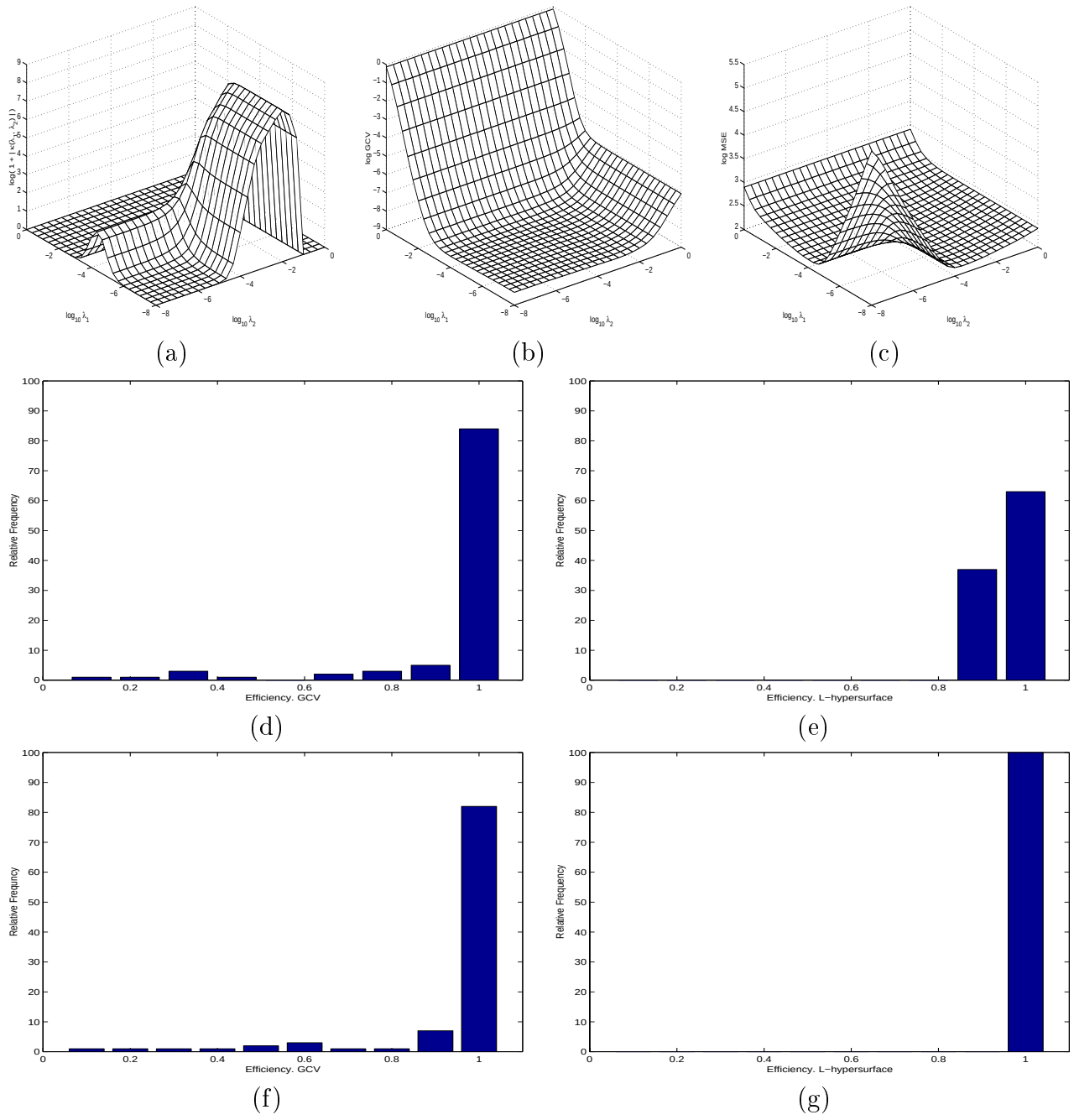


Figure 4.5: A typical (a) L-hypersurface, (b) GCV function, (c) mean square error, $\|\mathbf{f} - \mathbf{f}^*(\boldsymbol{\lambda})\|_2$ for the model experiment. Histogram of efficiency values obtained over 100 experiments for (d) GCV at 40dB SNR (e) L-hypersurface at 40dB SNR (f) GCV at 100dB SNR (g) L-hypersurface at 100dB.

4.4.2 Combining the L-hypersurface with the Multiscale Image Restoration Scheme: Numerical Results

In this section, we illustrate the performance of our proposed multiscale image restoration algorithm in a practical situation where the relevant regularization parameters were selected by the L-hypersurface method. We used the routines in Donoho's Wavelab toolbox [37] for the computation of the forward and inverse wavelet transforms with Daubechie's eight tap most symmetrical wavelets [28]. In all cases below, we limited the number of levels of wavelet decomposition to 3.

4.4.2.1 Example 1: Restoration of Mandrill Image

In our first example, we used a Gaussian convolutional kernel, $h(x,y) = \frac{1}{4\sigma_x\sigma_y}\exp\{-\frac{x^2+y^2}{2\sigma_x\sigma_y}\}$, with $\sigma_x = \sigma_y = 2.0$ to blur the 256×256 Mandrill image. Zero mean white Gaussian noise was added to set the SNR to 30dB. In Fig. 4.6 (a)-(b) we display the original and the blurred, noisy images.

We restored the degraded Mandrill image using our proposed multiscale regularization scheme, and the Constrained Least Squares (CLS) algorithm with a 2-D Laplacian regularizer [2]. The relevant regularization parameters were determined using the L-curve or the L-hypersurface method. In Fig. 4.6(c)-(e) we display the restored Mandrill images corresponding to the CLS, and the multiscale algorithm. For our multiscale image restoration method we computed three restorations, displayed in Fig. 4.6(d)-(f), according to the Model 1 and Model 2 regularization schemes described in Section 3. Figure 4.6 shows that our

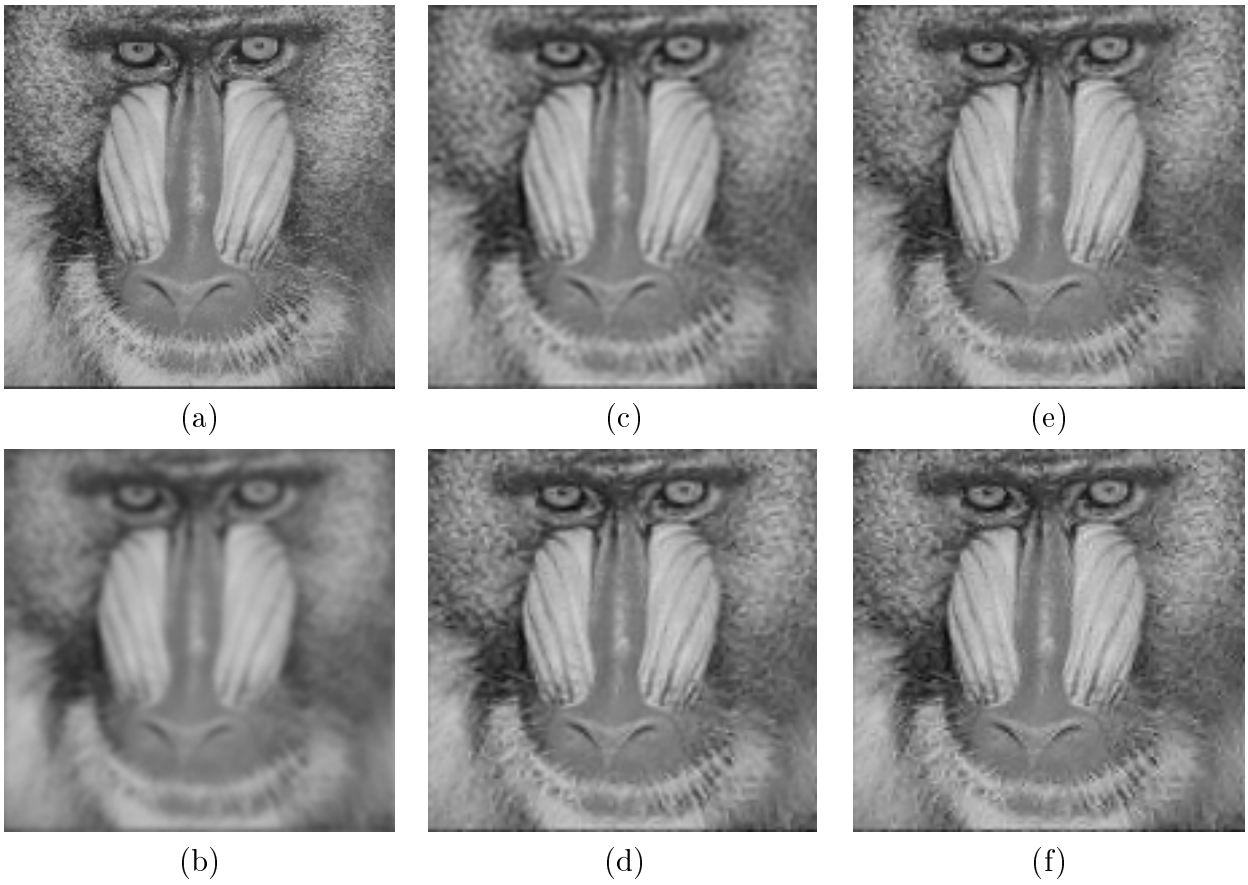


Figure 4.6: From left to right: (a) Original, (b) blurred and (c) restored by CLS (d) restored by Model 1 regularization scheme (e) restored by Model 2 regularization scheme (f) restored by Model 1 regularization scheme with optimal p and λ values. The regularization parameters were obtained by the L-hypersurface method.

algorithm produce restored images visually superior to the CLS algorithm. We also observe that the images restored by our algorithm are a little sharper than the image restored by the CLS algorithm and that the texture-like regions abundant in the Mandrill image (eg. the hairs around the mouth of the Mandrill) are better recovered by our algorithm.

For the Model 1 restoration in Fig. 4.6(d) the L-hypersurface was used to determine two parameters, λ_1 and λ_2 corresponding to the coarsest scale scaling coefficients and the wavelet

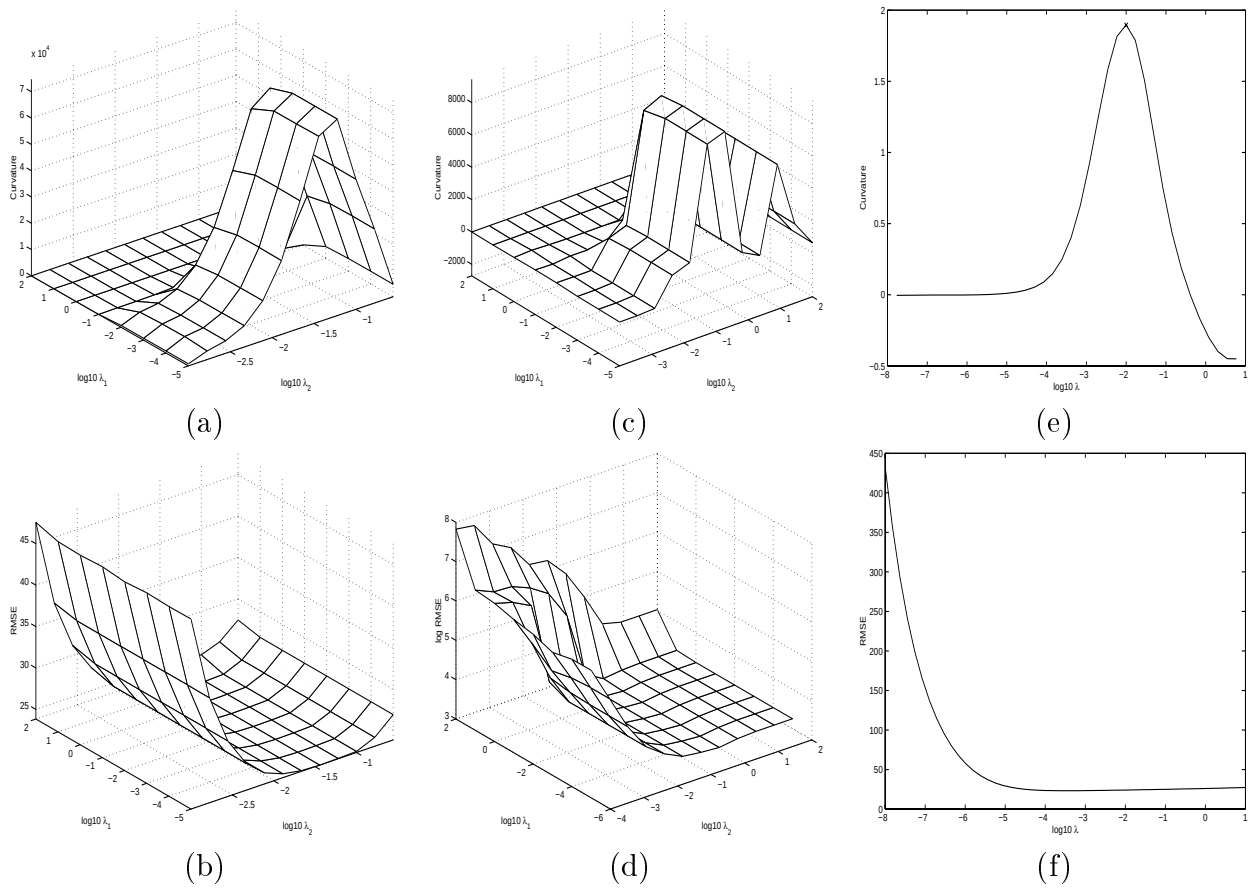
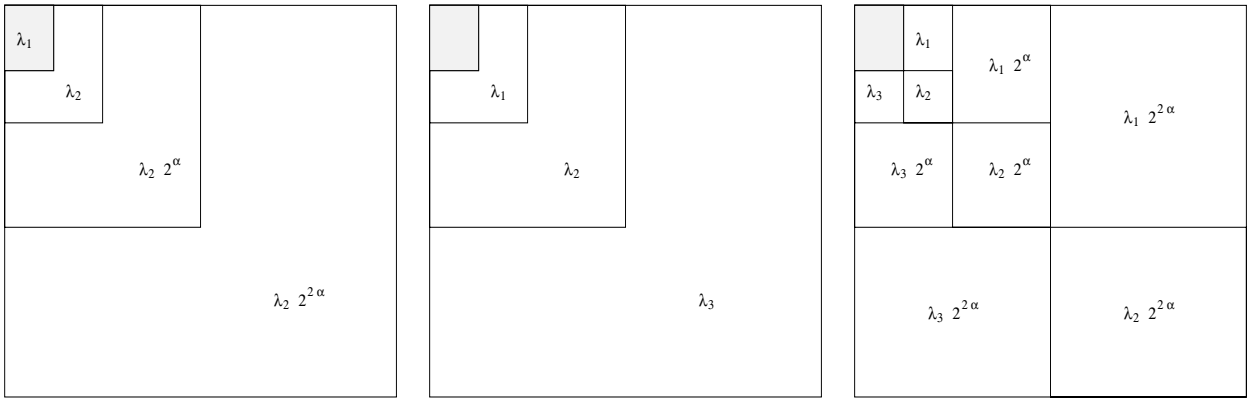


Figure 4.7: (a) Curvature of the L-hypersurface for Model 1 with $\alpha = 1.2$ and $p = 1.0$ and (b) the corresponding RMSE surface (c) Curvature of the L-hypersurface for Model 1 with $\alpha = 0.67$ and $p = 0.7280$ (optimal parameters) and (d) the corresponding RMSE surface (e) L-curve for the CLS algorithm and (f) corresponding RMSE curve.

coefficients as sen in Fig. 4.8 (a). In this case, the curvature of the L-hypersurface is a 2-D function of the regularization parameters as seen in Fig. 4.7(a). Also shown in Fig. 4.7(b) is a plot of RMSE, $\sqrt{\frac{1}{N^2}\|\mathbf{f} - \mathbf{f}^*(\boldsymbol{\lambda})\|_2^2}$, as a function of these regularization parameters. Examining these plots shows that the curvature surface has a distinct extended maxima along which the norm of the error is very close to being a minimum. Thus, we see that the restoration algorithm is not overly sensitive to the scaling coefficient regularization parameter and



(a) (b) (c)

Figure 4.8: (a) Model 1 (b) Model 2 (c) Model 3 regularization schemes as used in our experiments. In each Model, the required regularization parameters, λ_j, are selected by the L-hypersurface method. In Model 1 and 3 α is set to 1.2 *a priori*.

locating the correct regularization parameter for the wavelet coefficients is more important.

In the Model 2 restoration in Fig. 4.6(e), each scale in the wavelet domain is assigned a different regularization parameter as seen in Fig. 4.8 (b). Based on the L-hypersurface obtained for the Model 1 restoration in Fig. 4.7(a), we set the scaling coefficient regularization parameter to 10^{-5}. Figure 4.9(a)-(c) shows the curvature of the L-hypersurface obtained for this experiment. Since in this case the curvature is a 3-D function (one parameter for each wavelet scale), each of the 2-D plots in Fig. 4.9(a)-(c) is actually a slice of the curvature hypersurface with the regularization parameter corresponding to the coarsest scale being constant. Again, the maxima of the curvature of the L-hypersurface track well the minima of the the RMSE surface so that we are close to the "optimal" regularization parameters. Higher dimensional L-hypersurfaces can be obtained by just augmenting the parameter set.

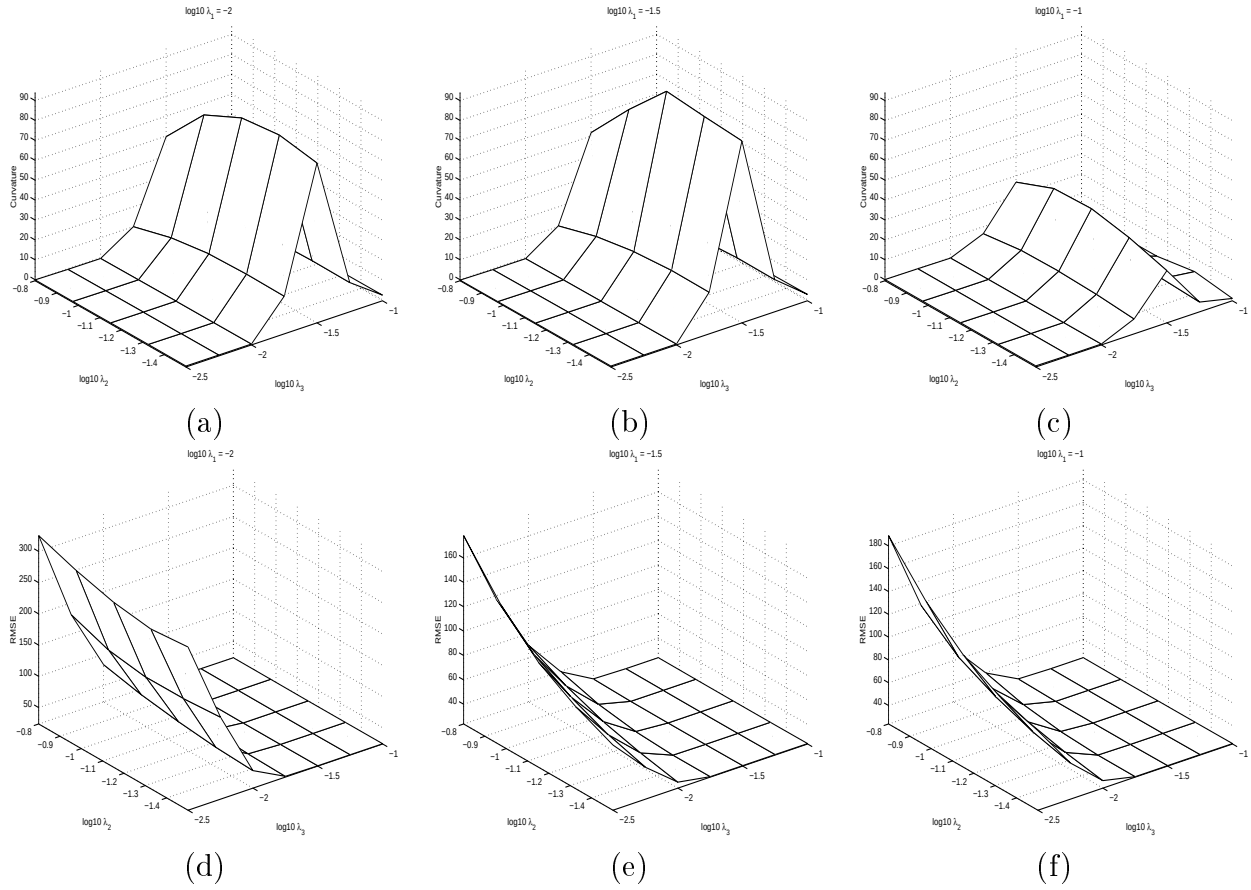


Figure 4.9: (a)-(c) Curvature of the L-hypersurface and (d)-(f) RMSE plots for the mandrill experiment.

We see little difference either in terms of the error norm or in terms of visual quality between the Model 1 and Model 2 restorations in Fig. 4.6(d) and (e). This example verifies the primary assumption of Model 1 scheme where it was assumed that the variance of the wavelet coefficients decrease uniformly across scales according to an exponential law.

Finally, in Fig. 4.6(f) we display the Model 1 restoration corresponding to an idealized case where the parameters α and p were estimated directly from the original image. Clearly,

this is not a realistic situation since in practice the original image is not available. Nonetheless, this example is interesting since it gives us an idea about how much improvement can be expected when using the optimal α and p values as opposed to fixed *a priori* choices $\alpha = 1.2$ and $p = 1.0$. The optimal p was estimated by using the method proposed in [75] and was found to be $p_{opt} = 0.7280$. The exponential parameter α_{opt} was estimated by computing the slope of the line fitted to the $\log \kappa_j$ for $j = j_0, \ldots, J-1$. It was found to be $\alpha_{opt} = 0.6117$. Since p_{opt} yields a non-convex optimization task, we computed the restorations for this case in 2 stages. The first stage starts with computing the restoration for $p = 1.0$, which is unique and then the restored image for $p = 1.0$ is fed as the starting point to the restoration algorithm with $p = 0.7280$. There is no guarantee that the restored image for p_{opt} corresponds to the global minimum of the cost function, nevertheless we obtained good results with this scheme. Figure 4.7(c)-(d) shows the L-surface and the RMSE surface for p_{opt}, respectively. Figure 4.6(f) is the restoration obtained for this case. Finally, comparison of Fig. 4.6(f) with Fig. 4.6(d) reveals that there is visually little difference between the restored images corresponding to p_{opt} and $p = 1.0$ cases. This example illustrates that using $p < 1$ does not yield a significant improvement in the performance of the multiscale algorithm.

4.4.2.2 Example 2: Restoration of Bridge Image

In our second example, we first blurred the original Bridge image in Fig. 4.10(a) with a 9×9 uniform motion blur and added white Gaussian noise to the degraded image to set the SNR at 40dB. The blurred image obtained by this way is shown in Fig. 4.10(b). We computed two

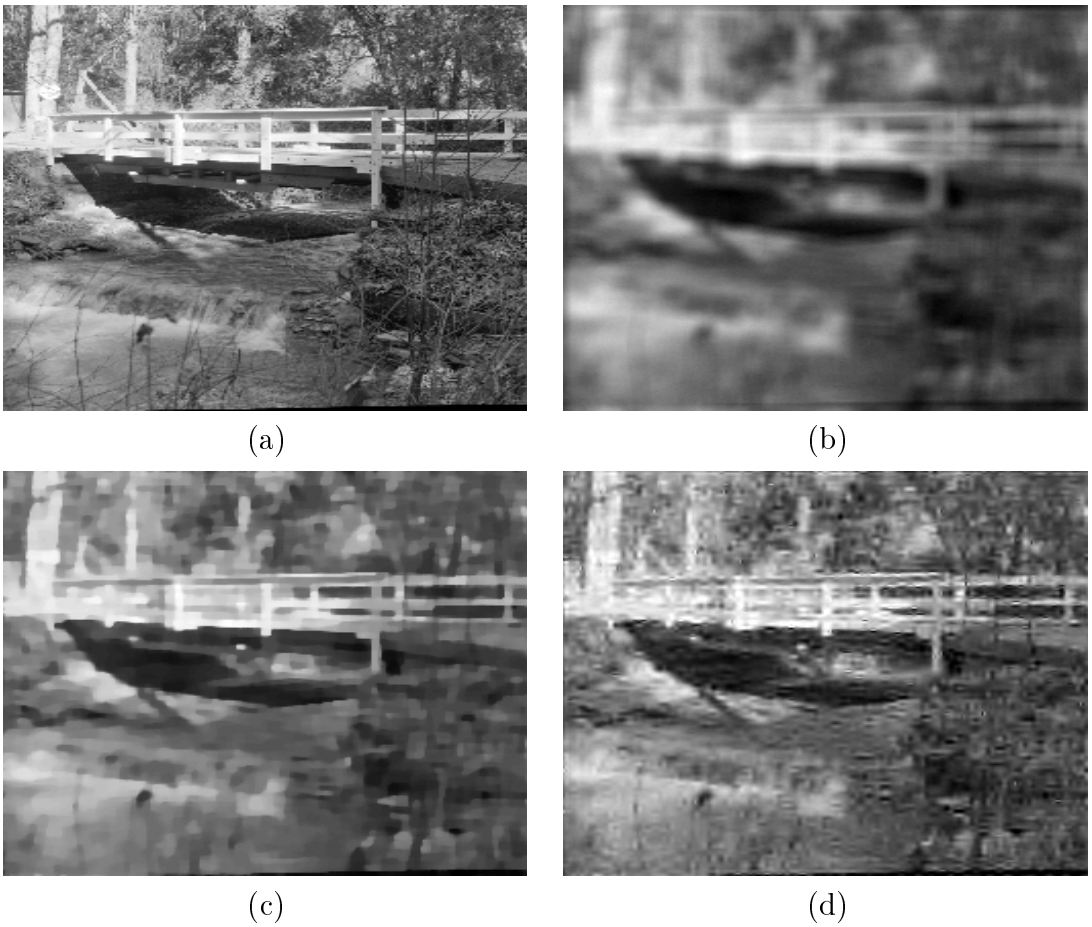


Figure 4.10: (a) Original bridge image (b) Blurred at SNR = 40dB (c) Restored by the TV algorithm (d) Restored by the Model 1 scheme

restorations corresponding to the Total Variation (TV) and our multiscale image restoration schemes. The TV algorithm is a special case of the generalized image restoration scheme in (4.6) for $j = 1$ in which taking $p = 1$ and $\mathbf{R}_1\mathbf{f} = \nabla\mathbf{f}$ gives the cost function for the TV algorithm. The TV algorithm is a member of the recently introduced edge-preserving image restoration algorithms whose penalty functions apply less regularization as the magnitude of the gradient increases. In this way, the edges in the image which are associated with large

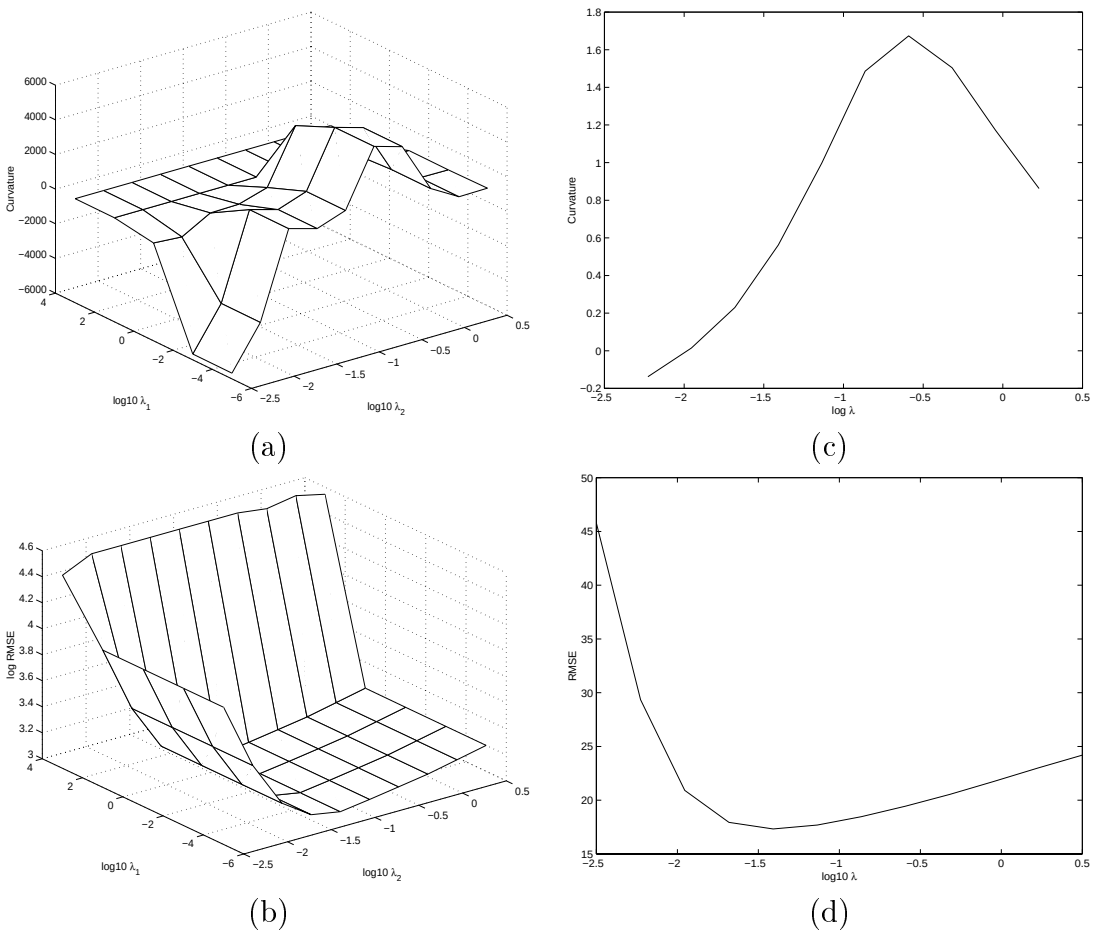


Figure 4.11: (a) Curvature of the L-hypersurface for the multiscale algorithm. (c) Curvature of the L-curve for the TV algorithm. (b)-(d) Corresponding RMSE plots.

gradients are well-preserved in the restoration process while the additive noise is suppressed. We used the "lagged diffusivity" algorithm due to Vogel and Oman [98] to compute the restored image and chose the regularization parameter by the L-curve method as seen in Fig. 4.11(c). The restoration obtained in this way is displayed in Fig. 4.10(c).

For our multiscale algorithm, we applied the Model 1 regularization scheme with the L-hypersurface choice of regularization parameters. As in the previous example, we determined

2 regularization parameters corresponding to the scaling and the wavelet coefficients (Fig. 4.8 (a)), respectively. The L-hypersurface and the corresponding RMSE plot obtained for this experiment are shown in Fig. 4.11(a)-(b). Figure 4.10(d) shows the Model 1 restoration.

We observe that the restored images in Fig. 4.10(c)-(d) exhibit vastly different visual characteristics. The TV algorithm fails to recover many of the small features in the image and produces an overly homogenized restoration resembling an "oil painting" of the original scene. The multiscale algorithm is able to reproduce finer detail thereby yielding a more visually appealing restoration. Note that we used the same value $\alpha = 1.2$ in both Mandrill and Bridge examples regardless of the image considered.

4.4.2.3 Example 3: Restoration of Stripes Image

In our third example, we demonstrate the orientation adaptive nature of our approach. In Fig. 4.12 (a), we display an artificial 32×32 image which has significant structure in the horizontal direction, but little in the vertical and diagonal directions. This image was blurred by a Gaussian convolutional kernel with $\sigma_x = \sigma_y = 1$, and zero mean white Gaussian noise was added to set the SNR at 30dB. Because of the large differences between the structure in the horizontal and vertical directions, an ideal image restoration algorithm should use different regularization parameters for vertical, horizontal and diagonal directions. With this in mind, in Fig. 4.12(c)-(d) we display the restorations obtained using Model 1 and Model 3 schemes which require three regularization parameters, λ_1, λ_2 and λ_3, as displayed in Fig. 4.8(b)-(c), respectively. The L-hypersurface was employed to determine

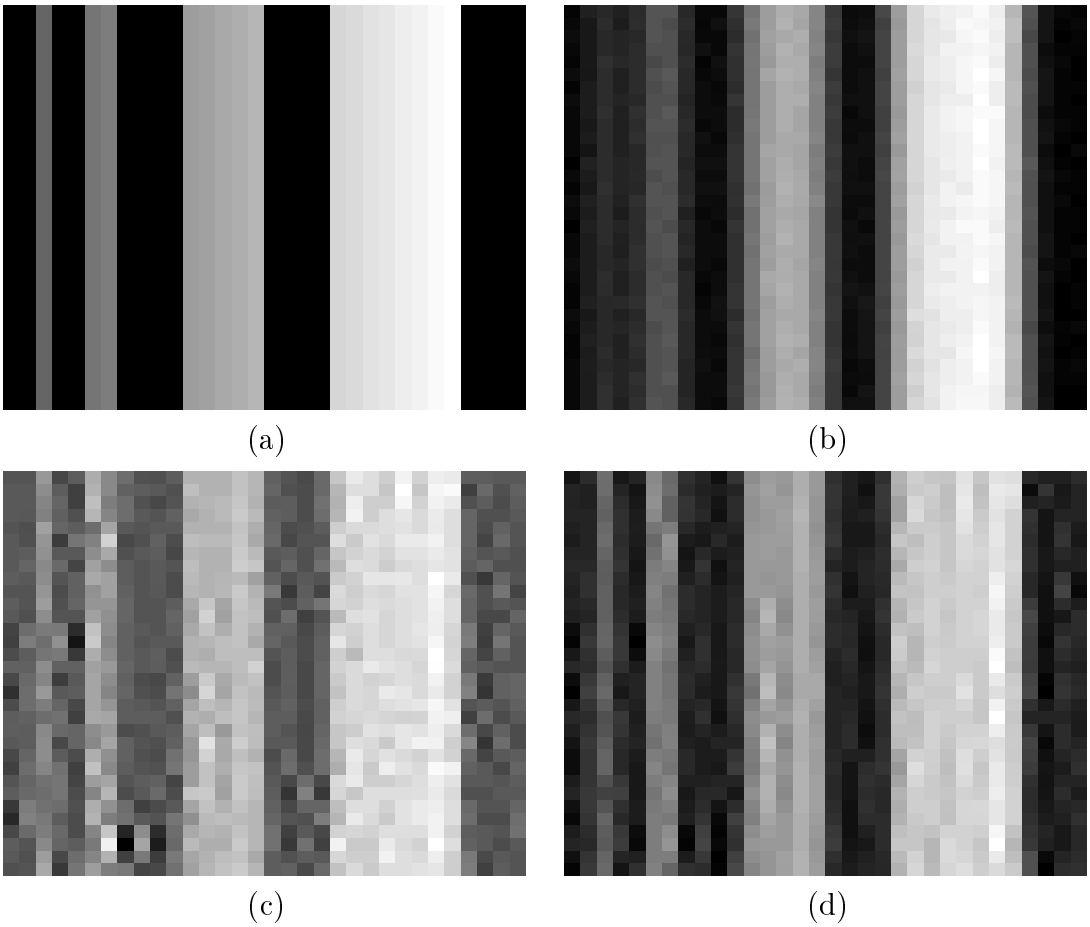


Figure 4.12: (a) Original image. (b) Blurred image, 30dB SNR. (c) Restored by the proposed algorithm with Model 2 (scale adaptive) regularization. (d) Restored by the proposed algorithm with Model 3 (orientation adaptive) regularization.

the required regularization parameters. For both Model 1 and Model 3 schemes we set the scaling coefficient regularization parameter to 10^{-5}. For the Model 3 restoration, the regularization parameters obtained for the vertical and diagonal orientations (in which the image is constant) were approximately two orders of magnitude larger than the regularization parameter obtained for the horizontal orientation. It is clear from Fig. 4.12(c)-(d) that the orientation adaptive algorithm produces a much better restoration than the scale adaptive

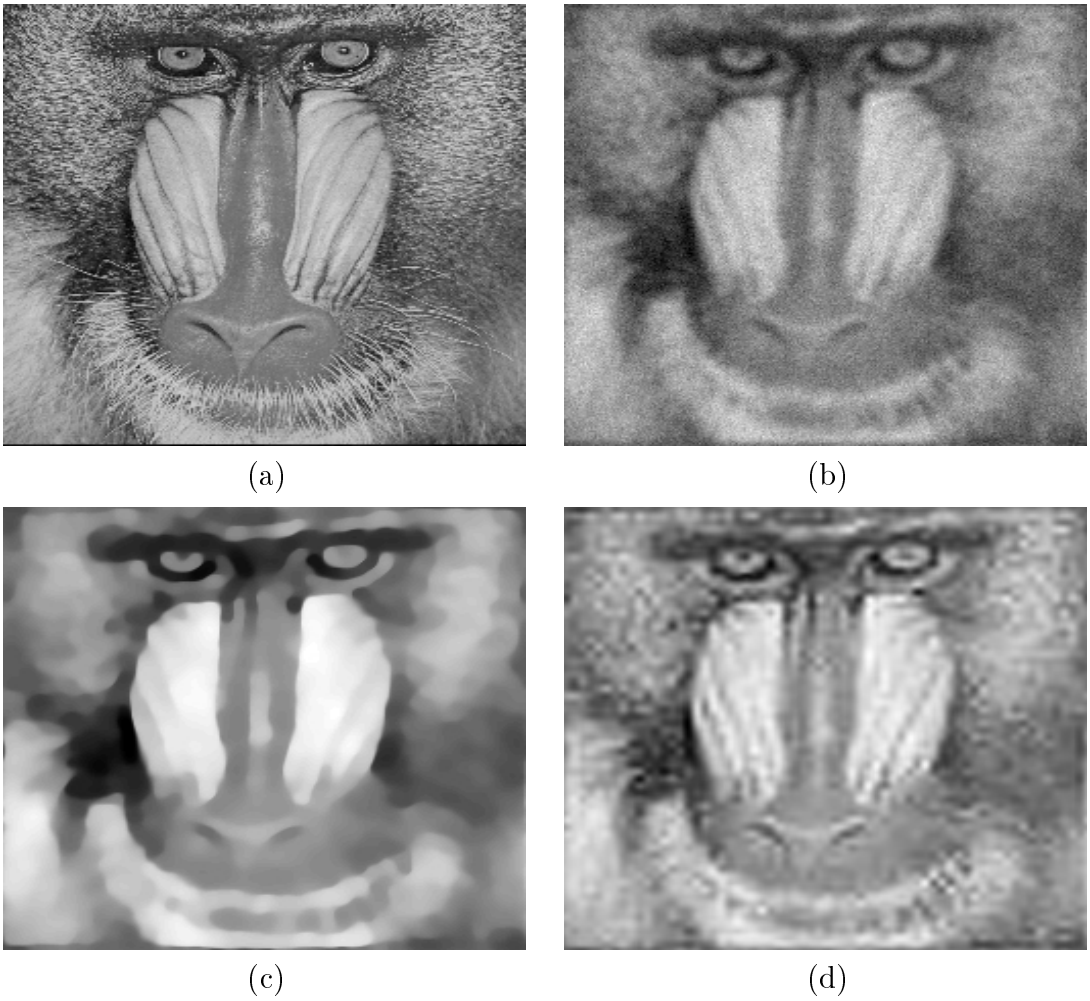


Figure 4.13: (a) Original Mandrill image (b) Blurred at SNR = 10dB (c) Restored by the TV algorithm (d) Restored by the Model 1 scheme.

algorithm.

4.4.2.4 Example 4: Restoration of the Mandrill Image; Low SNR Case

In this example, we explore the low SNR behavior of our multiscale algorithm as compared to the TV algorithm. The original Mandrill image as shown in Fig. 4.13 (a) was blurred with the Gaussian convolutional kernel from Example 1 but this time the noise level

was adjusted so that SNR is 10dB. The blurred, noisy image is shown in Fig. 4.13 (b). We restored the degraded Mandrill image using our multiscale regularization algorithm with Model 1 regularization scheme (see Fig. 4.8 (a)). We used $\alpha = 1.2$ for the exponential parameter and determined the λ_1 and λ_2 values by the L-hypersurface method. The curvature of the L-hypersurface and the corresponding RMSE surface obtained for our multiscale regularization scheme are displayed in Fig. 4.14 (a)-(b). For comparison, we also restored the

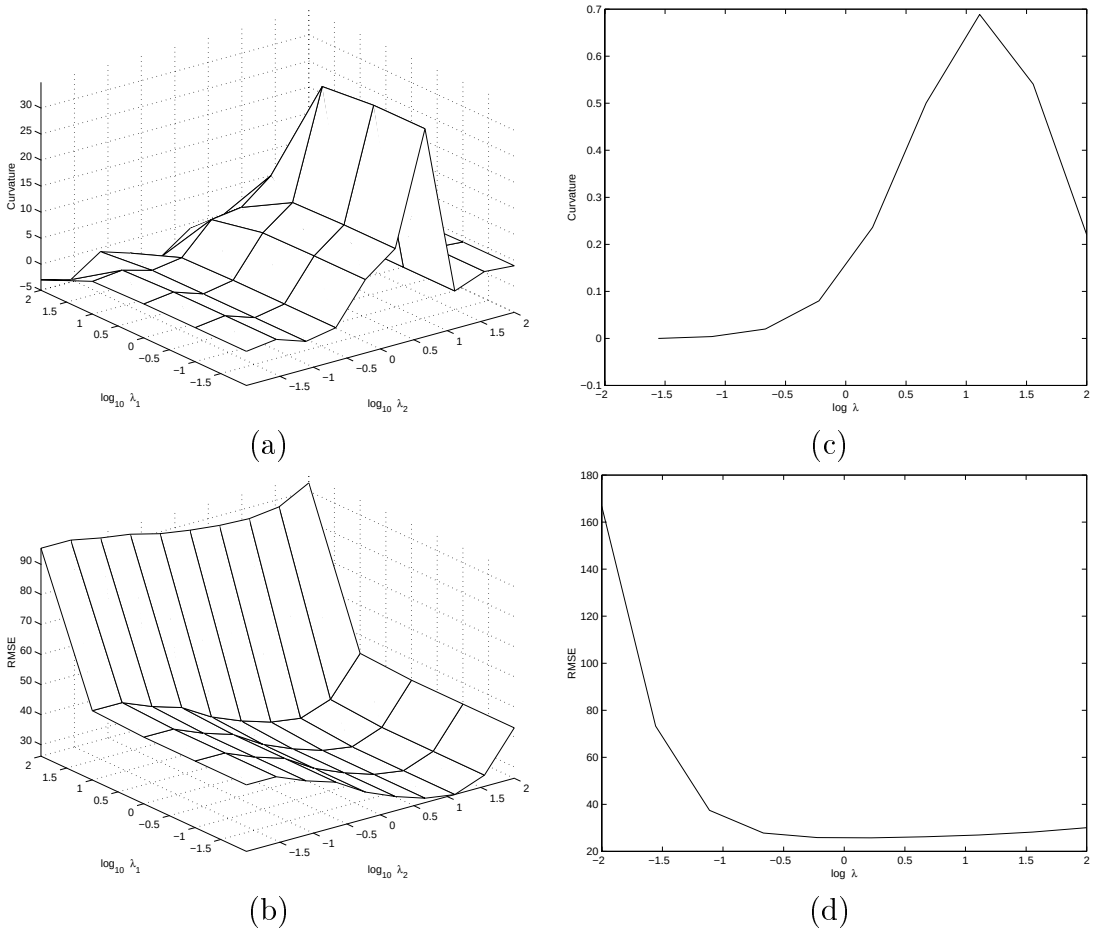


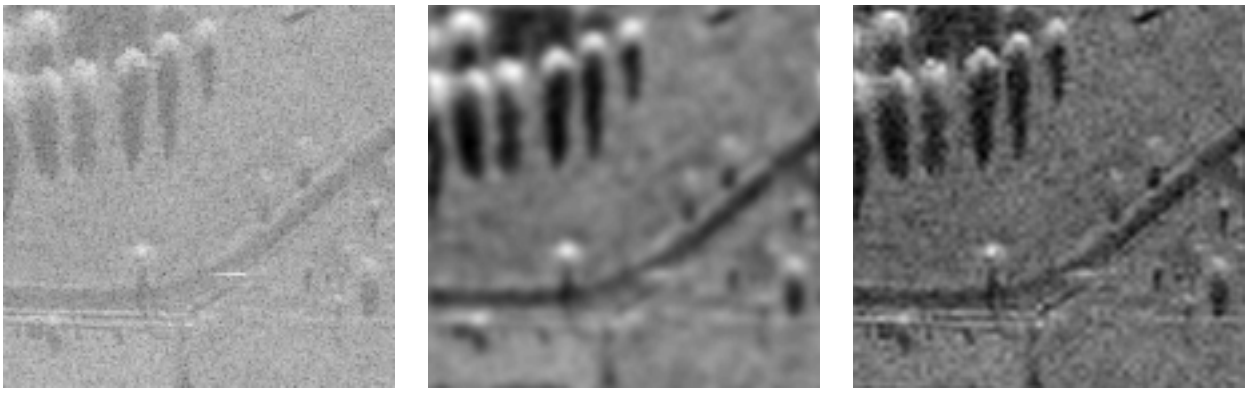
Figure 4.14: (a) Curvature of the L-hypersurface for the multiscale algorithm. (c) Curvature of the L-curve for the TV algorithm. (b)-(d) Corresponding RMSE plots.

degraded Mandrill image using the TV algorithm with the L-curve choice for the regularization parameter. In Fig. 4.14 (c)-(d) we display the curvature and the RMSE plots for the TV algorithm.

Restored images corresponding to our multiscale algorithm and the TV algorithm are shown in Fig. 4.13 (c) and (d), respectively. The extremely low SNR (10dB is regarded as very low in image restoration applications where the convolutional kernel is wide) demands relatively large regularization parameters (in other terms more smoothing) compared to the previous cases. As a result, many of the details present in the original Mandrill image are lost in both restorations. The TV restoration shown in Fig. 4.13 (c) contains large patches of constant magnitude areas and the restored image looks like a piecewise constant approximation to the original image. The multiscale restoration displayed in Fig. 4.13 (d), on the other hand, no longer seems to be edge-preserving. Because of the large penalty applied to the wavelet coefficients, most of the edges and details represented by the wavelet coefficients are lost. The resulting restoration is mostly composed of contributions coming from the scaling coefficients which capture the smooth information in the image. However, the restored image shown in Fig. 4.13 (d) is more realistic than the TV restoration in Fig. 4.13 (c). We note that there is a little more detail in multiscale restoration as compared to the TV restoration. We leave it to the reader to assess the quality of the restored images.

4.4.2.5 Example 5: Restoration of Synthetic Aperture Radar (SAR) Images

In this example, we apply our multiscale IRR algorithm to the problem of the restoration



(a) (b) (c)

Figure 4.15: (a) Original degraded image (b) Restored by the CLS algorithm (c) Restored by our multiscale algorithm with Model 1 regularization scheme.

of SAR images [66, 68]. SAR images are affected by speckle, which appears as a granular signal-dependent noise, whose effect is to degrade the performances of image segmentation and classification algorithms. A variety of techniques are applied for speckle reduction [66]. Here, we will employ our multiscale IRR algorithm to this problem and compare our results with that of the CLS algorithm. Here, we model the observed SAR as the following:

$$g(x,y) = \exp\{h(x,y) * f(x,y)\} \cdot e(x,y) \tag{4.17}$$

where x and y are the spatial coordinates, $*$ denotes convolution and f is the true image, h is a degradation operator, g is the recorded SAR image and e is the multiplicative noise, modeled here as a stationary, uncorrelated Gaussian process independent of f with mean one and variance σ_e^2. We assume here that the degradation operator h is a 2-D Gaussian function as in Example 1, with $\sigma_x = 1.0$ and $\sigma_y = 1.0$ (these values are experimentally determined from the spectral content of the original degraded image). We apply our multiscale IRR algorithm as well as the CLS algorithm to logarithmically transformed data, $\bar{g}(x,y) = \log g(x,y)$, so

that the problem becomes

$$\bar{g}(x,y) = h(x,y) * f(x,y) + e(x,y) \ . \tag{4.18}$$

We restored the degraded images by our multiscale algorithm with Model 1 restoration

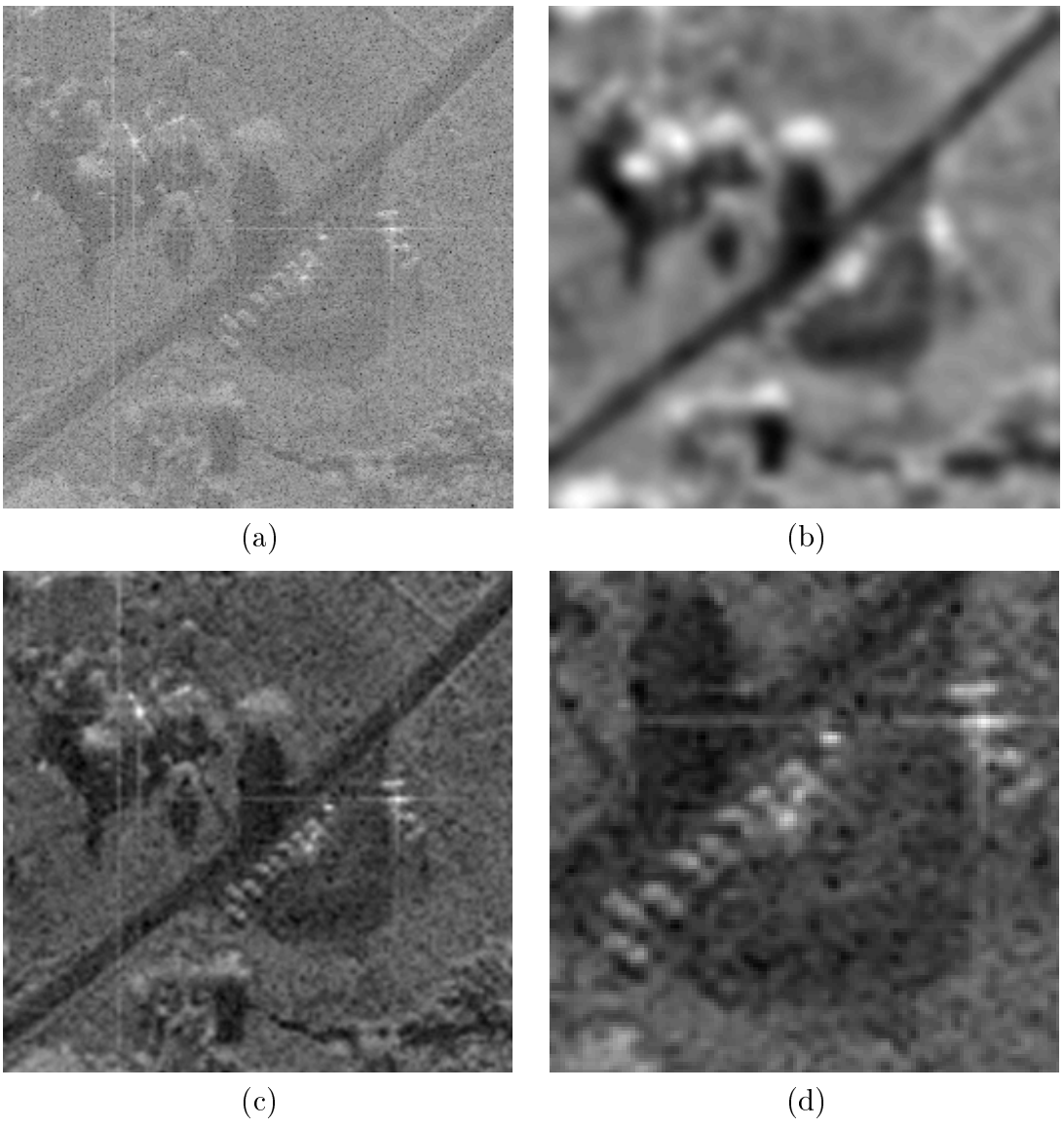


Figure 4.16: (a) Original degraded image (b) Restored by the CLS algorithm (c) Restored by our multiscale algorithm with Model 1 regularization scheme (d) A detail image from (c).

scheme as well as the CLS algorithm with 2-D Laplacian as the regularization operator. Regularization parameters were determined by the L-hypersurface or the L-curve method. In Fig. 4.15 (a) and Fig. 4.16 (a) we display the degraded images. Figure 4.15 (b) and Fig. 4.16 (b) show the restored images corresponding to the CLS algorithm and Figure 4.15 (c) and Fig. 4.16 (c) are the images restored by our multiscale algorithm. As observed, for both images multiscale restorations are much better than the restorations obtained by the CLS algorithm. The CLS algorithm eliminates the noise at the expense of losing many details such as the vehicles present in Fig. 4.16 (a) which are regarded as *targets* in many applications. Also note that the shadows of trees, vehicles and other objects, regarded as features for the purposes of identification, have blurred edges. Our multiscale algorithm is able to preserve the targets and the shadows in both images and at the same time does a remarkable job in eliminating the noise.

Chapter 5

Efficient Selection of Multiple Regularization Parameters in a Generalized L-curve Framework

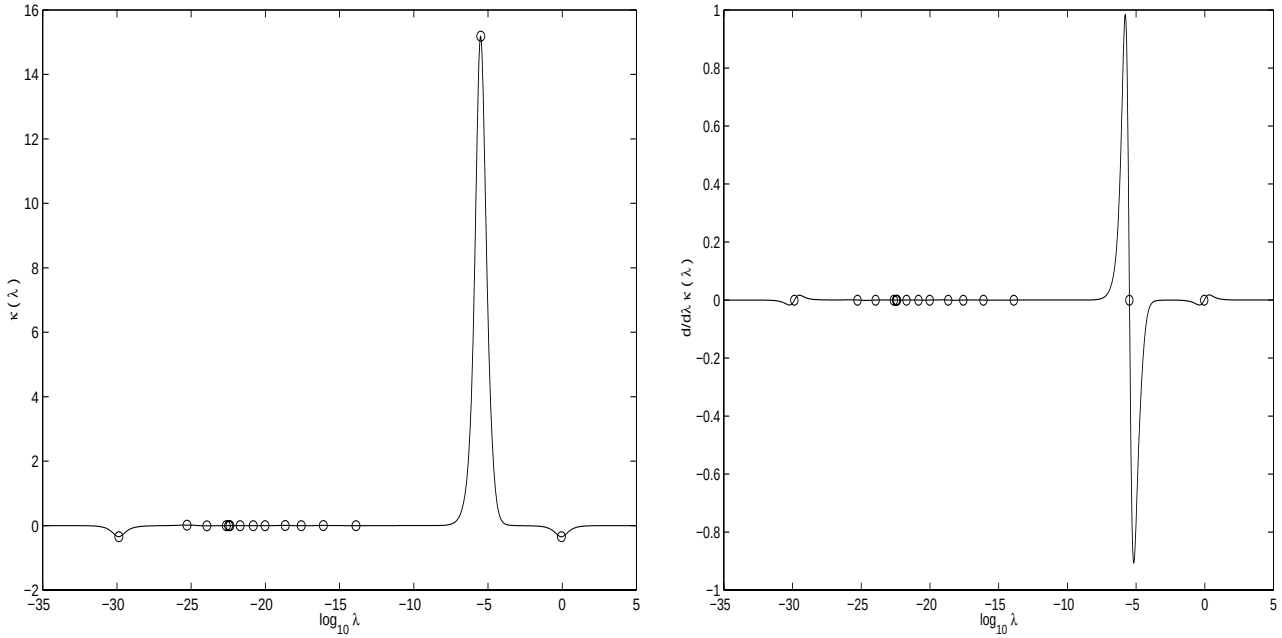
In this section, we consider the selection of multiple regularization parameters in a L-hypersurface framework. Based on the generalized L-curve, we develop what we denote as the *minimum distance function* (MDF) for approximating the regularization parameters corresponding to the generalized corner of the L-curve. It is shown through an L-curve model that the regularization parameters minimizing the MDF essentially maximize the curvature of the L-curve. Furthermore, the MDF approach leads to a simple fixed point iterative algorithm for computing regularization parameters. Examples indicate that the algorithm converges rapidly reducing the cost associated with the implementation of L-hypersurface

method significantly.

5.1 Introduction

Our goal in this section is to propose new methods for the choice of multiple regularization parameters through the use of the L-hypersurface method. Our basic motivation is to decrease the cost associated with the straightforward implementation of the L-hypersurface method. As we have seen in the preceding chapters, the L-hypersurface method is a reliable method with proven performance. It works very well with our multiscale IRR algorithm for which many conventional parameter selection methods are either prohibitively complex or lacks multi-variate extensions. In Chapter 4, we compared the performance of the L-curve method against GCV and supplied results for a number of IRR experiments illustrating good performance. Hence, the first question that comes to mind, "Does the L-hypersurface method really work?" has been answered. Now we tackle the second logical question: "How can we make the L-hypersurface method more efficient?"

The L-curve method has been applied successfully to many practical instances of the inverse problems [6, 14, 42, 56, 58] and is very-well known. There has been some publicized work devoted to increasing its efficiency [22, 67]. In [67], the authors propose approximating the regularized solution $\mathbf{f}^*(\lambda)$ by projecting the original problem onto a smaller dimensional space. Then the regularization parameter is determined by applying the L-curve method to regularize the projected problem. The regularized solution to the projected problem can be computed in a fraction of time, using a fraction of storage. On the other hand, Calvetti *et.*



(a) (b)

Figure 5.1: (a) Curvature, $\kappa(\lambda)$, of a typical L-curve. (b) Derivative of the curvature function, $\frac{d\kappa(\lambda)}{d\lambda}$, in (a). Circles indicate the maxima/minima of $\kappa(\lambda)$ in (a) and the zeros crossings of of $\frac{d\kappa(\lambda)}{d\lambda}$ in (b).

al. [22] propose a method based on computing what what they call an *L-ribbon* that contains the L-curve in its interior. An L-ribbon can be computed fairly inexpensively by partial Lanczos bi-diagonalization of the matrix of the given linear system of equations (see [22] for details). In [58], the authors consider computing the curvature of the L-curve by fitting a smoothing cubic spline to the points points on the L-curve. Since such a curve is smooth and is twice differentiable, the curvature can be computed in a numerically stable way from the fitted curve. They also propose an algorithm to locate the corner of the L-curve by first starting with a few points to the left and right of the corner and successively zooming in to the points where curvature is maximum.

In Chapter 4, we located the generalized corner of the L-hypersurface by doing an exhaustive search over all selected regularization parameters. In general, evaluating points on the curvature function of the L-hypersurface is computationally very demanding and one would prefer using a standard optimization strategy instead of exhaustive search to locate the best regularization parameters. The basic difficulty in this approach is that the curvature function typically possesses many maxima/minima and therefore it is not suitable for standard optimization techniques that might easily be caught in a local maxima/minima. This phenomena is perhaps best illustrated by the plots in Fig. 5.1. Figures 5.1(a) and (b) display the curvature $\kappa(\lambda)$ of a typical L-curve obtained for Tikhonov regularization with identity applied to the *Blocks* problem described in Chapter 2 and the derivative $\frac{d\kappa(\lambda)}{d\lambda}$ of the curvature function. In Fig. 5.1 (a), we marked the maximum/minimum points of the curvature function with a circle. As observed, there are many local maxima/minima (15 to be exact). Even if there were a single maximum, the maximization of the curvature function would be quite difficult because most optimization algorithms require the derivative of the function to be minimized necessitating the derivatives of the third order of the residual and the constraint norms.

Our plan in this work is to approach the problem not in terms of maximizing the curvature function but in terms of minimizing a surrogate function replacing the curvature. We design this surrogate function in such a way that it possesses a minimum point which is close to the corner of the L-hypersurface and additionally it is much easier to optimize. We prove that in the one parameter case, minimization of this surrogate function is essentially

equivalent to the maximization of the curvature function. We develop a simple and efficient fixed point iterative algorithm to compute the regularization parameters that correspond to the minimum of the surrogate function. We demonstrate through numerical examples that although there is little performance loss as compared to the maximization of the curvature, the computational burden is significantly smaller.

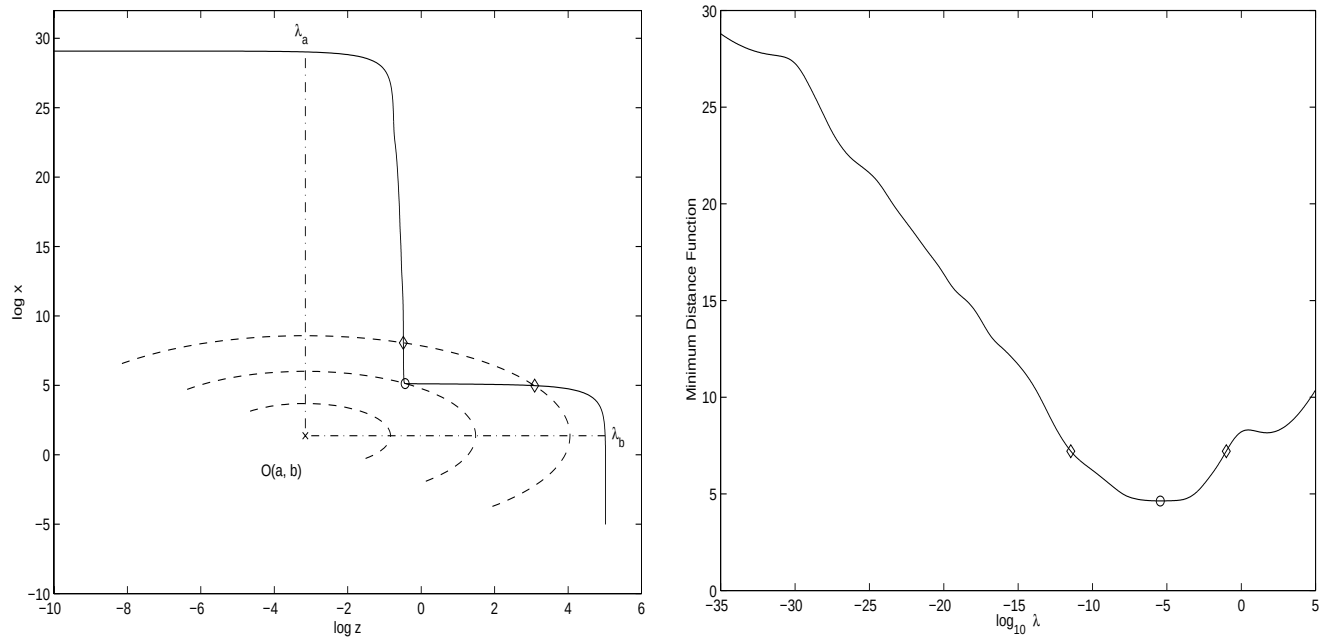
5.2 Problem Setting

In this work, we consider a generalized multiply constrained least squares approach as in Chapter 4:

$$\min_{\mathbf{f}} \left\{ \|\mathbf{g} - \mathbf{H}\mathbf{f}\|_2^2 + \sum_{i=1}^{M} \lambda_i \Phi_i(\mathbf{R}_i\mathbf{f}) \right\}, \quad \mathbf{R}_i \in \mathcal{R}^{m \times n} \ , \tag{5.1}$$

where $\mathbf{R}_i$ are regularization operators and λ_i are the corresponding regularization parameters, $\Phi_i(\mathbf{R}_i\mathbf{f}) = \sum_{j=1}^{m} \phi_i([\mathbf{R}_i\mathbf{f}]_j)$ and the notation $[\mathbf{R}_i\mathbf{f}]_j$ denotes the jth element of the vector $\mathbf{R}_i\mathbf{f}$ and M is the number of constraints. We assume that $\phi_i(t)$ satisfies the conditions presented in Chapter 4.2 so that there exists a half-quadratic algorithm to compute the unique minimum of (5.1). Note that for $M = 1$ and $\phi_1(t) = t^2$ and $\mathbf{R}_1 = \mathbf{I}$, (5.1) reduces to the conventional Tikhonov's regularization with identity. More exotic regularization schemes can be obtained by appropriately structuring $\phi_i(t)$ and $\mathbf{R}_i$. The solution $\mathbf{f}^*$ of (5.1) must satisfy the following equation:

$$\mathbf{K}_{\mathbf{f}^*}\mathbf{f}^* = \mathbf{H}^T\mathbf{g}, \quad \mathbf{H} \in \mathcal{R}^{m \times n} \tag{5.2}$$



(a) (b)

Figure 5.2: (a) The slowly extending bubble centered at $O(a, b)$ first touches the L-curve at the corner. (b) The distance from $O(a, b)$ to the L-curve drawn as a function of λ

where the matrix $\mathbf{K}_{\mathbf{f}^*}$ is given by

$$\mathbf{K}_{\mathbf{f}^*} = \mathbf{H}^T\mathbf{H} + \frac{1}{2}\sum_{i=1}^{M} \lambda_i \mathbf{R}_i^T \mathrm{diag}_{k=1,\ldots,m}\left[\frac{\phi_i'([\mathbf{R}_i\mathbf{f}^*]_k)}{[\mathbf{R}_i\mathbf{f}^*]_k}\right]\mathbf{R}_i. \tag{5.3}$$

5.2.1 An Alternative Definition of the Corner

A major shortcoming of the L-hypersurface method is that direct maximization by evaluating the curvature for a large number of regularization parameters is expensive. Furthermore, use of a conventional optimization technique to locate the maximum curvature point is hampered by the fact that the curvature function usually possesses multiple extrema. Considering these difficulties, we propose replacing the curvature function by a surrogate

function which is far easier to optimize. Our ultimate goal is to choose the surrogate function so that the regularization parameters obtained from the optimization of this function are close to those chosen by the L-hypersurface method. To give a flavor of the simple geometrical ideas behind our approach, we consider a typical L-curve as displayed in Fig. 5.2 (a). We denote the points where extreme solution norm and extreme residual norm regions start by a and b respectively. Formally, the points a and b are defined as

$$a = \psi(\|\mathbf{g} - \mathbf{H}\mathbf{f}^*(\lambda_a)\|_2^2), \quad b = \psi(\Phi[\mathbf{R}\mathbf{f}^*(\lambda_b)]) \tag{5.4}$$

where λ_a is the regularization parameter to the left of the corner where the L-curve becomes approximately horizontal and λ_b is the regularization parameter to the right of the corner where the L-curve becomes approximately vertical (see Fig. 5.2 (a)) and ψ is an appropriate scaling function such as $\psi(t) = \log t$ or $\psi(t) = \sqrt{t}$. For Tikhonov's method we have the following *a priori* estimates: $\lambda_a = \sigma_{\min}^2$, $\lambda_b = \sigma_{\max}^2$ where $\sigma_{\min}$ and $\sigma_{\max}$ are the smallest and the largest singular values of the matrix $\mathbf{H}$ (see discussion in page 49 of [59]).

We define an origin, $O = (a, b)$, and compute the distance from our origin O to the L-curve. Suppose that there is a slowly expanding bubble located exactly at the origin O. From the geometry, it is easy to see that the first point on the L-curve that the bubble touches will be close to the corner of the L-curve. Furthermore, as the bubble continues to expand, the circle describing the boundaries of the bubble intersects the L-curve at exactly two points at the left and right of the corner until the circle reaches extreme residual norm and extreme solution norm regions. The radius of the circle is in fact the value of our distance function. The statements concerning the behavior of the bubble describes our distance function. That

is, the distance function is minimum at a point close to the corner and the function increases as we go away from the origin until we reach extreme residual or signal norm regions. In this way, we have defined a function whose minimum occurs at or near the corner and possesses a single minimum in a wide range of regularization parameters. Hopefully, the newly defined distance function will make our optimization task much easier.

5.2.2 The Minimum Distance Function

We begin with defining our surrogate function for a single regularization parameter (i.e. the L-curve case) prove the associated optimality results and then give a generalized form of the surrogate function for the multi-dimensional case.

Definition 5.2.1 (Minimum Distance Function (MDF)) *: Let $O = (a, b)$ be the coordinates of an appropriate origin. The minimum distance function, $v(\lambda)$, is the distance from the origin O to the point $\boldsymbol{\beta}(\lambda) = (\beta_1(s), \beta_2(s)) = (\psi[z(\lambda)], \psi[x(\lambda)])$ on the L-curve:*

$$v(\lambda) = |\psi[z(\lambda)] - a|^2 + |\psi[x(\lambda)] - b|^2 \tag{5.5}$$

Based on the definition of the MDF, we define the minimum distance point as the following:

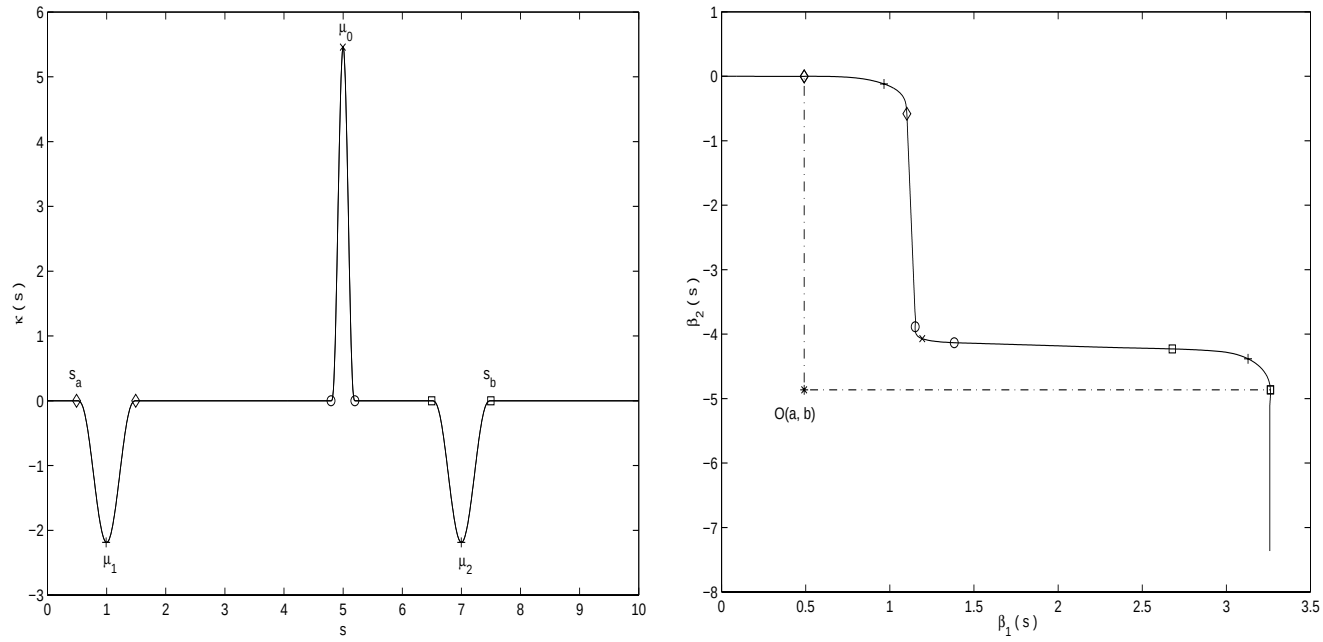
Definition 5.2.2 (Minimum Distance Point (MDP)) *: Let λ_a and λ_b be as defined after (5.4). The minimum distance point is the point where the curvature of the L-curve is positive and $v(\lambda)$ reaches a local minimum:*

$$\lambda^* = \min_{\lambda \in (\lambda_a, \lambda_b)} v(\lambda) \quad \text{or} \quad v(\lambda^*) = 0 \;\; \textit{with} \;\; \lambda \in (\lambda_a, \lambda_b) \; .$$

Next we go on to prove that the MDF function has at least one local minima in $\lambda \in (\lambda_a, \lambda_b)$. However, before proceeding any further we introduce a parameterized model for the L-curve which closely approximates the behavior of a family of L-curves while having a representation simple enough for algebraic manipulations.

5.2.3 A Parametric Model for the L-curve

In many cases, formulas describing the components of the L-curve and its curvature are so complicated that an exact theoretical analysis is intractable. However, the curvature function obtained for a large class of linear inverse problems share a common structure where the curvature is almost zero at all but three different regularization parameters; one of them is the corner where curvature is positive and the other two correspond to the *false corners* or *knees* of the L-curve where the curvature is negative and the regularization parameter is either too small or too large (see Fig. 5.1 for example). Note that this model does not capture most of the of maxima/minima (those that exist in the range $\lambda \in [10^{-25}, 10^{-10}]$) points of the curvature function in Fig. 5.1 (a). However, these maxima/minima points are located in a region where curvature is close to zero and ignoring them is unlikely to cause any problems. Based on these empirical observations, we propose a simple parametric model for the curvature function where the curvature is assumed to be zero at all points except for a narrow region around the "corners". In this model, we represent the curvature function with three little bumps as seen in Fig. 5.3 (a):



(a) (b)

Figure 5.3: (a) Assumed curvature function. (b) the corresponding L-curve. The coordinates of the origin is given by $O = (a, b) = (\beta_1(s_a), \beta_2(s_b))$.

$$\kappa(s) = \theta_0 \frac{1}{\sigma_0} \mathcal{G}\left(\frac{s-\mu_0}{\sigma_0}\right) - \theta_1 \frac{1}{\sigma_1} \mathcal{G}\left(\frac{s-\mu_1}{\sigma_1}\right) - \theta_2 \frac{1}{\sigma_2} \mathcal{G}\left(\frac{s-\mu_2}{\sigma_2}\right) \tag{5.6}$$

where subscripts 1 and 2 represent false corners where the L-curve is concave, subscript 0 represents the desired corner, and μ and σ are parameters adjusting the location and spread of the bumps. The function $\mathcal{G}(s)$ is given by

$$\mathcal{G}(s) = \begin{cases} \frac{35}{32}(1-s^2)^3, & \text{for } |s| \le 1, \\ 0 & \text{for } |s| > 1 \end{cases} \tag{5.7}$$

For convenience, we choose to parameterize the curvature function $\kappa(s)$ in terms of arclength

$$s = \int_{\lambda_0}^{\lambda} \sqrt{|\frac{d\psi(x[u])}{du}|^2 + |\frac{d\psi(z[u])}{du}|^2} du. \tag{5.8}$$

Since $x(\lambda)$ and $z(\lambda)$ are continuous functions of λ and $\frac{ds}{d\lambda}$ never vanishes, there is a one-to-one correspondence between s and λ. That is, given any s we can uniquely determine the

corresponding λ value [81].

There is a unique plane curve (up to a rigid motion) realizing $\kappa(s)$ in (5.6), as its curvature [32]:

$$\boldsymbol{\beta}(s) = \left(\int_0^s \cos\rho(u)du, \int_0^s \sin\rho(u)du\right) \tag{5.9}$$

$$\rho(u) = \int_0^u \kappa(t)dt \tag{5.10}$$

In our case, an explicit analytic formula for $\boldsymbol{\beta}(s)$ cannot be found. However, the equations in (5.10) can be numerically solved from the following differential equation

$$\frac{d\boldsymbol{\beta}(s)}{ds} = (\cos\rho(s), \sin\rho(s))\,. \tag{5.11}$$

Here, $\rho(s) = \int_0^s \kappa(t)dt$ can be analytically computed from (5.6). Fig. 5.3 shows the L-curve obtained from $\kappa(s)$ in (5.6) with $(\mu_0, \mu_1, \mu_2) = (5, 1, 7)$ and $(\sigma_0, \sigma_1, \sigma_2) = (0.2, 0.5, 0.7)$ It is easy to see from this figure that $\theta_0 < \frac{\pi}{2}$ determines the angle between the approximately horizontal and vertical parts of the L-curve to the right and left of the corner. On the other hand, σ_0 is a measure of the width of the crossover region (i.e. the regions separating horizontal and vertical parts). As σ_0 approaches zero, the corner of the L-curve becomes sharper.

5.2.4 Properties of MDF

In this section, we prove, using the L-curve model introduced, that the λ minimizing the MDF essentially maximizes the curvature of the L-curve. To begin with, we define the

tangent vector to the L-curve, $\mathbf{t}(s)$:

$$\mathbf{t}(s) = \frac{d\boldsymbol{\beta}(s)}{ds} = \left(\frac{d\beta_1(s)}{ds}, \frac{d\beta_2(s)}{ds} \right) = (\cos\rho(s), \sin\rho(s)) \,. \tag{5.12}$$

Based on (5.12), the unit normal, $\mathbf{n}(s)$, is defined as the unit length vector perpendicular to $\mathbf{t}(s)$.

Considering the actual behavior of an L-curve it is easy to see that, $\mathbf{t}(0)$ is parallel to the β_1 axis (L-curve becomes horizontal as $\lambda \to 0$) and that $\mathbf{t}(s)$ becomes parallel to the β_2 axis as $s \to \infty$ (L-curve becomes vertical as $\lambda \to \infty$). This, in turn imposes the following constraints on θ_i, $i = 0, 1, 2$:

$$\theta_1 - \theta_0 + \theta_2 = \frac{\pi}{2} \tag{5.13}$$

$$\theta_0, \theta_1, \theta_2 < \frac{\pi}{2} \tag{5.14}$$

Equation (5.13) ensures that there are exactly $\frac{\pi}{2}$ degrees between the tangent vectors $\mathbf{t}(0)$ and $\mathbf{t}(\infty)$ and (5.14) is a natural consequence of the fact that $\beta_2(s)$ is a monotone decreasing function of $\beta_1(s)$ [86].

We denote the origin chosen for the computation of $v(s)$ by $O = (a, b)$. Point a is given by $a = \beta_1(s_a)$ where $s_a < \mu_0 - \sigma_0$ and point b is given by $b = \beta_2(s_b)$ where $s_b > \mu_0 + \sigma_0$. Points on the L-curve where $s = \mu_i \pm \sigma_i,\ i = 0, 1, 2$ carry a special importance for us, namely they represent the points where $\kappa(s)$ is zero and the L-curve switches between linear and non-linear.

Before proceeding any further, we introduce the concept of *parallel transport* of a convex plane curve (i.e. $\kappa(s) \geq 0$).

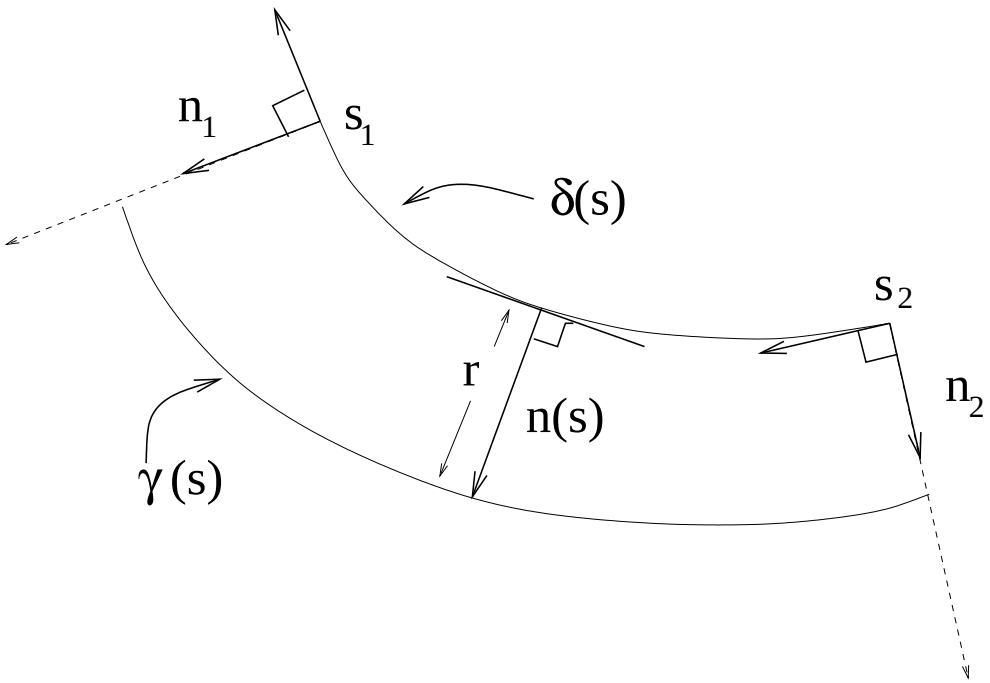


Figure 5.4: Parallel curve of a convex plane curve.

Definition 5.2.3 *Let $\delta(s)$ be a convex plane curve (a curve for which $\kappa(s) \geq 0$) positively oriented. The curve*

$$\boldsymbol{\gamma}(s) = \boldsymbol{\delta}(s) + r\mathbf{n}(s) \tag{5.15}$$

where $r > 0$ is a constant and $\mathbf{n}$ is the unit normal, is called a parallel curve to $\boldsymbol{\delta}(s)$ [32] (Fig. 5.4).

It is easy to see from the definition that the parallel curve of a plane curve is obtained by simply expanding the curve by a constant amount along the direction of the normal. The parallel transport of a convex plane curve $\boldsymbol{\delta}(s)$, denoted by $\mathcal{P}(\boldsymbol{\delta})$, is defined as the region covered by the parallel curves $\boldsymbol{\gamma}(s)$ for all $r \neq 0$ and all s such that $\boldsymbol{\gamma}(s)$ is defined. We use

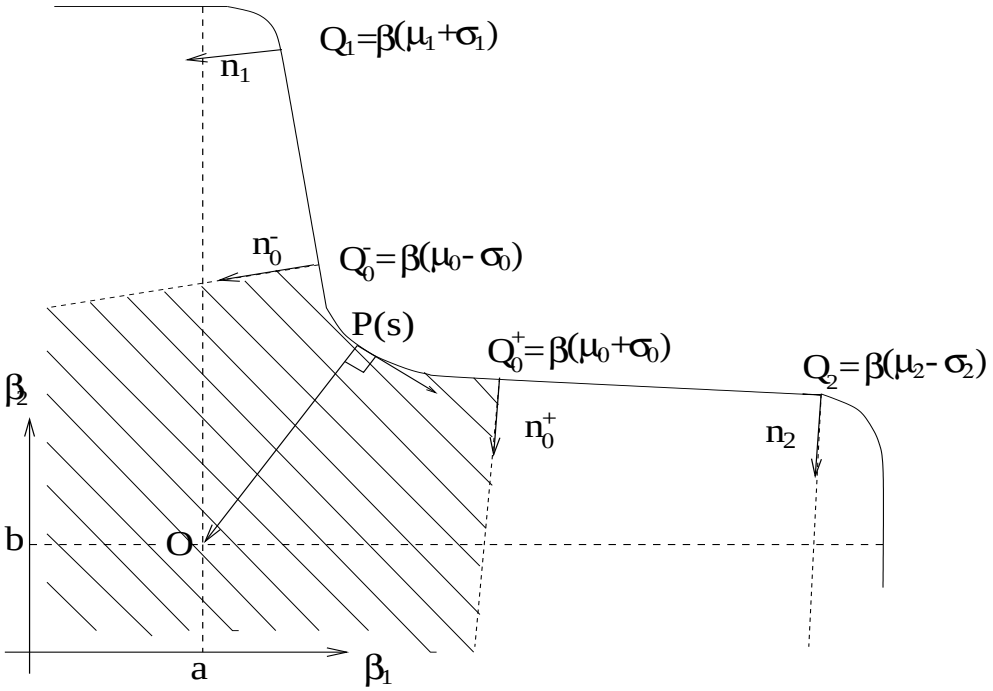


Figure 5.5: A typical L-curve.

the notation $\boldsymbol{\delta}_{[s_1,s_2]}(s)$ to denote the part of the curve $\boldsymbol{\delta}$ restricted to $s \in [s_1, s_2]$. Thus for the L-curve $\boldsymbol{\beta}$ in our example, $\mathcal{P}(\boldsymbol{\beta}_{[s_1,s_2]})$ covers all those points on the plane under the curve from which we can draw a line perpendicularly intersecting the curve $\boldsymbol{\beta}_{[s_1,s_2]}$. This property of a parallel transport will play a crucial role in the proofs of Theorems 5.2.3-5.2.9.

Theorem 5.2.1 *Let Q be a point on the plane below the convex curve $\boldsymbol{\delta}(s)$, which does not lie on $\boldsymbol{\delta}(s)$. A line from Q to $\boldsymbol{\delta}(s)$ intersecting $\boldsymbol{\delta}(s)$ perpendicularly can be drawn if $Q \in \mathcal{P}(\boldsymbol{\delta})$.*

Proof 5.2.2 Follows from the definition of $\mathcal{P}(\boldsymbol{\delta})$. □

By using the concept of parallel transport, we are able to prove the following.

Theorem 5.2.3 *$v(s)$ has a unique local minimum at $s^* \in [s_0^-, s_0^+] \equiv [\mu_0 - \sigma_0, \mu_0 + \sigma_0]$ if $O = (a, b)$ is in the region bounded above by the part of the L-curve lying between $Q_0^- = \boldsymbol{\beta}(\mu_0 - \sigma_0)$ and $Q_0^+ = \boldsymbol{\beta}(\mu_0 + \sigma_0)$ and the semi-infinite rays $\mathbf{n}_0^-, \mathbf{n}_0^+$ emanating from the points Q_0^- and Q_0^+ and perpendicular to the L-curve at the cited points (shaded region in Fig. 5.5).*

Proof 5.2.4 Given the point $O = (a, b)$, by the definition of parallel transport and Theorem 5.2.1 there exists a scalar $r^* > 0$ and $s^* \in [s_0^-, s_0^+]$ such that $O = \boldsymbol{\beta}_{[s_0^-, s_0^+]}(s^*) + r^* n(s^*)$. Define the point on $\boldsymbol{\beta}_{[s_0^-, s_0^+]}$ at s^* as $P = \boldsymbol{\beta}_{[s_0^-, s_0^+]}(s^*)$. Then we can write the vector from P to O, $\overrightarrow{PO}$, as $r^* n(s^*)$. If $t(s^*)$ denotes the tangent to $\boldsymbol{\beta}_{[s_0^-, s_0^+]}$ at P, it follows that $\overrightarrow{PO} \cdot t(s^*) = 0$. But by our earlier definitions, we have

$$t(s^*) = [\frac{d\psi[z(s^*)]}{ds}, \frac{d\psi[x(s^*)]}{ds}], \quad \overrightarrow{PO} = [(\psi[z(s^*)] - a), (\psi[x(s^*)] - b)].$$

Thus, the condition $\overrightarrow{PO} \cdot t(s^*) = 0$ implies

$$(\psi[z(s^*)] - a) \frac{d\psi[z(s^*)]}{ds} + (\psi[x(s^*)] - b) \frac{d\psi[x(s^*)]}{ds} = 0,$$

which is precisely the condition $v'(s^*) = 0$ in Definition 5.2.2. Therefore s^* is a critical point of $v(s)$ for $s \in [s_0^-, s_0^+]$.

Now suppose the s^* is not a minimum. Then there exists ϵ such that $v(s^* + \epsilon) < v(s^*)$. This would imply that $\boldsymbol{\beta}_{[s_0^-, s_0^+]}(s^* + \epsilon)$ must lie below $t(s^*)$ since it is closer to O than P is. But this would contradict the fact that $\boldsymbol{\beta}_{[s_0^-, s_0^+]}$ is convex. Therefore s^* must be a minimum of $v(s)$ in $[s_0^-, s_0^+]$. By a similar argument s^* is also seen to be unique. □

Theorem 5.2.5 *Let O be an origin satisfying the hypothesis of Theorem 5.2.3. Let s^* be the*

corresponding point minimizing $v(s)$ *in* $[s_0^-, s_0^+] \equiv [\mu_0 - \sigma_0, \mu_0 + \sigma_0]$. *Then* s^* *is the unique minimum of* $v(s)$ *for all* s *in* $(\mu_1 + \sigma_1, \mu_2 - \sigma_2)$.

Proof 5.2.6 By Theorem (5.2.3) we know that $v(s)$ has a single minimum, $v(s^*)$, at $s^* \in [\mu_0 - \sigma_0, \mu_0 + \sigma_0]$, so it suffices to prove that $v(s^*)$ is the unique minimum in $(\mu_1 + \sigma_1, \mu_0 - \sigma_0) \bigcup (\mu_0 + \sigma_0, \mu_2 - \sigma_2)$. Since the argument is the same for either subinterval, we assume without loss of generality that there is another minimum in the left subinterval, $s^{**} \in (s_0^+, s_2) \equiv (\mu_0 + \sigma_0, \mu_2 - \sigma_2)$. Let $P = \boldsymbol{\beta}(s^{**})$. By Theorem 5.2.1 applied to the curve $\boldsymbol{\beta}_{[s_0^+, s_2]}$, the only way to draw a perpendicular line to P from O is if $O \in \mathcal{P}(\boldsymbol{\beta}_{[s_0^+, s_2]})$. Since O is not in $\mathcal{P}(\boldsymbol{\beta}_{(s_0^+, s_2)})$ by our assumption, we cannot draw a perpendicular line to P. Therefore $\overrightarrow{PO} \cdot \mathbf{t}(s^{**}) = v'(s^{**}) \neq 0$, a contradiction. It follows that s^* is the unique minimum of $v(s)$ for $s \in (\mu_1 + \sigma_1, \mu_2 - \sigma_2)$. $\square$

Theorems 5.2.3 and 5.2.5 tell us that by placing the origin $O = (a, b)$ inside the region bounded by the perpendicular lines at zero curvature points $s = \mu_0 \pm \sigma_0$ on the sides and the L-curve above, we can actually create a function $v(s)$ such that the minimum of $v(s)$ is close to the corner of the L-curve (point on the L-curve for which $s = \mu_0$) and that $v(s)$ possesses a unique minimum for a wide range of s values. These are, of course, desirable properties for a surrogate function replacing the curvature since our initial goal was to create a function approximating the corner of the L-curve and having nice characteristics for the purpose of optimization. Although Theorems 5.2.3 and 5.2.5 tell us a great deal about the behavior of $v(s)$, they do not tell us how to choose an appropriate origin satisfying the condition in Theorem 5.2.3. However, as we will see in the next two theorems the choice of the origin

$O = (a, b)$ is not crucial for a well-behaved L-curve.

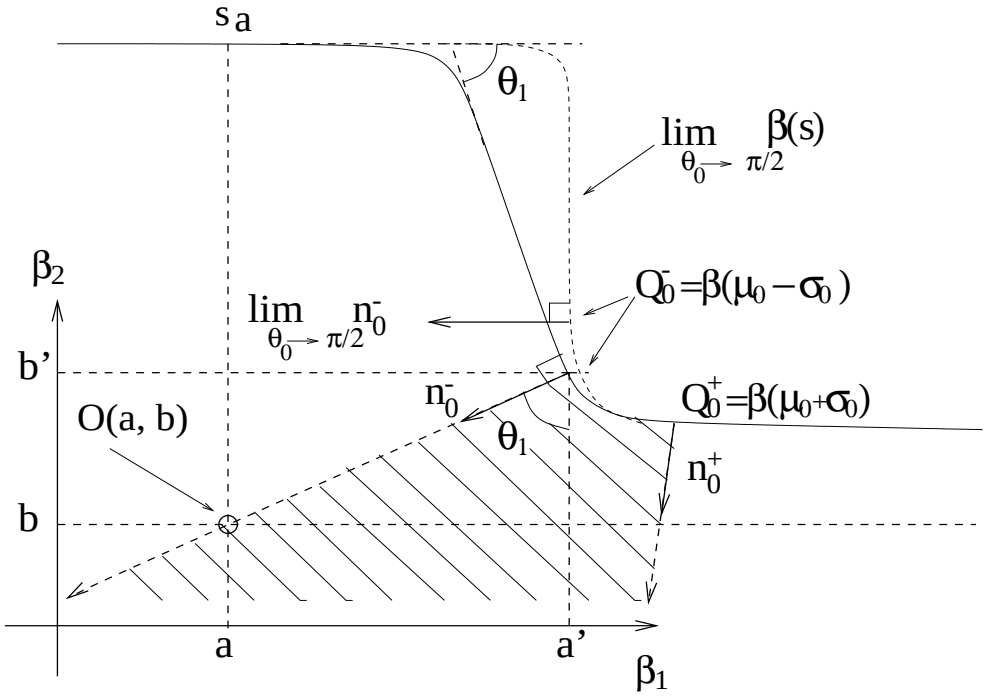


Figure 5.6: Illustration for the proof of Theorem 5.2.7. Shaded region is $\mathcal{P}(\boldsymbol{\beta}_{[\mu_0-\sigma_0,\mu_0+\sigma_0]})$.

Theorem 5.2.7 *As θ_0 approaches $\frac{\pi}{2}$, any origin $O = (a, b) = (\beta_1[s_a], \beta_2[s_b])$, such that $s_a < \mu_0$ and $s_b > \mu_0$, lies in $\mathcal{P}(\boldsymbol{\beta}_{[\mu_0-\sigma_0,\mu_0+\sigma_0]})$ for $s \in (\mu_0 - \sigma_0, \mu_0 + \sigma_0)$ and $r > 0$.*

Proof 5.2.8 $O = (a, b)$ falls to the outside of $\mathcal{P}(\boldsymbol{\beta}_{[\mu-\sigma_0,\mu_0+\sigma_0]})$ only if either O is in the region to the left of $\mathbf{n}_0^- = \mathbf{n}(\mu_0 - \sigma_0)$ or O is in the region to the right of $\mathbf{n}_0^+ = \mathbf{n}(\mu_0 + \sigma_0)$ (Fig. (5.6)). We investigate, without loss of generality, the first case.

Let us consider an origin $O(a, b) = (\beta_1[s_a], \beta_2[s_b])$ with $s_b < \mu_0$ fixed and whose β_1 coordinate a is determined by the intersection of the horizontal line $\beta_2 = b$ with $\mathbf{n}_0^-$. Since $\mathcal{P}(\boldsymbol{\beta}_{[\mu_0-\sigma_0,\mu_0+\sigma_0]})$ (shaded region in Fig. 5.6) for $r > 0$ lies below $\mathbf{n}_0^-$, $O = (a, b)$ as well as any origin whose β_1 coordinate is smaller than a results in an MDF whose minimum is outside

the region covered by $s \in (\mu_0 - \sigma_0, \mu_0 + \sigma_0)$. In other words, the intersection of the line $\beta_2 = b$ with $\mathbf{n}_0^-$ determines the boundary beyond which an origin falling outside the region $\mathcal{P}(\boldsymbol{\beta}_{[\mu_0-\sigma_0,\mu_0+\sigma_0]})$ can be found.

Since $\theta_1 - \theta_0 + \theta_2 = \frac{\pi}{2}$ and $\theta_1, \theta_2, \theta_0 < \frac{\pi}{2}$ by our assumptions, $\theta_0 \to \frac{\pi}{2}$ implies that $\theta_1, \theta_2 \to \frac{\pi}{2}$ and as $\theta_0 \to \frac{\pi}{2}$ the L-curve takes the limiting shape shown in Fig. 5.6 by the dashed line. In the limit, as $\theta_0 \to \frac{\pi}{2}$, $\mathbf{n}_0^-$ becomes horizontal and the line $\beta_2 = b$ cannot intersect $\mathbf{n}_0^-$ meaning that an $O(a, b)$ that is outside $\mathcal{P}(\boldsymbol{\beta}_{[\mu-\sigma_0,\mu_0+\sigma_0]})$ cannot be found.

Hence, as θ_0 approaches $\frac{\pi}{2}$, $\mathcal{P}(\boldsymbol{\beta}_{[\mu-\sigma_0,\mu_0+\sigma_0]})$ for $r > 0$ extends in such a way that the β_1 coordinate, a, of any origin $O(a, b)$, satisfying the conditions in Theorem 5.2.7, falls in $\mathcal{P}(\boldsymbol{\beta}_{[\mu-\sigma_0,\mu_0+\sigma_0]})$. $\square$

Now, we are ready to prove our final result.

Theorem 5.2.9 *Denote the point where $v(s)$ achieves a local minimum in $[s_0^-, s_0^+] \equiv [\mu_0 - \sigma_0, \mu_0 + \sigma_0]$ by s^*. Let $O = (a, b) \in \mathcal{P}(\boldsymbol{\beta}_{[s_0^-,s_0^+]})$ for $s \in [\mu_0 - \sigma_0, \mu_0 + \sigma_0]$ and $r > 0$. Then,* $\lim_{\sigma_0 \to 0} s^* = \mu_0$

Proof 5.2.10 By Theorem 5.2.3,

$$\mu_0 - \sigma_0 \leq s^* \leq \mu_0 + \sigma_0$$

The desired result is obtained by letting $\sigma_0 \to 0$. $\square$

By combining the results of Theorem 5.2.7 and 5.2.9 and recalling that $v(s)$ is a function of $\beta_1(s), \beta_2(s)$ we obtain the following result.

Corollary 5.2.1 *Denote the point where $v(s)$ achieves a local minimum in $s \in (\mu_0 - \sigma_0, \mu_0 + \sigma_0)$ by s^*. Let $O = (a, b)$ be such that $a = \beta_1(s_a)$ where $s_a < \mu_0$ and $b = \beta_2(s_b)$ where $s_b > \mu_0$. Then,*

$\lim_{\sigma_0 \to 0\ \theta_0 \to \frac{\pi}{2}} s^* = \mu_0$

In other words, Corollary 5.2.1 says that as $\theta_0 \to \frac{\pi}{2}$ and $\sigma_0 \to 0$ (e.g. the more the curve looks like the letter L), the point s^* for which $v(s^*)$ is a minimum coincides with the corner of the L-curve μ_0 no matter where we choose the origin (provided that a falls to the right of the corner and b falls below the corner). We use a heuristic for choosing such points in the examples: namely, we take $O = (\log z(\sigma^2_{min}), \log x(\sigma^2_{max}))$, where $\sigma_{min}, \sigma_{max}$ denote the smallest and largest singular values (or approximations thereof) of $\mathbf{H}$.

5.2.5 Multidimensional Extension of MDF

Just as we have defined the MDF in the case of an L-curve, we may consider a multidimensional extension of MDF in (5.5). The MDF for multiple regularization parameters is defined as follows:

$$v(\boldsymbol{\lambda}) = |\psi[z(\boldsymbol{\lambda})] - a|^2 + \sum_{i=1}^{M} |\psi[x_i(\boldsymbol{\lambda})] - b_i|^2 \tag{5.16}$$

where $O = (a, b_1, \ldots, b_M)$ denotes the coordinates of our origin. Extension of the analysis of the MDF in a multidimensional setting is quite complicated and intended for future work: rather, we give an intuitive explanation of why MDF is expected to work for multiple parameters.

First of all, examining Fig. (4.4) (a) reveals that the L-hypersurface is convex in the

vicinity of the maximum Gaussian curvature point, $\boldsymbol{\lambda}^*$ (i.e. $\kappa(\boldsymbol{\lambda}) > 0$). Therefore, the L-hypersurface has a bowl shaped appearance around $\boldsymbol{\lambda}^*$ and any point on the L-hypersurface lies above the tangent plane at $\boldsymbol{\lambda}^*$. Hence, the unit normal to the L-hypersurface at $\boldsymbol{\lambda}^*$, $\mathbf{N}(\boldsymbol{\lambda}^*)$, defines a line whose points, when used as an origin for the computation of $v(\boldsymbol{\lambda})$, yields an MDF which has a local minimum at $\boldsymbol{\lambda}^*$. Thus, if we choose our origin O in the close vicinity of the line defined by $\mathbf{N}(\boldsymbol{\lambda}^*)$, the minimum of $v(\boldsymbol{\lambda})$ hits a close point to the generalized corner of the L-hypersurface. One heuristic for choosing such a point when $M = 2$ (see Example 3) is to take $O = (\log z(\sigma^2_{min}, \sigma^2_{min}), \log x_1(\sigma^2_{max}, 0), \log x_2(0, \sigma^2_{max}, 0))$, and analogously for $M > 2$.

5.3 An Iterative Algorithm for Approximating $\boldsymbol{\lambda}^*$

Generally, we may use any appropriate optimization technique for finding the $\boldsymbol{\lambda}$ value which minimizes $v(\boldsymbol{\lambda})$. However, many optimization algorithms require higher order partial derivatives of $z(\boldsymbol{\lambda})$ and $x_i(\boldsymbol{\lambda})$ with respect to $\lambda_i,\ i = 1, \ldots, m$. It is shown in Appendix A that each of these partials can be computed from $\frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i}$. $\frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i}$, in turn, is obtained by solving a linear system whose size is the same as that of the original problem. Clearly, the computational effort associated with computing the required partials can be prohibitively large if the size of the problem is big as is the case for any IRR problem. However, using elementary properties of the MDF we can easily derive a fixed point algorithm for $\boldsymbol{\lambda}^*$. Differentiating (5.16) with

respect to λ_j, and equating the result to zero we obtain the following equation:

$$\sum_{i=1}^{M} \left(\psi[x_i] - b_i\right) \psi'[x_i] \frac{\partial x_i}{\partial \lambda_j} + \left(\psi[z] - a\right) \psi'[z] \frac{\partial z}{\partial \lambda_j} = 0 \tag{5.17}$$

Using (4.8, 4.9, 4.10), it is easy to show the following:

$$\frac{\partial z}{\partial \lambda_j} = 2(\mathbf{H}\mathbf{f}^*(\boldsymbol{\lambda}) - \mathbf{g})^T \mathbf{H} \frac{\partial}{\partial \lambda_j}\mathbf{f} \tag{5.18}$$

$$\frac{\partial x_i}{\partial \lambda_j} = \mathbf{f}^*(\boldsymbol{\lambda})^T \mathbf{R}_i^T \ \mathrm{diag}_{\ k=1,\ldots,m} \left[\frac{\phi_i'([\mathbf{R}_i\mathbf{f}^*]_k)}{[\mathbf{R}_i\mathbf{f}^*]_k}\right] \mathbf{R}_i \frac{\partial}{\partial \lambda_j}\mathbf{f} \tag{5.19}$$

Next, we consider (5.18):

$$\begin{aligned}
\frac{\partial z}{\partial \lambda_j} &= 2(\mathbf{H}\mathbf{f}^*(\boldsymbol{\lambda}) - \mathbf{g})^T \mathbf{H} \frac{\partial}{\partial \lambda_j}\mathbf{f} \\
&= 2(\mathbf{H}\mathbf{K}_{\mathbf{f}^*}^{-1}\mathbf{H}^T\mathbf{g} - \mathbf{g})^T \mathbf{H} \frac{\partial}{\partial \lambda_j}\mathbf{f} \\
&= 2(\mathbf{H}^T\mathbf{H}\mathbf{K}_{\mathbf{f}^*}^{-1}\mathbf{H}^T\mathbf{g} - \mathbf{H}^T\mathbf{g})^T \frac{\partial}{\partial \lambda_j}\mathbf{f} \\
&= 2\mathbf{g}^T\mathbf{H}\mathbf{K}_{\mathbf{f}^*}^{-T}(\mathbf{H}^T\mathbf{H} - \mathbf{K}_{\mathbf{f}^*})^T \frac{\partial}{\partial \lambda_j}\mathbf{f} \\
&= -2\mathbf{f}^*(\boldsymbol{\lambda})^T \left(\frac{1}{2}\sum_{i=1}^{M} \lambda_i \mathbf{R}_i^T \mathrm{diag}_{k=1,\ldots,m} \left[\frac{\phi_i'([\mathbf{R}_i\mathbf{f}^*]_k)}{[\mathbf{R}_i\mathbf{f}^*]_k}\right] \mathbf{R}_i\right) \frac{\partial}{\partial \lambda_j}\mathbf{f} \qquad (5.20) \\
&= -\sum_{i=1}^{M} \lambda_i \frac{\partial x_i}{\partial \lambda_j} \qquad (5.21)
\end{aligned}$$

where the last step follows from (5.19). Substituting (5.21) into (5.17) we obtain the following equation for $j = 1, \ldots, M$:

$$\sum_{i=1}^{m} \left((\psi[x_i] - b_i)\psi'[x_i] - \lambda_i(\psi[z] - a)\psi'[z]\right) \frac{\partial x_i}{\partial \lambda_j} = 0 \tag{5.22}$$

Note that (5.22) is actually a collection of M different equations. We can arrange those M equations into a matrix-vector equation:

$$\mathbf{J}\mathbf{r} = 0 \tag{5.23}$$

where $[\mathbf{J}]_{j,i} = \frac{\partial x_i}{\partial \lambda_j}$ and

$$[\mathbf{r}]_i = (\psi[x_i] - b_i)\psi'[x_i] - \lambda_i(\psi[z] - b)\psi'[z], \quad i = 1, \ldots, M. \tag{5.24}$$

If $\mathbf{J}$ is nonsingular, (5.23) has only the trivial solution $\mathbf{r} = \mathbf{0}$. However, the non-singularity of $\mathbf{J}$ follows from our assumption that the surface is regular (observe that $\mathbf{J}$ is one of the minors of $d\mathcal{S}$ in 5.21 obtained by taking the first M rows and M columns of $d\mathcal{S}$. Hence $\mathbf{J}$ must be nonsingular.) [32].

Thus (5.23) implies $\mathbf{r} = \mathbf{0}$. Therefore, the solution of (5.22) is given by

$$\lambda_i^* = \frac{\psi'[x_i]}{\psi'[z]} \frac{\psi(x_i[\boldsymbol{\lambda}^*]) - b_i}{\psi(z[\boldsymbol{\lambda}^*]) - a}, \quad i = 1, \ldots, M \tag{5.25}$$

If $\psi(t) = \log t$, (5.25) reduces to the following

$$\lambda_i^* = \frac{z(\boldsymbol{\lambda}^*)}{x_i(\boldsymbol{\lambda}^*)} \left(\frac{\log x_i(\boldsymbol{\lambda}^*) - b_i}{\log z(\boldsymbol{\lambda}^*) - a} \right), \quad i = 1, .., m \tag{5.26}$$

Because (5.25) $x_i = x_i(\boldsymbol{\lambda}^*)$ and $z = z(\boldsymbol{\lambda}^*)$ are also functions of $\boldsymbol{\lambda}^*$, (5.26) defines $\boldsymbol{\lambda}^*$ implicitly. Based on the formula in (5.26), we propose the following iterative algorithm to approximate $\boldsymbol{\lambda}^*$ in log scale:

$$\lambda_i^{(k+1)} = \frac{z(\boldsymbol{\lambda}^{(k)})}{x_i(\boldsymbol{\lambda}^{(k)})} \left(\frac{\log[x_i(\boldsymbol{\lambda}^{(k)})] - b_i}{\log[z(\boldsymbol{\lambda}^{(k)})] - a} \right), \quad i = 1, .., m \tag{5.27}$$

where $\boldsymbol{\lambda}^{(k)}$ is the vector of regularization parameters at step k. The algorithm is started with an appropriate initial regularization parameter vector $\lambda^{(0)}$ and iterated until the relative change in the iterates is determined to be sufficiently small.

Under some assumptions, we are able to prove[1] that if M=1, the fixed point iteration converges to a minimum of v:

[1] Proof is due to Dr. Misha Kilmer

Theorem 5.3.1 *Assume $v(t)$ has only one critical point, say $t^* \geq 0$, in (λ_a, λ_b) and that v is a minimum at that critical point. Further, assume that $z(t), -x(t)$ are strictly increasing functions of t over the interval. Then if the starting guess $t^{(0)}$ satisfies $\lambda_a < t^{(0)} \leq t^*$ and*

$$\frac{z(t^{(0)})}{z(\lambda_a)} \geq 10,$$

and t^* is such that

$$\frac{x(t^*)}{x(\lambda_b)} > 10,$$

then the fixed point iteration (5.27) converges to t^*.

Proof 5.3.2 First, define the iteration function as

$$\Theta(t) \equiv \frac{z(t)}{x(t)} \left(\frac{\log[x(t)] - b}{\log[z(t)] - a} \right).$$

Then the fixed point iteration is written $t^{(k+1)} = \Theta(t^{(k)})$.

The case $t^{(0)} = t^*$ is trivial, so in the remainder, we assume $t^{(0)} < t^*$.

Let $\mathcal{I}$ denote the closed interval $[\epsilon, t^*]$ where ϵ satisfies $\frac{z(\epsilon)}{z(\lambda_a)} = 10$.

For all t in $\mathcal{I}$, our assumptions imply $v'(t) < 0$. Using (5.21) we get

$$v'(t) = 2x'(t) \left(\frac{1}{z(t)} (\log[z(t)] - a) - t \frac{1}{x(t)} (\log[x(t)] - b \right) < 0.$$

But since $x'(t) < 0$, it follows that for all t in $\mathcal{I}$,

$$\Theta(t) > t.$$

In particular, this implies $\cdots t^{(k)} > t^{(k-1)} > \cdots > t^{(0)}$ when $t^{(0)} \in \mathcal{I}$.

Now it is straightforward to show that the derivative of the iteration function is given as

$$\Theta'(t) = \zeta(t)\frac{z'(t)}{x}\left(1 - \frac{1}{\log[z(t)] - a}\right) + \frac{z(t)(-x'(t))}{x^2(t)}\left(\zeta(t) - \frac{1}{\log[z(t)] - a}\right),$$

where

$$\zeta(t) = \left(\frac{\log(x(t)) - b}{\log(z(t)) - a}\right).$$

Using the fact that $\frac{x(t)}{b} \geq \frac{x(t^*)}{b} > 10$ and $\frac{z(t)}{a} \geq 10$ together with the positivity of $x, z, -x', z'$, it is easy to show that $\Theta'(t) > 0$ for t in $\mathcal{I}$.

Let $t^{(k)} < t^*$ (by definition, $t^{(k)} \in \mathcal{I}$). By the Mean Value Theorem, there exists $c \in (t^{(k)}, t^*)$ such that

$$\Theta'(c) = \frac{\Theta(t^{(k)}) - \Theta(t^*)}{t^{(k)} - t^*}.$$

Since we have assumed $t^{(k)} - t^* < 0$ and we know $\Theta'(c) > 0$, it must be the case that $t^{(k+1)} = \Theta(t^{(k)}) < \Theta(t^*)$.

Therefore, the iteration is producing an increasing sequence $\left\{t^{(k)}\right\}_{k=0}^{\infty}$ on the closed interval $\mathcal{I}$ and the sequence is bounded above by t^*. Thus, $t^{(k)} \to t^*$ if $t_0 \in \mathcal{I}$. $\square$

As a consequence, we know that if we pick a starting point for the single dimensional fixed point iteration which satisfies the hypotheses, the iterates will all be positive. For the multidimensional fixed point algorithm, it is difficult to determine if and when the algorithm is guaranteed to converge. We therefore leave this study of convergence for the multidimensional case for future research, but note that in practice (see Example 3), this has not been a difficulty for judicious choice of origin.

5.4 Numerical Examples

In this section, we verify the statements made concerning the behavior of the MDF and demonstrate the effectiveness of the iterative algorithm derived in 5.3 for both one and multidimensional parameter selection problems.

5.4.1 Example 1

We generated a test problem of the form $\mathbf{Hf} = \bar{\mathbf{g}}$ by using the function `shaw(100)` in Hansen's Regularization Toolbox [57] in MATLAB. We modified the exact right hand side $\bar{\mathbf{g}}$ by adding normally distributed noise, $\mathbf{n}$, scaled so that $\frac{\text{variance}(\mathbf{n})}{\text{variance}(\bar{\mathbf{g}})} = 10^{-5}$. We employed Tikhonov's regularization with the identity to estimate the original solution $\mathbf{f}$. The L-curve for this problem was then computed by sampling λ in 500 logarithmically equi-spaced points between 10^{-37} and 10^{2}. The resulting L-curve is displayed in Fig. 5.7 (a). We chose three different origins and computed corresponding $v(\lambda)$ functions. Each one of the three origins chosen were indicated by the symbols `'o'`, `'+'` and `'x'` in Fig. 5.7 (a). Origin `'o'` is especially important since it is the one we advocated using. Its coordinates were calculated as follows: First we estimated the smallest and largest singular values of the matrix $\mathbf{H}$ which were found to be $\sigma_{min} \approx 10^{-18}$ and $\sigma_{max} \approx 10$. It is known that [56], for $\lambda < \sigma_{min}^2$ the L-curve becomes almost a horizontal line and for $\lambda > \sigma_{max}^2$ the L-curve becomes almost vertical. The exact expression for the coordinates of the origin is $O = (a, b) = (\log z(\sigma_{min}^2), \log x(\sigma_{max}^2))$. In Fig. 5.7 (b) we display $v(\lambda)$ functions for each of the three origins selected. The minimum of $v(\lambda)$ for each case is marked with the appropriate symbol from Fig. 5.7 (a). Also shown

in Fig. 5.7 (c) is the plot of $\log(1 + |\kappa(\lambda)|)$ and the location of the minimum of the $v(\lambda)$ functions for each case. The region in between the dash-dotted lines in Fig. 5.7 (b)-(c) represents the part of the L-curve for which $\kappa(\lambda) \geq 0$. It is nicely seen from Fig. 5.7 (c) that for all three cases the minimum of $v(\lambda)$ is inside the cross-over region of the L-curve and that the minimum of $v(\lambda)$ for origin 'o' comes very close to the maximum curvature point. We also observe from Fig. 5.7 (b) that all three $v(\lambda)$ posses a single minimum in the region where $\kappa(\lambda) \geq 0$ as predicted by Theorem 5.2.5.

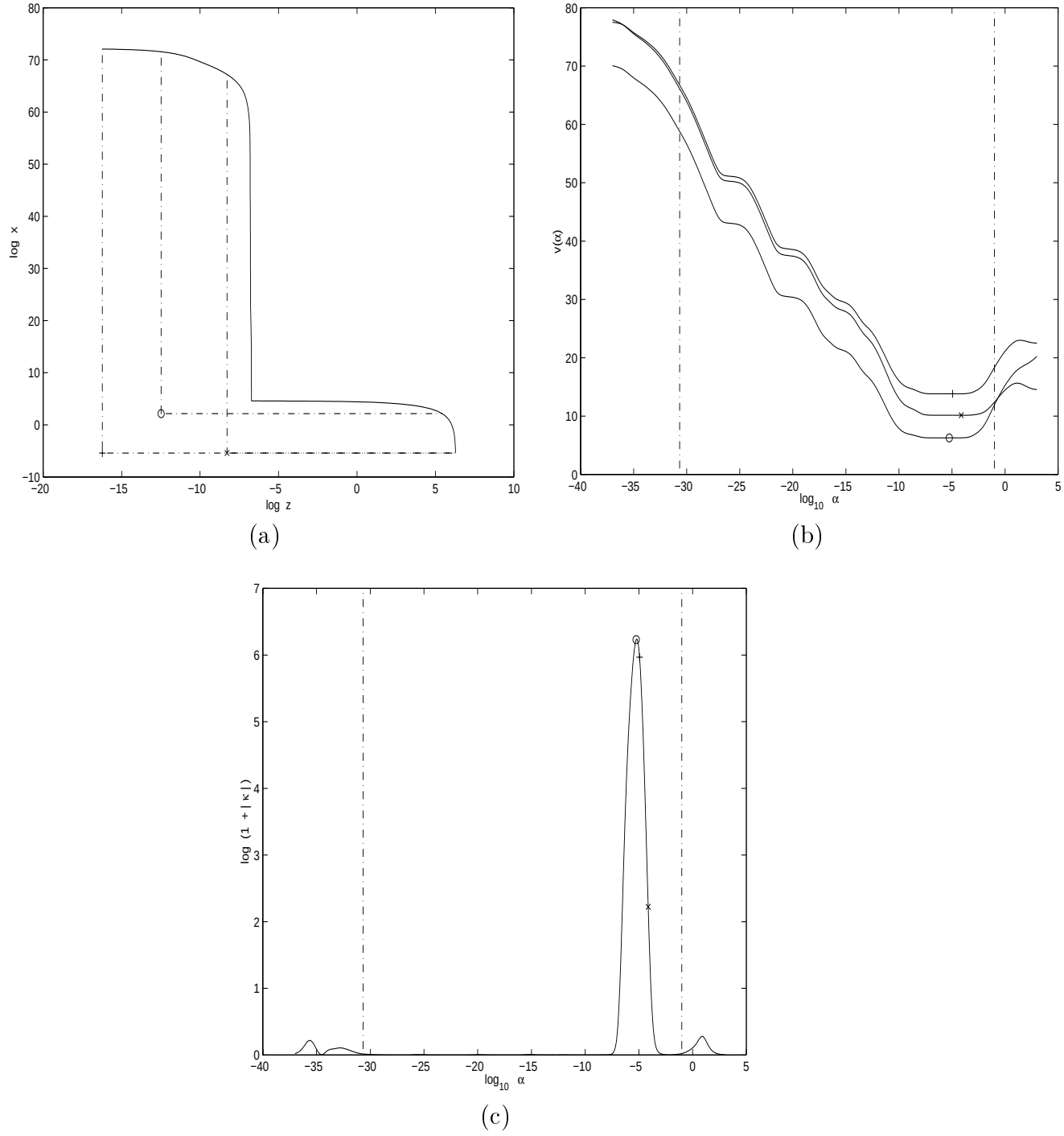


Figure 5.7: (a) L-curve for problem shaw (b) three different MDF for different choices of the origin (c) curvature of the L-curve in (a).

5.4.2 Example 2

For our second experiment we generated a 100×100 system $\mathbf{Hf} = \bar{\mathbf{g}}$ by using the `baart(100)` command in Hansen's' Regularization Toolbox in MATLAB. The exact right hand side $\bar{\mathbf{g}}$ is modified by adding normally distributed noise scaled so that $\frac{\text{variance}(\mathbf{n})}{\text{variance}(\bar{\mathbf{g}})} = 10^{-10}$. We sampled the L-curve for this problem by using 500 logarithmically equi-spaced point in the interval $(10^{-37}, 10^{2})$ as seen in Fig. 5.8 (a) . The largest and smallest singular values of $\mathbf{H}$ were estimated to be $\sigma_{max} \approx 3.2$, $\sigma_{min} \approx 2.5 \times 10^{-18}$ and the origin associated with these points, $(\log z(\sigma_{min}^2), \log x(\sigma_{max}^2))$, is marked with a `'o'` in Fig. 5.8 (a) .

To verify the conclusion reached in Theorem 5.2.7 and to demonstrate the convergence behavior of the iterative algorithm in (5.27), we first found the regularization parameter λ_{corner} maximizing the curvature of the L-curve and then used a number of different origins, $O(a, b) = (\log z(\lambda_a), \log x(\lambda_b))$, such that $\lambda_a < \lambda_{corner}$ and $\lambda_b > \lambda_{corner}$, to compute $v(\lambda)$. The region covered by the origins selected this way is indicated with the dash-dotted rectangle in Fig. 5.8 (a) . We ran our iterative algorithm to find the λ^* minimizing $v(\lambda)$ for each case. We employed the Δ^2-method of Aitken [1] to accelerate the convergence of our algorithm. In the Δ^2-method of Aitken, $\lambda^{(k)}$ values obtained in the previous iterations are extrapolated to provide a new sequence which converges faster than the original sequence

$$\lambda^{(k+1)} = \Theta(\lambda^{(k)}) - \frac{\left(\Theta(\lambda^{(k)}) - \lambda^{(k)}\right)^2}{\Theta(\lambda^{(k)}) - 2\lambda^{(k)} + \lambda^{(k-1)}} \, . \tag{5.28}$$

Note that Δ^2-method of Aitken has the same computational complexity as the original fixed point iteration.

To provide a comparison we also minimized the MDF function for each origin by using

a quasi-Newton method called BFGS [21] with a line search to ensure global convergence. For both our algorithm and BFGS the stopping condition was $\frac{|\log_{10}\lambda^{(k+1)}-\log_{10}\lambda^{(k)}|}{|\log_{10}\lambda^{(k)}|} < 10^{-3}$ and the starting value was $\lambda^{(0)} = 10^{\frac{1}{2}(\log_{10}\lambda_a+\log_{10}\lambda_b)}$. The results of this experiment were illustrated in Fig. 5.8 (a)-(f). In Fig. 5.8 (b), we plotted the curvature and indicated each point computed by our iterative scheme by placing a dot at the corresponding position on the curvature plot. Thus, the part of the curvature plot which appears bold signifies those points computed by our algorithm. It is seen from Fig. 5.8 (b), regularization parameters minimizing $v(\lambda)$ falls into the cross-over region of the L-curve independent of the chosen origin. This verifies our assertion in Theorem 5.2.7.

A sample run of the iterative algorithm, for the origin we proposed, is demonstrated in Fig. 5.8 (c), by indicating each point computed by the iterative algorithm with a '+' on the MDF. Circle, in this figure indicates the final point converged. Figure 5.8 (d) shows a normalized histogram of the number of iterations needed for each run of our algorithm (the algorithm was run for 56,244 different origins within the box and it always converged to the minimum of v). As illustrated in the figure, for most of the origins chosen the fixed point algorithm converges in fewer than 10 iterations. Now BFGS can require multiple function and gradient evaluations at each iteration because of the line search. But evaluation of either the function or the gradient requires the solution of a linear system of the size of the original problem, making each iteration of BFGS a minimum of 2 times more expensive than an iteration of our fixed point algorithm. Therefore, while the number of function evaluations required by our algorithm is the same as the number of iterations, the number of

function and gradient evaluations required by BFGS with line search is typically more than twice the number of iterations (Figure 5.8 (f) depicts the number of function and derivative evaluations for BFGS with line search) thereby making our algorithm more efficient than BFGS.

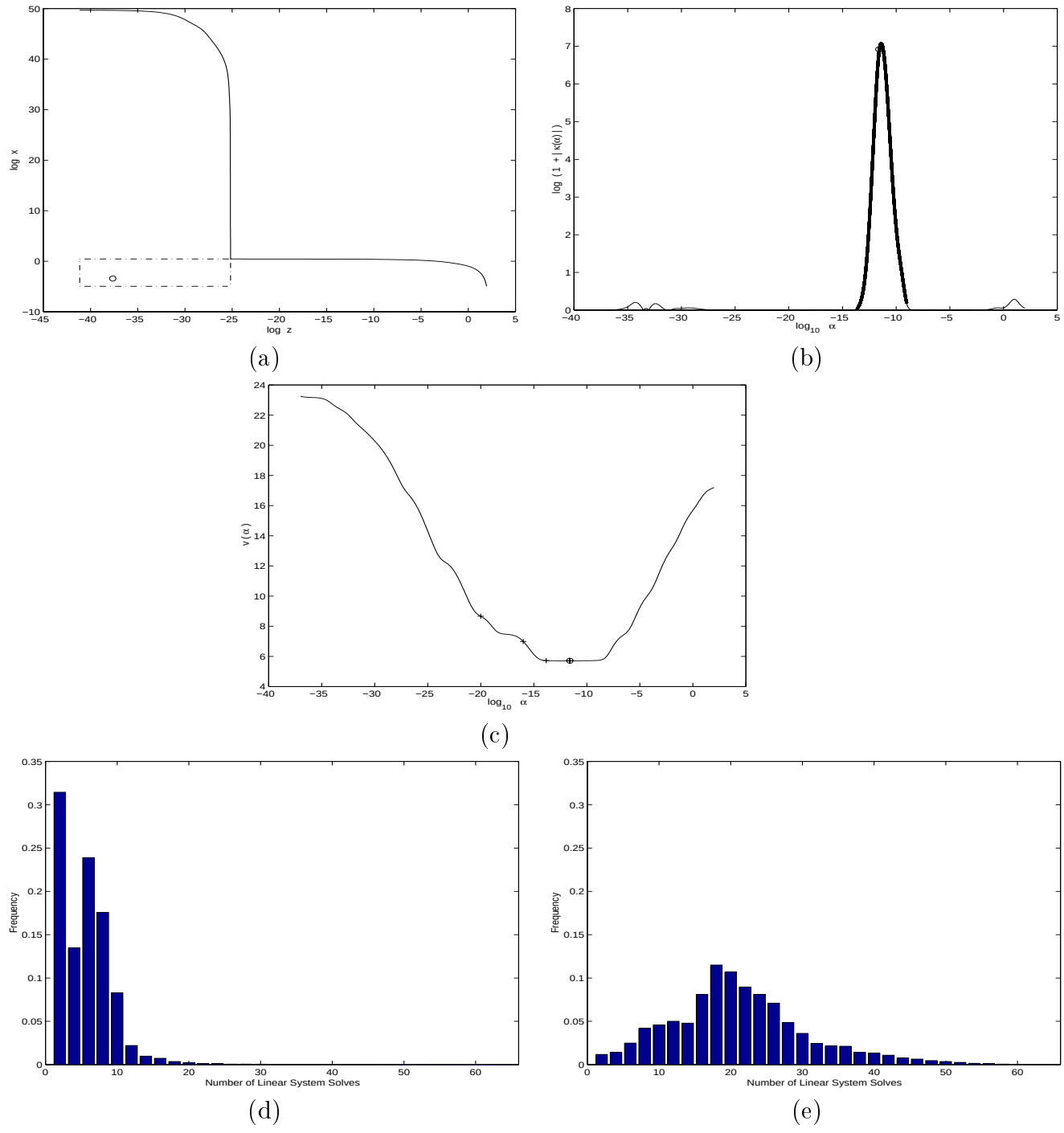


Figure 5.8: (a) L-curve for baart problem. The rectangle covers the region spanned by all possible choices of origin. 'o' is the origin we advocated. (b) Curvature plot. (c) A typical run of our iterative algorithm. (d) Histogram of number of linear systems solved (equal to the number of iterations needed) needed for each run of our algorithm. (e) Histogram of number of linear system solved (greater than or equal to the twice the number of iterations) for each run of BFGS with line search.

5.4.3 Example 3

In our final example, we try to demonstrate the utility of the MDF in a multiple regularization parameter setting. The test problem of interest was generated by using `phillips(100)` function. The exact right hand side is again modified by adding normally distributed random noise scaled so that $\frac{\text{variance}(\mathbf{n})}{\text{variance}(\bar{\mathbf{g}})} = 1.7 \times 10^{-3}$. We obtained the regularized solution by using Tikhonov's regularization in the following way

$$\mathbf{f}^*(\lambda_1, \lambda_2) = \left(\mathbf{H}^T\mathbf{H} + \lambda_1\mathbf{I} + \lambda_2\mathbf{L}^T\mathbf{L}\right)^{-1}\mathbf{H}^T\mathbf{g} \tag{5.29}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{L}$ is a discrete approximation to first order differentiation. The minimum and the maximum eigenvalues of $\mathbf{H}$ were $\sigma_{min} \approx 2.2 \times 10^{-6}$ and $\sigma_{max} \approx 5.8$. The origin chosen for the computation of the MDF was $O = (\log z[\sigma^2_{min}, \sigma^2_{min}], \log x_1[\sigma^2_{max}, 0],$ $\log x_2[0, \sigma^2_{max})]$.

We computed the MDF (v) and the mean square error (MSE), $\frac{1}{N}\|\mathbf{f} - \mathbf{f}^*(\lambda_1, \lambda_2)\|_2^2$, by sampling regularization parameters at 20 logarithmically equi-spaced points between 10^{-8} and 10^3. The resulting MDF and MSE surfaces are displayed in Fig. 5.9 (a)-(b). We compared the performance of BFGS with line search, which is guaranteed to converge to a minimum of the MDF, with our fixed point iteration. Ideally, we are only interested in $\boldsymbol{\lambda} \geq 0$. We note that neither of these methods is guaranteed to have non-negative iterates, but since the non-negativity constraint is not violated at the minimum of v and both converge to the minimum for reasonable starting points, we chose to ignore the non-negativity constraint.

We started our iterative algorithm with three different initial values of $\boldsymbol{\lambda} = [\lambda_1, \lambda_2]^T$. The stopping criteria we used for BFGS was that the norm of the gradient be less than 10^{-6}.

The stopping criteria we used for our fixed point algorithm was

$$\max_{i=1,2} |\lambda_i^k - \lambda_i^{k-1}|/|\lambda_i^{k-1}| \leq 10^{-4}.$$

In fact, 10^{-4} may be a smaller tolerance than is necessary to get good λ_1, λ_2 for a practical reconstruction; however, since BFGS tended to converge to solutions where this measure was of order of 10^{-4}, this tolerance was useful for comparison purposes.

For each run, the points computed by our algorithm at each iteration are indicated on both the MDF and MSE surfaces. In Fig. 5.9 (a)-(b), '+' indicates the trajectory of the algorithm for $\boldsymbol{\lambda}^{(0)} = [10^{-3}, 10^{-7}]$, 'o' is the trajectory of the algorithm for $\boldsymbol{\lambda}^{(0)} = [10^{-5}, 10^{-5}]$ and 'x' indicates the trajectory of the algorithm for $\boldsymbol{\lambda}^{(0)} = [10^{-7}, 10^{-3}]$. In all three cases, our fixed point algorithm converged to the same point in fewer than 9 iterations. Recall that for every iteration, one system of the form (5.29) needs to be solved.

In contrast, BFGS took 28 or 29 iterations, depending on the starting point, to converge using the stopping criteria based on the norm of the gradient. Further, it took about 49 function evaluations plus the same number of gradient evaluations to reach convergence for each of the three cases. Each function evaluation is equivalent to solving one linear system of the form (5.29). Additionally, each gradient evaluation requires solving 2 additional linear systems of the form (5.29). Thus, about 49×3 linear systems need to be solved before convergence is reached. Therefore the number of linear systems required for BFGS with line search to reach convergence was roughly 15 times more than for our fixed point algorithm.

Fig. 5.9 (b) shows that the λ_1, λ_2 values found by our fixed point iterative algorithm indeed result in a close-to-optimal MSE. Fig. 5.10 shows the MDF with the trajectory of

BFGS for $\boldsymbol{\lambda}^{(0)} = [10^{-5}, 10^{-5}]$.

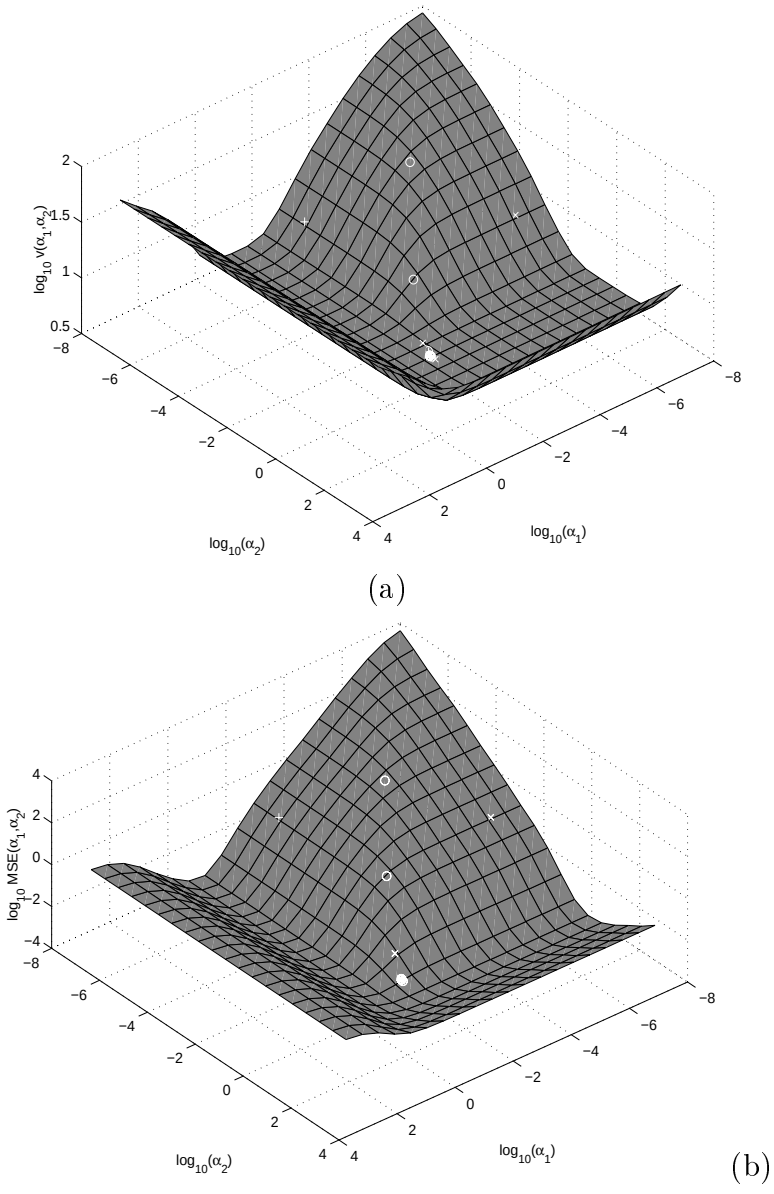


Figure 5.9: (a) Plot of v function. (b) Corresponding MSE surface. '+', 'o' and 'x' indicate the trajectory of our iterative algorithm for different starting points.

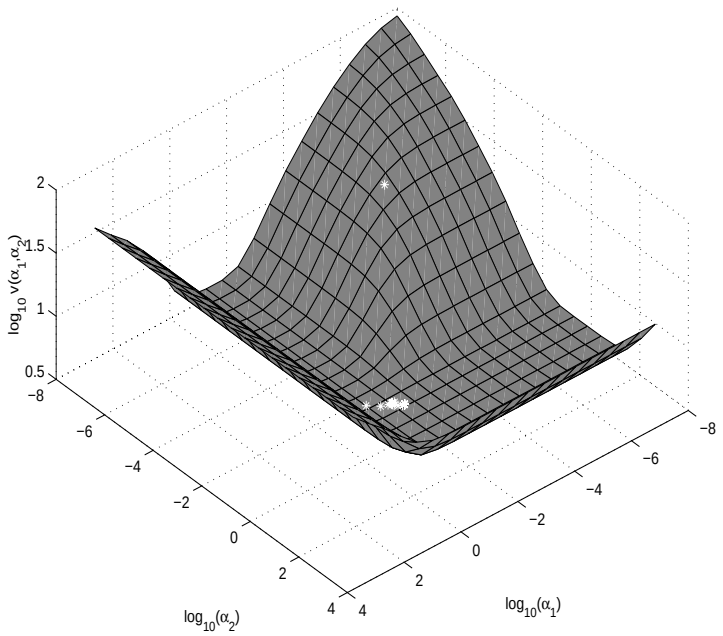


Figure 5.10: Plot of v function with trajectory of BFGS for origin $[\lambda_1, \lambda_2] = [10^{-5}, 10^{-5}]$.

Chapter 6

Conclusions and Future Work

In this thesis, we have explored the two basic problems related to the regularized solution of discrete linear inverse problems: specifying an appropriate prior that captures important features of the object we desire to recover, and the selection of unknown parameters governing the behavior of this prior model. Our approach to the problem combined the essential elements of both the statistical and the functional theoretic interpretations of the regularization method. Specifically, we interpreted the theory in terms of Bayesian estimation in developing our prior models and made use of the techniques developed in the context of functional analytic interpretation of the regularization method to deal with estimating multiple parameters required by our inversion algorithms.

In particular, in the first part of this thesis, we introduced a wavelet domain multiscale image restoration algorithm for use in linear image restoration problems. Following the recent results in the area of image denoising and coding, we developed a statistical prior

model for modeling the wavelet coefficients of images. Our priors are able to capture spatial, scale and orientational characteristics of images accurately. We developed a half-quadratic algorithm to solve the nonlinear optimization problem resulting from using such priors.

In the second part of this thesis, we made use of a multi-variate generalization of the conventional L-curve method, the L-hypersurface, for choosing multiple regularization parameters. To resolve the difficulties associated with the visualization and the interpretation of the complicated behavior of the L-hypersurface, we extended the notion of the curvature for plane curves to the notion of Gaussian curvature for hypersurfaces. Gaussian curvature can be viewed as a geometrically meaningful transformation of the L-hypersurface which emphasizes important points on the hypersurface. From numerical examples, it was seen that the points where Gaussian curvature reaches a local maxima appear to be closely tied to the local minima of the mean square error surface. We chose the regularization parameters as those maximizing the Gaussian curvature of the L-hypersurface. Monte Carlo simulations showed that this selection method is more robust than GCV and is capable of producing results comparable to the optimal method. We integrated the L-hypersurface method with our multiscale IRR algorithm to develop a highly flexible framework for adaptively determining the appropriate level of regularization as a function of the underlying structure in the image; in particular, scale-to-scale or orientation based features. We verified the performance of this restoration scheme on a variety of images, comparing the results both to smoothness constrained methods and the TV restorations. Experimental results showed that our algorithm can produce restorations which are visually significantly better than that

of the traditional techniques and at least comparable, if not better, than that of the the edge-preserving algorithms in practical situations.

In the last part of the thesis, we dealt with reducing the computational load of the L-hypersurface method. Based on simple geometrical ideas, we defined a surrogate function, called the minimum distance function (MDF), to replace the curvature function. The analysis we carried out on a proposed L-curve model indicated that, in the single parameter case, the regularization parameters minimizing the MDF approximately maximizes the curvature as the corner of the L-curve becomes sharper. This latter point was confirmed by numerical examples performed on actual L-curves. We also developed an iterative fixed point algorithm to approximate the regularization parameters minimizing the MDF. In the case of a single regularization parameter, we were able to prove the fixed point converges to a minimum of the MDF under certain assumptions. It was shown through numerical experiments that the iterative algorithm quickly converges. Thus, the computational effort associated with computing approximations to the regularization parameters that correspond to the generalized corner of the L-hypersurface has been greatly reduced. The potential tradeoff is a slight degradation in the MSE of the reconstruction if the origin chosen is not optimal.

Even though the work presented in this thesis provides interesting ideas about the solution to the discrete linear inverse problems and the associated multiple regularization parameter selection problem, the issues we dealt with suggest numerous avenues for possible extensions and future work. In the area of multiscale IRR scheme described in this thesis, the following is a list of interesting future directions that require further investigation:

- Adaptive determination of the p and α parameters appearing in our multiscale algorithm.

- Determining how to structure the regularizer (i.e. which model to use) automatically.

- Analyzing the performance of the proposed multiscale MAP estimator by computing the Cramer-Rao bounds.

In the case of the L-hypersurface method as discussed in this thesis the following outlines interesting future work:

- Discrete approximations to the curvature of the L-hypersurface. That is, computing the Gaussian curvature from the knowledge of the regularized solution $\mathbf{f}^*(\boldsymbol{\lambda})$ alone.

- Analysis of the properties of multiple parameter Minimum Distance Function (MDF) and the convergence analysis of the of the fixed point iterative algorithm for the determination of multiple parameters (FIDMP) described in Chapter 5.

- Application of the same geometrical modeling framework for the derivation and convergence analysis of a fixed point iterative algorithm for the minimization of the Reginska function [86]. The Reginska function, $y(\lambda) = \|\mathbf{g} - \mathbf{H}\mathbf{f}^*(\lambda)\|\|\mathbf{L}\mathbf{f}^*(\lambda)\|$, is suggested as an alternative to the maximum curvature point for the determination of a single regularization parameter. A local minimum of the Reginska function approximates the maximum curvature point.

Appendix A

Formulas for the Implementation of the L-hypersurface

In this appendix, we provide analytical formulas for the numerical computation of the Gaussian curvature of an M-parameter L-hypersurface represented as the following:

$$\mathcal{S}(\boldsymbol{\lambda}) = (\bar{x}_1(\boldsymbol{\lambda}), \ldots, \bar{x}_M(\boldsymbol{\lambda}), \bar{z}(\boldsymbol{\lambda}))\,, \tag{A.1}$$

where $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_M]^T$, $\bar{x}_i = \log x_i, \quad i = 1, \ldots, M$ and $\bar{z} = \log z$ and x_i and z are the constraint norms and the residual norm as given in (4.9) and (4.10), respectively. The curvature $\kappa(\boldsymbol{\lambda})$ at a point $\boldsymbol{\lambda}$ on the L-hypersurface is given by

$$\kappa(\boldsymbol{\lambda}) = \frac{(-1)^M}{w^{M+1}} |\mathbf{P}| \tag{A.2}$$

where $w^2 = 1 + \sum_{i=1}^{M} (\frac{\partial \bar{z}}{\partial \bar{x}_i})^2$, $\mathbf{P}_{i,j} = \frac{\partial^2 \bar{z}}{\partial \bar{x}_i \partial \bar{x}_j}$. From this formula it is apparent that the first and second order partial derivatives of $\bar{z}$ with respect to $\bar{x}_i, \;\; i = 1, \ldots, M$ are needed. Since $z(\boldsymbol{\lambda})$

is not explicitly defined in terms of the constraint norms, we obtain the necessary derivatives by implicit differentiation. We first define the following quantities:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \bar{x}_1}{\partial \lambda_1} & \cdots & \frac{\partial \bar{x}_M}{\partial \lambda_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial \bar{x}_M}{\partial \lambda_1} & \cdots & \frac{\partial \bar{x}_1}{\partial \lambda_M} \end{bmatrix}, \tag{A.3}$$

$$\mathbf{D}_\lambda^2 = \begin{bmatrix} \frac{\partial^2}{\partial \lambda_1^2} & \cdots & \frac{\partial}{\partial \alpha_1 \partial \lambda_M} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2}{\partial \lambda_M \partial \lambda_1} & \cdots & \frac{\partial^2}{\partial^2 \lambda_M} \end{bmatrix}, \quad \mathbf{d}_\lambda = \begin{bmatrix} \frac{\partial}{\partial \lambda_1} \\ \vdots \\ \frac{\partial}{\partial \lambda_M} \end{bmatrix}. \tag{A.4}$$

Note that, $\mathbf{d}_\lambda$ and $\mathbf{D}_\lambda^2$ are operators which represent the first and second order differentiation with respect to the variables $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_M]^T$. By the definitions above, the partial differentials of $\bar{z}$ with respect to $\bar{x}_i$ are given by:

$$\frac{\partial \bar{z}}{\partial \bar{x}_i} = \mathbf{e}_i^T \mathbf{J}^{-1} \mathbf{d}_\lambda[\bar{z}] \tag{A.5}$$

$$\frac{\partial^2 \bar{z}}{\partial \bar{x}_i \partial \bar{x}_j} = \mathbf{e}_j^T \mathbf{J}^{-T} \mathbf{D}_\lambda^2[\bar{z}] \mathbf{J}^{-1} \mathbf{e}_i - \mathbf{d}_\lambda^T[\bar{z}] \mathbf{J}^{-T} \begin{bmatrix} \mathbf{e}_j^T \mathbf{J}^{-T} \mathbf{D}_\lambda^2[\bar{x}_1] \mathbf{J}^{-1} \mathbf{e}_i \\ \vdots \\ \mathbf{e}_j^T \mathbf{J}^{-T} \mathbf{D}_\lambda^2[\bar{x}_M] \mathbf{J}^{-1} \mathbf{e}_i \end{bmatrix}, \tag{A.6}$$

where $\mathbf{e}_i$ denotes the unit vector with all zero entries except for the ith one and $1 \leq i, j \leq M$. Remembering the formulas $\mathbf{P}_{i,j} = \frac{\partial^2 \bar{z}}{\partial \bar{x}_i \partial \bar{x}_j}$ and $w^2 = 1 + \sum_{i=1}^M (\frac{\partial \bar{z}}{\partial \bar{x}_i})^2$ we realize that the curvature at the point $\boldsymbol{\lambda}$ can be readily computed by just plugging in the values $\frac{\partial \bar{z}}{\partial \lambda_i}$, $\frac{\partial^2 \bar{z}}{\partial \lambda_i \partial \lambda_j}$, $\frac{\partial \bar{x}_k}{\partial \lambda_i}$, $\frac{\partial^2 \bar{x}_k}{\partial \lambda_i \partial \lambda_j}$ into the equations in (A.6). These partial derivatives are given by:

$$\frac{\partial \bar{z}}{\partial \lambda_i} = \frac{1}{z} \frac{\partial z}{\partial \lambda_i}, \quad \frac{\partial^2 \bar{z}}{\partial \lambda_i \partial \lambda_j} = \frac{1}{z} \frac{\partial^2 z}{\partial \lambda_i \partial \lambda_j} - \frac{\partial \bar{z}}{\partial \lambda_i} \frac{\partial \bar{z}}{\partial \lambda_j}; \tag{A.7}$$

$$\frac{\partial \bar{x}_k}{\partial \lambda_i} = \frac{1}{x_k} \frac{\partial \bar{x}_k}{\partial \lambda_i}, \quad \frac{\partial^2 \bar{x}_k}{\partial \lambda_i \partial \lambda_j} = \frac{1}{x_k} \frac{\partial^2 \bar{x}_m}{\partial \lambda_i \partial \lambda_j} - \frac{\partial \bar{x}_k}{\partial \lambda_i} \frac{\partial \bar{x}_k}{\partial \lambda_j}, \quad i, j, k = 1, \ldots, M \tag{A.8}$$

and

$$\frac{\partial z}{\partial \lambda_i} = 2(\mathbf{H}\mathbf{f}^*(\boldsymbol{\lambda}) - \mathbf{g})^T \mathbf{H} \frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i}, \tag{A.9}$$

$$\frac{\partial x_k}{\partial \lambda_i} = \mathbf{f}^*(\boldsymbol{\lambda})^T \mathbf{R}_k^T \text{ diag }_{l=1,\ldots,m} \left[\frac{\phi_k'([\mathbf{R}_k\mathbf{f}^*]_l)}{[\mathbf{R}_k\mathbf{f}^*]_l} \right] \mathbf{R}_k \frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i} \tag{A.10}$$

Finally, we have

$$\frac{\partial^2 z}{\partial \lambda_i \partial \lambda_j} = 2\frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i} \mathbf{H}^T\mathbf{H} \frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_j} + 2(\mathbf{H}\mathbf{f}^*(\boldsymbol{\lambda}) - \mathbf{g})^T \mathbf{H} \frac{d^2\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i d\lambda_j}, \tag{A.11}$$

$$\begin{aligned} \frac{\partial^2 x_k}{\partial \lambda_i \partial \lambda_j} &= \frac{d\mathbf{f}^*(\boldsymbol{\lambda})^T}{d\lambda_i} \mathbf{R}_k^T \text{ diag }_{l=1,\ldots,m} \left[\frac{\phi_l''([\mathbf{R}_k\mathbf{f}^*]_l)}{[\mathbf{R}_k\mathbf{f}^*]_l} \right] \mathbf{R}_k \frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_j} \\ &\quad + \mathbf{f}^*(\boldsymbol{\lambda})^T \mathbf{R}_k^T \text{ diag }_{l=1,\ldots,m} \left[\frac{\phi_l'([\mathbf{R}_k\mathbf{f}^*]_l)}{[\mathbf{R}_k\mathbf{f}^*]_l} \right] \mathbf{R}_k \frac{d^2\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i d\lambda_j} \end{aligned} \tag{A.12}$$

for $i, j, k = 1, \ldots, M$. The only remaining quantities that we need to obtain $\kappa(\boldsymbol{\lambda})$ are the vectors $\frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i}$ and $\frac{d^2\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i d\lambda_j}$ which are provided below:

$$\frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i} = -\frac{1}{2}\tilde{\mathbf{K}}_{\mathbf{f}^*}^{-1} \mathbf{R}_i^T \text{ diag }_{k=1,\ldots,m} \left[\phi_i'([\mathbf{R}_i\mathbf{f}^*]_k)\right] \mathbf{R}_i \mathbf{f}^*(\boldsymbol{\lambda}) \tag{A.13}$$

$$\begin{aligned} \frac{d^2\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i d\lambda_j} &= -\frac{1}{2}\tilde{\mathbf{K}}_{\mathbf{f}^*}^{-1} \left[\mathbf{R}_i^T \text{ diag }_{k=1,\ldots,m} \left[\phi_i''([\mathbf{R}_i\mathbf{f}^*]_k)\right] \mathbf{R}_i \frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i} \right. \\ &\quad + \mathbf{R}_j^T \text{ diag }_{k=1,\ldots,m} \left[\phi_j''([\mathbf{R}_j\mathbf{f}^*]_k)\right] \mathbf{R}_j \frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_j} \\ &\quad \left. + \sum_{l=1}^{M} \lambda_l \mathbf{R}_l^T \text{ diag }_{k=1,\ldots,m} \left[\phi_l'''([\mathbf{R}_l\mathbf{f}^*]_k)\right] \mathbf{R}_l \left(\frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i} .* \frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i} \right) \right] \end{aligned} \tag{A.14}$$

where $.*$ denotes element-wise multiplication as in MATLAB and $\tilde{\mathbf{K}}_{\mathbf{f}^*}^{-1}$ is given by

$$\tilde{\mathbf{K}}_{\mathbf{f}^*}^{-1} = \mathbf{H}^T\mathbf{H} + \frac{1}{2}\sum_{i=1}^{M} \lambda_i \mathbf{R}_i^T \text{ diag }_{k=1,\ldots,m} \left[\phi_i''([\mathbf{R}_i\mathbf{f}^*]_k)\right] \mathbf{R}_i \tag{A.15}$$

Note that each of the derivatives of the regularized solution, $\frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i}$ and $\frac{d^2\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i d\lambda_j}$, can be obtained by solving a single linear system with a positive definite system matrix $\tilde{\mathbf{K}}_{\mathbf{f}^*}$.

To summarize, the computation of the Gaussian curvature at a point $\boldsymbol{\lambda}$ starts with computing $\frac{d\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i}$ and $\frac{d^2\mathbf{f}^*(\boldsymbol{\lambda})}{d\lambda_i d\lambda_j}$. These vectors are then used to calculate $\frac{\partial z}{\partial \lambda_i}$, $\frac{\partial^2 z}{\partial \lambda_i \partial \lambda_j}$, $\frac{\partial x_k}{\partial \lambda_i}$, $\frac{\partial^2 x_k}{\partial \lambda_i \partial \lambda_j}$ through (A.9-A.9) and (A.11-A.12). Then we obtain the partials of *logged* quantities from (A.7-A.8). These partials are then used to compute $\frac{\partial \bar{z}}{\partial \bar{x}_i}$ and $\frac{\partial^2 \bar{z}}{\partial \bar{x}_i \partial \bar{x}_j}$ by (A.5-A.6) and then we finish with plugging in these values to the Gaussian curvature formula in (A.2).

Bibliography

[1] A. C. Aitken. On bernoulli's numerical solution of algebraic equations. *Proc. Roy. Soc. Edinburgh*, 46:289–305, 1926.

[2] H. C. Andrews and B. R. Hunt. *Digital Image Restoration*. Prentice Hall, Englewood Cliffs, NJ, 1977.

[3] G. Archer and D. M. Titterington. On some bayesian regularization methods for image restoration. *IEEE Trans. Image Processing*, 4(7):989–995, July 1995.

[4] M. R. Banham and A. K. Katsaggelos. Spatially adaptive wavelet-based multiscale image restoration. *IEEE Trans. Image Processing*, 5(4):619–633, April 1996.

[5] J. Baumeister. *Stable Solution of Inverse Problems*. Vieweg, Braunschweig, 1987.

[6] F. Benyah and L. S. Jennings. The l-curve in regularization of optimal control computation. *J. Austral. Math. Soc. Ser. B*, 40:138–172, 1998.

[7] J. O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, NewYork, 1985.

[8] M. Bertero. Regularization methods for linear inverse problems. In G. Talenti, editor, *Inverse Problems*, volume 1225 of *Lecture Notes in Mathematics*, pages 52–112. Springer-Verlag, 1986.

[9] M. Bertero, T. A. Poggio, and V. Torre. Ill-posed problems in early vision. *Proc. IEEE.*, 76(8):869–889, 1988.

[10] P. Blomgren and T. F. Chan. Color tv: Total variation methods for restoration of vector-valued images. *IEEE Trans. Image Processing*, 7(3):304–309, March 1998.

[11] P. Blomgren, T. F. Chan, P. Mulet, and C. K. Wong. Total variation image restoration: Numerical methods and extensions. In *IEEE International Conference on Image Processing*, volume 3, Santa Barabara, CA, October 26-29 1997.

[12] C. Bouman and K. Sauer. A generalized gaussian image model for edge-preserving map estimation. *IEEE Trans. Image Processing*, 2(7):296–310, July 1993.

[13] Y. Bressler and A. Macovski. Three-dimensional reconstruction from projections with incomplete and noisy data by object estimation. *IEEE Trans. Acoust. Speech Signal Proc.*, ASSP-35(8):1139–1151, August 1987.

[14] D. H. Brooks, G. N. Maratos, G. Ahmad, and R. S. MacLeod. Augmented inverse problem of electrocardiography: combined time and space regularization. In *Proceedings of the Annual Conference on Engineering in Medicine and Biology*, volume 15, pages 773–774, Piscataway, NJ, Oct. 28-31 1993. IEEE Engineering in Medicine and Biology Proceedings, IEEE Publications.

[15] D.H. Brooks, G.F. Ahmad, R.S. MacLeod, and G.M. Maratos. Inverse electrocardiography by simultaneous imposition of multiple constraints. *IEEE Trans. Biomed. Eng.*, September 1998.

[16] D. C. Brown and B. H. Barber. *Progress in electrical impedance tomography*, chapter 10, pages 151–164. Inverse problems in partial differential equations. SIAM, 1990.

[17] A. M. Brukstein, B. C. Levy, and T. Kailath. Differential methods in inverse scattering. *SIAM J. Appl. Math*, 45(2):312–389, April 1985.

[18] K. P. Bube and R. Burridge. The one dimensional inverse problem of reflection seismiology. *SIAM Review*, 25(4):497–555, October 1983.

[19] R. W. Buccigrossi and E. P. Simoncelli. Image compression via joint statistical characterization in the wavelet domain. Technical Report 414, GRASP Lab. University of Pennsylvania, May 1997.

[20] J. Buckheit and D. Donoho. Wavelab and reproducible research. In A. Antoniadis, editor, *Wavelets and statistics*, pages 53–71. Springer-Verlag, 1995.

[21] M. Goldfarb C. G. Broyden, R. Fletcher and Shanno. The convergence of a class of double rank minimization algorithms. *J. Inst. Math. Appl.*, 6:222–231, 1970.

[22] D. Calvetti, G. H. Golub, and L. Reichel. Estimation of the l-curve via lanczos bidiagonalization. Technical Report SCCM-97-12, Stanford University, 1997.

[23] A. Chambolle, R. E. DeVore, N. Lee, and B. J. Lucier. Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Trans. Image Processing*, 7(3):320–335, March 1998.

[24] T. F. Chan, J. Olkin, and D. . Cooley. Solving quadratically constrained least squares using black box unconstrained solvers. *BIT*, 32:481–495, 1992.

[25] P. Charbonnier, G. Aubert L. Blanc-Feraud, and M. Barlaud. Stochastic relaxation, gibbs ditribution, and the bayesian restoration of images. *IEEE Trans. Image Processing*, 6(2):298–311, February 1997.

[26] P. Craven and G. Wahba. Smoothing noisy data with spline functions. *Numer. Math.*, 48:671–698, 1986.

[27] J. J. M. Cuppen. Numerical solution of the inverse problem of electrocardiography. Ph.d. thesis, University of Amsterdam, Department of Mathematics, 1983.

[28] I. Daubechies. *Ten Lectures on Wavelets*. SIAM Press, New York, 1992.

[29] G. Demoment. Image reconstruction and restoration: Overview of common estimation structures and problems. *IEEE Trans. Acoust. Speech Signal Proc.*, 37(12):2024–2036, December 1989.

[30] L. Desbat and D. Girard. Regularization parameters: some more efficient methods and their application to deconvolution problems. *SIAM J. Sci. Comput.*, 16:1387–1403, 1995.

[31] R. A. DeVore, B. Jawerth, and B. J. Lucier. Image compression through wavelet transform coding. *IEEE Trans. Information Theory*, 38(2):719–747, March 1992.

[32] M. P. DoCarmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, New Jersey, 1976.

[33] D. Donoho. Denoising by soft-thresholding. *IEEE Trans. Inform Theory*, 41(3):613–627, 1995.

[34] D. Donoho. Nonlinear solution of linear inverse problems by wavelet-vaguelette decomposition. *Appl. Comput. Harmon. Anal.*, 2(2):101–126, 1995.

[35] D. Donoho and I. M. Johnstone. Ideal spatial adaptation by waveletshrinkage. *Biometrika*, 81(3):425–455, 1994.

[36] D. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *J. Amer. Statist. Assoc.*, 90(432):1200–1224, 1995.

[37] D. Donoho and I. M. Johnstone. Minimax estimation via wavelet shrinkage. *Annals of Statistics*, 26(3):879–921, 1998.

[38] D. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard. Wavelet shrinkage: asymptopia? *J. Roy. Statist. Soc. Ser. B*, 57(2):301–369, 1995.

[39] D. L. Donoho. Unconditional bases are optimal bases for data compression and for statistical estimation. *Appl. Comput. Harmonic Anal.*, 1(1):100–115, December 1993.

[40] R. Dufour and E. L. Miller. Statistical signal restoration with wavelet domain prior models. Submitted to Signal Processing, manuscript available at `http://claudius.cdsp.neu.edu/nuwiirl/publications.html`

[41] R. Dufour and E. L. Miller. Statistical signal restoration with wavelet domain prior models. Submitted to Signal Processing, manuscript available at `http://claudius.cdsp.neu.edu/nuwiirl/publications.html`

[42] H. W. Engl and W. Grever. Using the l-curve for determining optimal regularization parameters. *Numer. Math*, 69(1):25–31, 1994.

[43] H. W. Engl and C. W. Groetsch. *Inverse and Ill-Posed Problems.* Academic Press, London, 1987.

[44] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of inverse problems*, volume 375 of *Mathematics and its applications.* Kluwer Academic Publishers, Dordrecht (Holland), 1996.

[45] H. Flanders. *Differential Forms with Applications to the Physical Sciences.* Dover, New York, 1989.

[46] D. Geman and C. Yang. Nonlinear image recovery with half-quadratic regularization and fft's. *IEEE Trans. Img. Proc.*, 4:932–946, July 1995.

[47] H. Gferer. An *a posteriori* parameter choice for the ordinary and iterated tikhonov regularization of ill-posed problems leading to the optimal convergence rates. *Math. Comput.*, 49:507–522, 1987.

[48] G. H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21:215–223, 1979.

[49] G. H. Golub and C. E. Van Loan. *Matrix Computations, 2nd Edition.* John Hopkins University Press, Baltimore, 1989.

[50] A. Graeme and D. M. Titterington. On some some bayesian regularization methods for image restoration. *IEEE Trans. Img. Proc.*, 4(7):989–995, 1995.

[51] C. W. Groetsch. *Inverse Problems in the Mathematical Sciences.* Vieweg, Wiesbaden, 1993.

[52] P. Hall and D. M. Titterington. On some smoothing techniques used in image restoration. *J. Roy. Statist. Soc. Ser. B*, 48(3):330–343, 1986.

[53] P. Hall and D. M. Titterington. Common structure of techniques for choosing smoothing parameters in regression problems. *J. Roy. Statist. Soc. Ser. B*, 49(2):184–198, 1987.

[54] M. Hanke. Limitations of the l-curve method in ill-posed problems. *BIT*, 36(2):287–301, 1996.

[55] P. C. Hansen. The discrete picard condition for discrete ill-posed problems. *BIT*, 30:658–672, 1990.

[56] P. C. Hansen. Analysis of discrete ill-posed problems by means of the l-curve. *SIAM Review*, 34:561–580, 1992.

[57] P. C. Hansen. Regularization tools: A matlab package for analysis and solution of discrete ill-posed problems. *SIAM J. Sci. Comput.*, 14:1487–1503, 1993.

[58] P. C. Hansen. The use of l-curve in the regularization of discrete ill-posed problems. *SIAM J. Sci. Comput.*, 14:1487–1503, 1993.

[59] P. C. Hansen. *Rank-deficient and discrete ill-posed problems.* Technical University of Denmark, Lyngby, Denmark, 1996. Doctoral Dissertation.

[60] R. F. Harrington. *Field Computations by Moment Methods.* Macmillan, New York, 1968.

[61] H. M. Hudson and T. C. M. Lee. Maximum likelihood restoration and choice of smoothing parameter in deconvolution of image data subject to poisson noise. *Comput. Stat. and Data Anal.*, 26(4):393–410, February 1998.

[62] B. Iavors'kyi and T. Rafa. Selection of regularization parameters of inverse problem by computer simulation model. In *Mathematical Methods in Electromagnetic Theory*, pages 478–480, Ukraine, September 10-13 1996. IEEE.

[63] D. Isaacson and M. Cheney. *Progress in electrical impedance tomography*, chapter 9, pages 151–164. Inverse problems in partial differential equations. SIAM, 1990.

[64] A. C. Kak. Image reconstruction from projections. In M. P. Ekstrom, editor, *Digital Image Processing Techniques*, pages 111–170. Academic Press, 1984.

[65] A. K. Katsaggelos. Iterative image restoration algorithms. *Opt. Eng.*, 28(7):735–748, July 1989.

[66] D. A. Kaufmann and G. H. Kerr. Evaluation of sythetic aperture radar noise reduction techniques for the smoothing of electric speckle pattern interferometry fringes. *J. Modern Optics*, 42(9):1795–1804, September 1995.

[67] M. E. Kilmer and D. P. O'Leary. Choosing regularization parameters in iterative methods for ill-posed problems. Technical report, University of Maryland Computer Science Dept. Report CS-TR-3937, 1998. Submitted to SISC.

[68] D. T. Kuan, A. A. Sawchuk, T. C. Strand, and P. Chavel. Adaptive noise smoothing filter for images with signal-dependent noise. *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-7(2):165–167, March 1985.

[69] S. Osher L. I. Rudin and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[70] R. L. Lagendijk and J. Biemond. *Iterative Identification and Restoration of Images*. Kluwer Academic Publishers, Boston, 1991.

[71] M. R. Luettgen, W. C. Karl, and A. S. Willsky. Efficient multiscale regularization with applications to the computation of optical flow. *IEEE Trans. Img. Proc*, 3(1):41–64, 1994.

[72] M. A. Lukas. Asymptotic optimality of generalized cross-validation for choosing the regularization parameter. *Numer. Math.*, 66:41–66, 1993.

[73] C. L. Lwason and R. J. Hanson. *Solving Least Squares Problems.* Prentice-Hall, Englewood Cliffs, NJ, 1974.

[74] P. Mathieu M. Antonini, M. Barlaud and I. Daubechies. Image coding using wavelet transform. *IEEE Trans. Image Processing*, 1(2):205–220, February 1992.

[75] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(July):674–693, 1989.

[76] Y. Meyer. *Wavelets and Operators.* Cambridge Univ. Press, New York, 1992.

[77] Eric L. Miller and Alan S. Willsky. A multiscale approach to sensor fusion and the solution of linear inverse problems. *Appl. Comput. Harmonic Anal.*, 2:127–147, 1995.

[78] Eric L. Miller and Alan S. Willsky. Multiscale, statistically-based inversion scheme for the linearized inverse scattering problem. *IEEE Trans. Geosc. Remote Sens.*, 34(2):346–357, March 1996.

[79] V. A. Morozov. *Methods for Solving Incorrectly Posed Problems.* Springer-Verlag, New York, 1984.

[80] P. Moulin and J. Liu. Analysis of multiresolution image denoising schemes using generalized-gaussian and complexity priors. *IEEE Trans. Information Theory*, 38(4):719–747, April 1998.

[81] J. Oprea. *Differential Geometry and Its Applications*. Prentice Hall, Englewood Cliffs, NJ, 1977.

[82] J. Ory and R. G. Pratt. Are our parameter estimators biased? the significance of finite difference regularization operators. *Inverse Problems*, 11:397–424, 1995.

[83] T. Raus. On the discrepancy method for the solution of ill-posed problems. *Uch. Zap. Tartu Gos. Univ.*, 672:16–26, 1984.

[84] T. Raus. On the residue principle for the solution of ill-posed problems with non-selfadjoint operator. *Uch. Zap. Tartu Gos. Univ.*, 715:12–20, 1984.

[85] S. J. Reeves and R. Mersereau. Optimal estimation of the regularization parameter and stabilizing functional for regularized image restoration. *Optical Engineering*, 29(5):446–454, May 1990.

[86] T. Reginska. A regularization parameter in discrete ill-posed problems. *SIAM Journal on Sci. Comput.*, 17(3), 1996.

[87] E. P. Simoncelli and E. Adelson. Noise removal via bayesian wavelet coring. In *Proceedings of the 1996 IEEE Int. Conf. on Image Proc.*, volume 1, pages 379–382, Lausanne, Switzerland, September 1996.

[88] E. P. Simoncelli and E. H. Adelson. Noise removal via bayesian wavelet coring. In *ICIP'96*, volume 3, pages 185–188, 1996.

[89] July 1998.

[90] A. M. Thompson, J. C. Brown, D. M. Titterington, and J. W. Kay. A study of methods of choosing the smoothing parameter in image restoration by regularization. *IEEE Trans. Pattern Analy. Mach. Intell.*, 13(4):326–339, April 1991.

[91] A. Tikhonov and V. Arsenin. *Solution of Ill-Posed Problems*. Wiley, New York, 1977.

[92] K. Timmermann and R. D. Nowak. Multiscale modeling and estimation of poisson processes with application to photon-limted imaging. *IEEE Trans. Inf. Theory*, 45(3):846–862, April 1999.

[93] D. M. Titterington. Common structure of smoothing techniques in statistiscs. *Internat. Statist. Rev.*, 53(2):141–170, 1985.

[94] D. M. Titterington. General structure of regularization procedures in image reconstruction. *Atronom. and Astrophys.*, 144(2):381–387, 1985.

[95] D. M. Titterington. Choosing the regularization parameter in image restoration. In *Spatial Statistics and Imaging*, volume 20 of *IMS Lecture Notes Monogr. Ser*, pages 392–402. Institute of Mathematical Statistics, Hayward, CA, 1991.

[96] D. T. Vuza U. Amato. Besov regularization, thresholding and wavelets for smoothing data. *Numer. Funct. Anal. and Opt.*, 18(5&6):461–493, 1997.

[97] C. R. Vogel. Non-convergence of the l-curve regularization parameter selection method. *Inverse Problems*, 12(4):535–547, 1996.

[98] C. R. Vogel and M. E. Oman. Fast, robust total variation-based reconstruction of noisy, blurred images. *IEEE Trans. Image Processing*, 7(7):813–824, July 1998.

[99] G. Wahba. Practical approximate solutions to linear operator equations when the data are noisy. *SIAM J. Numeri. Anal.*, 14:651–667, 1977.

[100] G. Wang, J. Zhang, and G. W. Pan. Solution of inverse problems in image processing by wavelet expansions. *IEEE Trans. Image Processing*, 4(5):579–593, 1995.

[101] M. E. Zervakis and T. M. Kwon. Generalized study of the weighted least-squares measure for the selection of the regularization parameter in inverse problems. In *IEEE International Symposium on Circuits and Systems, Chicago, IL*, volume 1, pages 415–418, May 3-6 1993.

[102] M. E. Zervakis, T. M. Kwon, and J-S. Yang. Multiresolution image restoration in the wavelet domain. *IEEE Trans. Circuits and Systems-II: Analog and Digit. Sig. Proc.*, 42(9):578–591, September 1995.