

Regularization of linear inverse problems

Copyright: This work is licensed under the Creative Commons Attribution 3.0 Licence, see <https://creativecommons.org/licenses/by/3.0/>

ISBN: 978-94-6423-257-8

Cover: The cover shows a Ricker wavelet. The image at the bottom is generated using the poststack seismic modeling operator from the Pylops toolbox, and the signal in the middle is a vertical section of the image. The formula may be used to reconstruct the title from the image at the bottom.

Print: Proefschriftmaken | www.proefschriftmaken.nl.

Regularization of linear inverse problems

Regularizatie van lineaire inverse problemen

(met een samenvatting in het Nederlands)

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Universiteit Utrecht
op gezag van de
rector magnificus, prof.dr. H.R.B.M. Kummeling,
ingevolge het besluit van het college voor promoties
in het openbaar te verdedigen op

woensdag 2 juni 2021 des middags te 2.15 uur.

door

Nicolaas Antonius Luiken

geboren op 17 oktober 1991
te Hengelo

Promotor: Prof. dr. Rob Bisseling

Co-promotor: Dr. Tristan van Leeuwen
Dr. Eric Verschuur

This work is sponsored by the Delphi consortium

Aan mijn ouders

Contents

	Page
1 Introduction	1
1.1 Imaging and inverse problems	2
1.2 Challenges in inverse problems	10
1.3 Outline and contributions	11
2 Preliminaries	13
2.1 The pseudoinverse	14
2.2 Tikhonov regularization	14
2.3 Non-smooth regularization	21
3 Comparing RSVD and Krylov methods for linear inverse problems	27
3.1 Introduction	29
3.2 Theory	31
3.3 Algorithms	40
3.4 Numerical experiments	44
3.5 Conclusion	51
3.6 Appendix	52
4 Block-Krylov methods for Multi-Dimensional Deconvolution	53
4.1 Introduction	54
4.2 Block Krylov methods	54
4.3 Analysis via SVD	56
4.4 Multi-Dimensional Deconvolution	57
4.5 Discussion	61
5 Seismic wavefield redatuming with regularized Multi-Dimensional Deconvolution	63
5.1 Introduction	65

5.2 Source redatuming	68
5.3 Constrained least squares	73
5.4 Numerical experiments	77
5.5 Conclusion, discussion and outlook	86

6 Relaxed regularization for linear inverse problems 89

6.1 Introduction	90
6.2 Analysis of SR3	95
6.3 Approximating the value function	102
6.4 Implementation	105
6.5 Numerical experiments	107
6.6 Conclusion and outlook	112

7 Conclusion and outlook 115

7.1 Conclusion	116
7.2 Discussion and outlook	117

Backmatter

Summary	129
----------------	------------

Samenvatting	131
---------------------	------------

Dankwoord	133
------------------	------------

Curriculum Vitae	135
-------------------------	------------



Introduction



1.1 Imaging and inverse problems

Imaging is the art of making hidden objects visible. Over the last century, technological innovation has made it possible to visualize the inside of the human body and the structure of the earth. One can think of ultrasound, Magnetic Resonance Imaging (MRI), Computerized Tomography (CT), Positron Emission Tomography (PET), but also maps of the layers of the subsurface via seismic imaging. Of course, these images are not images like the ones a camera makes, but rather, these are images in the sense that they reflect properties of a material, which can be visualized. The common denominator among these imaging methods is that the object of interest is not accessible, and that it is hidden beneath some other structure that can or should not be destructed. Indeed, one can imagine that it is undesirable to take a 3-month old baby out of the womb to see what it looks like, or to cut open a limb to check for a broken bone. The question then is, how an image is obtained. This is done by indirect measurements generated by some source. For example, in MRI one measures the energy released by hydrogen protons different tissues after they have been manipulated by a strong magnetic field. In PET, a tracer is injected into the body that leads to radiation that can be measured. Another popular method is wavefield imaging, where a sound source generates waves that penetrate the object and interact with it. The waves either penetrate the object or reflect from some material, which changes its speed and amplitude. By measuring the waves coming out of the object one can determine the material properties inside the object and form an image. Another example from geosciences is gravity surveying. Here, an unknown mass distribution in the earth generates a gravity field that can be measured at the surface. The gravity measurements can then be used to derive various rock properties in the subsurface.

Determining the material properties inside the object is very hard because we can only measure the radiation or waves outside of the object. However, if we would have access to measurements that penetrate the object from all sides, the image would reflect the actual material properties quite well. Unfortunately, the acquisition geometry is generally limited by constraints. In CT for example, the X-rays can have a damaging effect on the patient [19, 14]. Similarly, PET scans use a radiative substance inside the patient's body that is potentially damaging [1]. For CT, this means that the acquisition time is limited, and that we cannot make too many measurements. In PET, the data are very noisy due to scattering effects and so called random events. In case of seismic acquisition and seismology, one only has access to data from one side, as we do not have the luxury to place sources and receivers around the entire globe. In addition to these difficulties, all data are noisy, either due to imperfect measurements or due to events in the data that cannot be explained due to imperfect modeling. To overcome acquisition and modeling limitations, we have to resort to mathematics to provide us with the tools to make the image.

Consider now the following scenario. Given, for example, the structure of a brain or a detailed map of the subsurface, how would the wave penetrating the brain or

the subsurface behave? This problem is much better understood than the problem of determining an object from the measurements of the wave, and is called the *forward problem*. The brain or the subsurface are the input for the model, or operator, which determines how the wave behaves given a certain input, which produces data. Forward problems are problems for which we can explicitly form an operator mapping input to data. For example, we can explicitly form (at least approximately) the equations based on physics that govern the behavior of the wave in a certain medium. There is a sharp distinction that we have to make at this point, between linear and non-linear problems. In linear problems, like CT, the operator depends linearly on the input, whereas in non-linear problems, like ultrasound and seismic imaging, the operator depends non-linearly on the wavespeed.

In this thesis, we are concerned with linear inverse problems, which may be written by the mathematical formula

$$Ax = b, \quad (1.1)$$

where A is the model describing the physics, x is the input, and b are the measurements. The *inverse problem* is now to determine x , given b , given by the mathematical formula

$$x = A^\dagger b, \quad (1.2)$$

where the operator A^\dagger denotes the operator that gives the input based on the measurements. Unfortunately, in inverse problems, this operator often does not exist, and if it does, its form may not be known, or it may be too expensive to compute. To guarantee a solution, three conditions have to be satisfied, called the *Hadamard conditions* [47]:

1. A solution has to exist,
2. The solution has to be unique,
3. The solution has to be stable with respect to perturbations in the data.

If all of these conditions are satisfied the problem is called *well-posed*. If one of the conditions is not satisfied, the problem is called *ill-posed*. Generally, forward problems are well-posed, whereas their associated inverse problems are ill-posed.

So if we cannot construct the operator A^\dagger , how do we obtain x given b ? This can be done by solving

$$\min_x \|Ax - b\|, \quad (1.3)$$

where $\|\cdot\|$ is a measure of distance between Ax and b . If the measurements are ideal, i.e. no noise, then this distance is, ideally, 0. However, if no solution exists, we have to be satisfied with the minimal distance between Ax and b . Problems of the form (1.3) are called *optimization problems*. These problems are solved iteratively: at every step we try to get Ax as close to b as possible.

If the problem is ill-posed, the optimization problem (1.3) may not yield the desired solution. Therefore, we have to resort to prior knowledge about the solution to steer the solution in the right direction. This yields an optimization

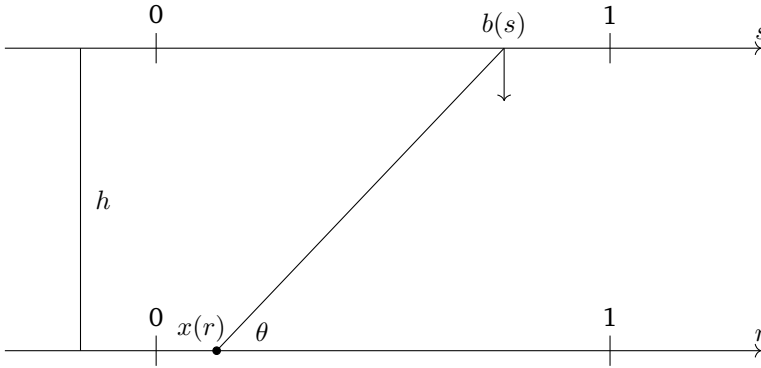


Figure 1.1: Schematic depiction of the gravity surveying problem.

problem of the form

$$\min_x \|Ax - b\| + \mathcal{R}(Lx), \quad (1.4)$$

where \mathcal{R} encodes the prior knowledge of Lx , and L is some distinct feature of the solution. The regularization $\mathcal{R}(\cdot)$ and the operator L depend on the application at hand. Below we will show some examples that reflect choices for $\mathcal{R}(\cdot)$ and L that are typical in inverse problems. For each example, the regularizer and the operator L encode specific prior information about the solution. All the examples are simplified 1D problems.

Gravity surveying

Gravity surveying is a method to derive properties of the subsurface by measuring the gravity field produced by various rocks, sedimentary and other material [30]. Figure (1.1) shows a schematic setup of this problem. Materials in the earth have a certain density and mass distribution which produce a gravity field at the surface. This gravity field is measured using an accelerometer. Generating the gravity field is the forward problem. On the other hand, deriving the mass distribution in the subsurface given the measured gravity field at the surface, is the inverse problem. Figure (1.2) shows an example of a possible mass distribution and the reconstruction using equation (1.2). The reconstruction does not resemble the true mass density at all. The reconstruction has very large amplitudes, which is due to the fact that the third Hadamard condition is violated, namely that the reconstruction is stable. Small errors in the data yield very large errors in the reconstruction. Therefore, we can add a penalty of the form $\mathcal{R}(x) = \lambda \|x\|_2^2 = \lambda \sum_{i=1}^n x_i^2$. This is a measure of the length of x . This forces the amplitude of the reconstruction to be small, and the result can be seen from figure (1.2c). Note that this type of regularization does not provide any structural information about the solution, other than that the solution should not blow up.

Traveltime tomography

Traveltime tomography is a process where the slowness, which is the inverse of velocity, is determined from the time it takes waves to travel to only a few reflectors

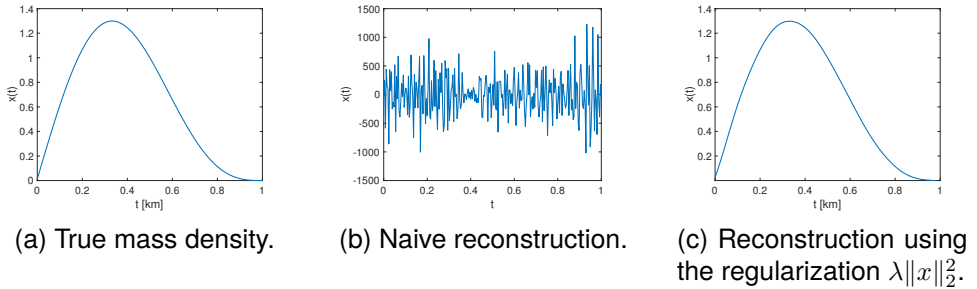


Figure 1.2: The figure on the left shows the true mass density, the figure in the middle shows the naive reconstruction and the figure on the right shows the reconstruction using the regularization $\lambda\|x\|_2^2$.

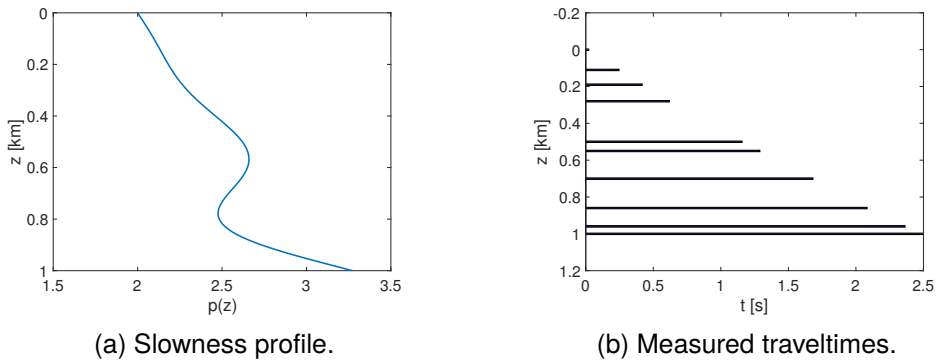


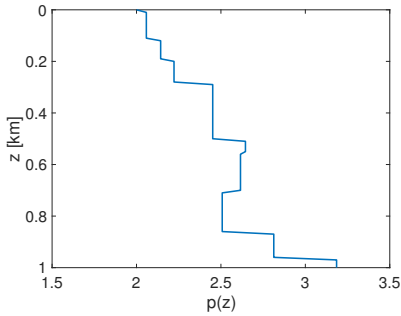
Figure 1.3

located at different known depths. We show an example of a possible slowness profile in figure (1.3a) with associated traveltimes in figure (1.3b). The naive reconstruction by using equation (1.3) yields the slowness profile shown in figure (1.4a). Although the reconstruction is far from perfect, the residual is actually 0. Due to the fact that we only have the traveltimes from a limited number of reflectors, we cannot determine the correct solution. In this case, condition 2 of the Hadamard conditions is violated, namely that the solution is not unique.

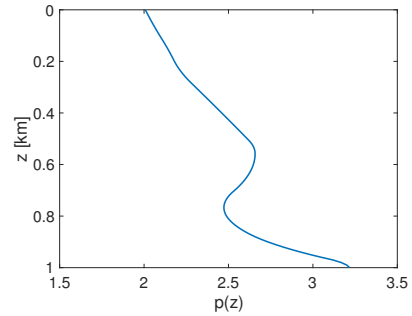
What we can observe is that the velocity profile is smooth, which is also a valid a-priori assumption. We can include this as prior information using the penalty $\mathcal{R}(Lx) = \lambda\|\nabla x\|_2^2$, i.e., we want the change in slowness as a function of depth to be small. This yields the reconstruction shown in figure (1.4b).

Spiky deconvolution

An image of the subsurface is obtained by sending in a pulse and deriving physical properties, like the acoustic impedance, from the reflected signal, the measurement. The reflectivity of the subsurface represents jumps in the velocity. We show an



(a) Naive reconstruction. The residual is zero, but too few measurements make that there exists no unique solution.



(b) Reconstruction using prior information about the smoothness of the slowness profile.

Figure 1.4

example of a subsurface model and the associated reflectivity in figure (1.5). The

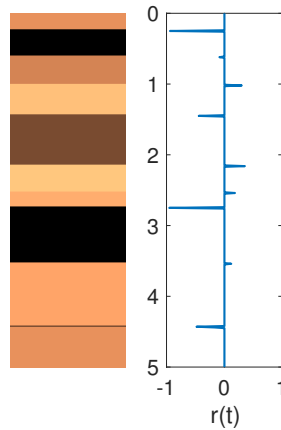


Figure 1.5: The left panel shows a schematic drawing of the subsurface, where the different colors represent different layers. The right figure shows the associated reflectivity.

reflected signal may be seen as a sum of the effects of the individual reflectors on the input signal. This can be described mathematically by *convolution*, and the process of retrieving the reflectors is called *deconvolution* [102]. Figure (1.6) shows the input signal, the reflectivity, and the measured data. For this problem the naive solution is also unstable and therefore, we may try to use the regularization $\lambda \|x\|_2^2$ from the gravity example. The reconstruction is shown in the left figure in (1.7). Note that, although the reconstruction has picked up the correct location of the spikes, the spikes are not reconstructed. In fact, they look like a dispersed copy of the input

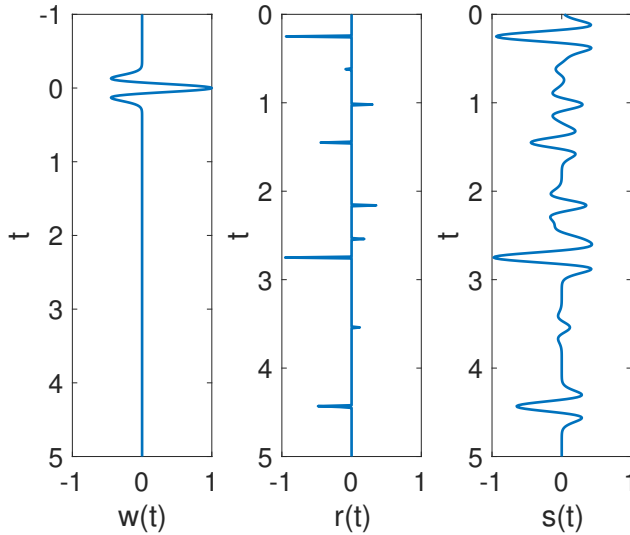


Figure 1.6: From left to right: the input signal, reflectivity and the measured data. The input signal travels downward. When it hits a reflector the signal is reflected and we see a pulse in the measured signal.

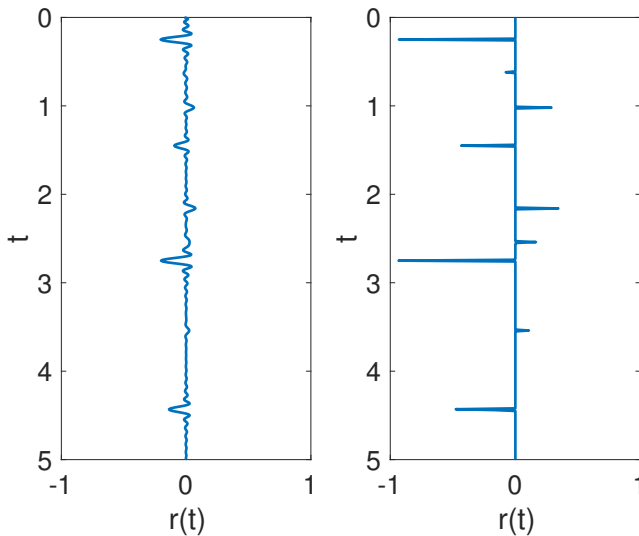


Figure 1.7: The left figure shows the reconstruction using the regularization $\lambda \|x\|_2^2$. The right figure shows the reconstruction using the regularization $\lambda \|x\|_1$.

signal. This phenomenon is due to the fact that the input signal is bandlimited and the high frequency components are missing. Another way to look at this is to say

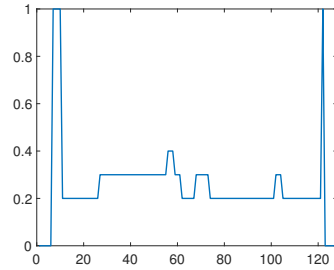
that the system A cannot distinguish spikes from smoother signals, and hence A^\dagger cannot reproduce them. In this case, our prior knowledge is that the reflectivity consists of only a few spikes. This is called a *sparse* signal, and sparsity can be enforced by the regularization $\lambda\|x\|_1 := \lambda \sum_{i=1}^n |x_i|$. Using this regularization, we get the reconstruction on the right in figure (1.7).

Brain scan

The Shepp-Logan is a famous phantom from the medical community that is a simplified version of a brain scan [105], see figure (1.8). The image can be



(a) The Shepp-Logan phantom.



(b) The vertical cross section in the middle of the phantom.

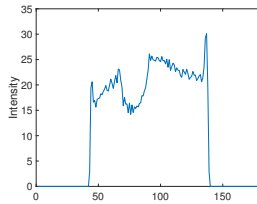
Figure 1.8

reconstructed using X-rays at various angles. Figure (1.9) shows the setup and the measured data. Figure (1.9a) shows the acquisition setup with the phantom. The X-rays are emitted on one side of the phantom. The X-rays travel through the phantom and are attenuated. Figure (1.9b) shows the measured intensity for the particular angle from figure (1.9a) and figure (1.9c) shows the intensities for all angles, called the sinogram. By measuring the difference in intensity between the X-ray before and after passing through the phantom we can derive what material is inside the phantom.

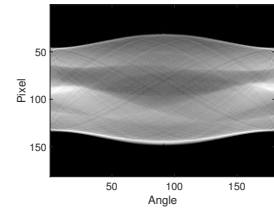
The quality of the reconstruction depends strongly on the acquisition setup. If we measure at all angles, we get a perfect reconstruction. In practice, the angles at which one can measure are generally restricted. It may be possible that one can measure along the full 180 degrees, but only at a limited subset of angles, or there may be a limited angle setup, where one can only measure from, for example, 0 to 60 degrees. We show the naive reconstruction for the full angle and limited angle setup in figures (1.10c) and (1.10a). The question arises what regularization is suited for this phantom. The answer can be found by looking at figure (1.8b). Note that the slice has a "blocky" structure, where the function is mostly constant but with a few jumps. This prior information can be encoded by using the regularization $\lambda\|\nabla x\|_1$, which means that the changes in the solution (∇x) have to be sparse. The reconstruction using this prior is shown in figure (1.10d). It is interesting to look at figure (1.10b), which is obtained by reconstructing the phantom with a limited angle setup but using prior information. Note that,



(a) The acquisition setup of the tomography problem.

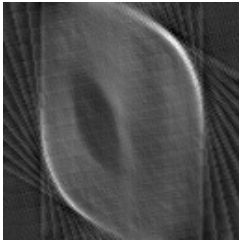


(b) The measured intensity for the setup of figure (1.9a).

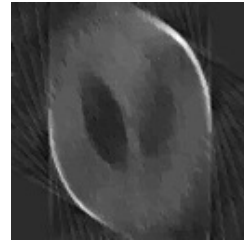


(c) The full sinogram, i.e., the measurements from all angles.

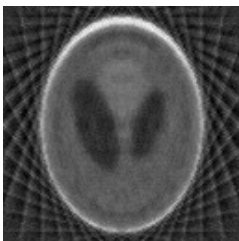
Figure 1.9



(a) Naïve reconstruction for the limited angle setup, where we measure only from 0-60 degrees.



(b) Reconstruction with prior information for the limited angle setup, where we measure only from 0-60 degrees.



(c) Naïve reconstruction for the full range setup, but with only a few angles.



(d) Reconstruction with prior information for the full range setup, but with only a few angles.

Figure 1.10

although we have good prior information, we are not able to reconstruct the

phantom, due to the limited angle setup. This shows that regularization is no magic fix: the measurements have to be sufficient as well. Unfortunately, we are not always able to obtain good measurements, and we have to be satisfied with a sub-optimal reconstruction.

1.2 Challenges in inverse problems

Until now we have not discussed how to actually solve these optimization problems. If $\mathcal{R}(Lx) = \lambda \|Lx\|_2^2$ the objective function is smooth and we can use standard optimization techniques. However, if $\mathcal{R}(Lx) = \lambda \|Lx\|_1$, the objective is not smooth and we need specialized algorithms that can deal with this type of regularizer. Furthermore, algorithms for solving regularizers where $L \neq I$ are generally different than the algorithms used for solving problems with $L = I$.

So far, all of the regularizers we have described are of the form $\lambda \|Lx\|_p^p$, but we have never specified the parameter λ and its role. This is a user-specified parameter that is notoriously difficult to estimate and at the same time heavily influences the reconstruction. It can be seen as a parameter that balances reliance on the data and reliance on the prior information. The optimal regularization parameter is defined as the parameter that minimizes the difference between the true input and the reconstruction. However, the true input is never known.

Let us revisit the example of spiky deconvolution. In figure (1.11) we show the reconstruction of the reflectivity for various values of λ . Figure (1.11a) and (1.11b) show a value that is too small, figure (1.11c) shows the optimal value and figure (1.11d) shows a value that is too large. What we see in figure (1.11) is the increasing influence of the regularizer. In figure (1.11a) we see a bad reconstruction, with waves instead of spikes. This is due to the fact that high frequency information is lost in the convolution, because the input wavelet does not contain them. As we start to increase λ , the reconstruction becomes more spiky, which can be seen in figure (1.11b). For the optimal λ we get a near perfect reconstruction, as can be seen in figure (1.11c). However, if we then increase λ , a few reflectors are not reconstructed, because the high λ requires the solution to be very sparse.

We see that the regularization parameter has a large influence on the reconstruction. If the parameter is too small the reconstruction is noisy, and if the parameter is too large we miss vital information about the solution. One might ask whether trying for a few different values and picking the best one is a good strategy. The problem is that for every parameter, we have to solve the entire inverse problem. However, in seismic exploration, solving certain inverse problems may take as long as a month, and in medical imaging, the image sometimes has to be produced in real time, which places constraints on the computational time. Moreover, so far we have only been able to determine the optimal regularization parameter given the ground truth. Therefore, there is a need for criteria to determine the regularization parameter, and for fast algorithms. It is not always

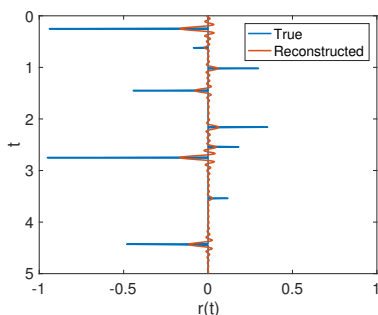
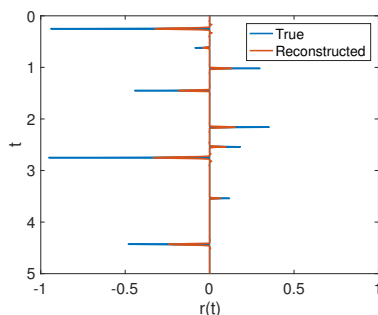
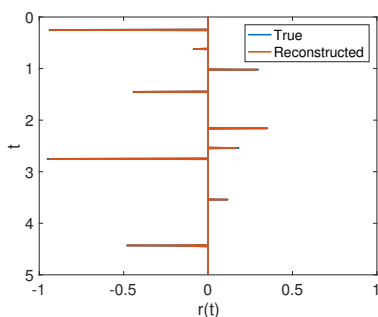
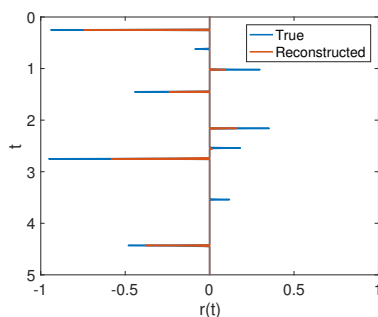
(a) Reconstruction for $\lambda = 0.00001$.(b) Reconstruction for $\lambda = 0.001$.(c) Reconstruction for $\lambda = 0.06$.(d) Reconstruction for $\lambda = 2$.

Figure 1.11

necessary to compute a solution with very high precision. An approximate solution that can be computed much faster is maybe even more valuable.

1.3 Outline and contributions

This thesis is organized as follows. In chapter 2 we present the necessary background theory for the contents of the rest of the thesis. We present some theory on regularization and describe the state-of-the-art algorithms that are used to solve linear inverse problems with different regularizers.

In chapter 3 we present an overview of some parameter selection methods for estimating the regularization parameter for the regularizer $\lambda\|x\|_2^2$, and discuss how to efficiently estimate them. We compare two different state-of-the-art model order reduction methods, namely Krylov based model reduction and the Randomized Singular Value Decomposition. The goal is to use these model order reduction methods to construct a low-dimensional space to estimate the parameter λ , for the particular regularizer $\mathcal{R}(x) = \lambda\|x\|_2^2$.

In chapters 4 and 5 we discuss an inverse problem from geophysics,

Multi-Dimensional Deconvolution (MDD). This is a problem that arises in multiple problems in geophysics. Two wavefields that hit a reflector from above and below respectively, are related to the impulse response of the reflector via convolution, and the objective is to solve for the impulse response. The MDD problem can be solved either in the frequency domain or in the time domain. In chapter 4 we solve the problem in the frequency domain. For every frequency, we have to solve a linear inverse problem with multiple right-hand sides, that can be written as $AX = B$. Every right-hand side corresponds to a particular receiver, that records the seismic data. To exploit the abundance of data for a particular model A , we propose using block Krylov methods and show that this is a competitive alternative to standard methods that are used to solve the MDD problem.

The impulse response is bound by two constraints that have to be satisfied. These are not structural constraints, where we assume that we (approximately) know something about the structure of the solution, but instead are physical constraints, that have to be satisfied, in order for the solution to be in accordance with the laws of physics. In chapter 5, we describe these constraints from a mathematical viewpoint, and discuss how they can be incorporated into the optimization problem. In order to deal with the constraints, we propose to solve the problem in the time domain. Hence, we solve for all frequencies at once, which amounts to solving a large linear system where the operator A is block-diagonal. We show that additional regularization is needed to stabilize the solution. We discuss the additional use of Tikhonov regularization, and show that parameter selection methods do not work for this particular problem. Our experiments are on non-inverse crime data.

In chapter 6 we investigate algorithms for solving linear inverse problems with a non-differentiable regularizer. We extend the analysis on a recently introduced algorithm, called Sparse Relaxed Regularized Regression (SR3) [130]. The algorithm essentially replaces the regularizer with its Moreau envelope by introducing an auxiliary variable, hence the term "relaxed". SR3 is an algorithm that applies to a large class of regularizers. Basically, it applies to any regularizer $\mathcal{R}(Lx)$ for which the proximal operator of $\mathcal{R}(x)$ exists. The analysis in [130] shows that if $L = I$, SR3 forms a new system with improved spectral properties that leads to faster convergence. We extend this analysis to show what happens if $L \neq I$. We analyze the relation between the Pareto curve of the original problem and the relaxed problem, and quantify the distance between the curves. Furthermore, we show that this Pareto curve can be used to get an estimate of the correct regularization parameter. Finally, we propose an inexact version of SR3 with an automated stopping criterion that makes the algorithm suitable for large-scale optimization.

Lastly, in chapter 7 we draw our conclusions and give a short outlook on possible further directions of research.



Preliminaries



This chapter is intended to provide some more background and explanation on the topics in the subsequent chapters. Section 2.2 provides background for the material treated in chapters 3 and 5 and section 2.3 provides background material for chapter 6.

2.1 The pseudoinverse

The solution to

$$Ax = b,$$

where $A \in \mathbb{R}^{m \times n}$ can not be directly obtained unless A is invertible, in which case we have $x = A^{-1}b$ and the solution is unique. If A is not square or full rank we can not hope to get a unique solution, but we are satisfied with a solution $z = A^\dagger b$ such that $Az = b$, or, if no such z exists, a "solution" such that $\|Az - b\|_2$ is minimized. The operator A^\dagger is called a *pseudoinverse*. The most popular pseudoinverse is the *Moore-Penrose inverse*, which satisfies the following four conditions:

1. $A^\dagger AA^\dagger = A^\dagger$
2. $AA^\dagger A = A$
3. $(AA^\dagger)^T = AA^\dagger$
4. $(A^\dagger A)^T = A^\dagger A$

Let $A \in \mathbb{R}^{m \times n}$. If $m < n$ the problem is underdetermined, and there exist many solutions. If A is full rank, the Moore-Penrose inverse provides the minimum-norm solution. On the other hand, if $m > n$, the problem is overdetermined and there exists no solution, unless $b \in \mathcal{R}(A)$. In this case, if A is full rank, the Moore-Penrose inverse provides the minimum residual "solution".

Assume that A is full rank. If $m < n$ the Moore-Penrose inverse provides a right inverse, i.e. $AA^\dagger = I_m$, where

$$A^\dagger = A^T(AA^T)^{-1}.$$

If $m > n$ the Moore-Penrose inverse provides a left inverse, i.e. $A^\dagger A = I_n$, where

$$A^\dagger = (A^T A)^{-1} A^T.$$

2.2 Tikhonov regularization

In this section we consider linear inverse problems of the form

$$\min_x \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|Lx\|_2^2, \quad (2.1)$$

with $A \in \mathbb{R}^{m \times n}$, $L \in \mathbb{R}^{p \times n}$. A nice feature of this type of regularization is that the objective is differentiable. Therefore, the solution can be given in closed-form:

$$x_\lambda = (A^T A + \lambda L^T L)^{-1} A^T b. \quad (2.2)$$

We distinguish the case $L = I$ and $L \neq I$. If $L = I$, (2.2) reduces to

$$x_\lambda = (A^T A + \lambda I)^{-1} A^T b. \quad (2.3)$$

The Singular Value Decomposition

The Singular Value Decomposition (SVD) of A [42], is given by

$$A = U\Sigma V^T,$$

where $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r), \sigma_i \geq 0$ is a diagonal matrix and U and V are orthonormal. The SVD has the following important properties:

1. If Σ has k nonzero values then $\text{rank}(A) = k$.
2. The first k columns of U form a basis for $\mathcal{R}(A)$ and the last $n - k$ columns of V form a basis for $\mathcal{N}(A)$.
3. σ_i^2 are the eigenvalues of the matrices $A^T A = V\Sigma^T \Sigma V^T$ and $AA^T = U\Sigma \Sigma^T U^T$.

These properties make the SVD a very useful tool for analyzing inverse problems, as the singular values give information about the invertibility of the matrix A . Moreover, the SVD is used to calculate the Moore-Penrose inverse:

$$A^\dagger = V\Sigma^\dagger U^T = V\text{diag}(1/\sigma_r, \dots, 1/\sigma_1, 0, \dots, 0)U^T.$$

The SVD can also be used to analyze the effect of Tikhonov regularization. Plugging in the SVD of A in (2.2) gives

$$x_\lambda = V(\Sigma^T \Sigma + \lambda I)^{-1} \Sigma^T U^T b = \sum_{i=1}^n v_i \frac{\sigma_i^2}{\sigma_i^2 + \lambda} \frac{u_i^T b}{\sigma_i}. \quad (2.4)$$

If σ_i is large then $\frac{\sigma_i}{\sigma_i^2 + \lambda} \approx \sigma_i^{-1}$ and if σ_i is small then $\frac{\sigma_i}{\sigma_i^2 + \lambda} \approx \lambda^{-1}$. Therefore, Tikhonov regularization acts as a filter for the small singular values that make the inversion unstable.

Closely related to Tikhonov regularization is the Truncated Singular Value Decomposition (TSVD), which produces a regularized solution via

$$x_k = \sum_{i=1}^k v_i \frac{u_i^T b}{\sigma_i}. \quad (2.5)$$

Here, rather than smoothing by adding a constant to counteract the inadvertent effects of the small singular values, we simply truncate them. The index k plays the role of the regularization parameter.

The Generalized Singular Value Decomposition

If $L \neq I$ the SVD is no longer useful for analyzing the system, because A and L do not diagonalize under the same basis V , the right singular vectors of A . A natural tool to analyze this class of problems is the Generalized Singular Value Decomposition (GSVD) [96].

Definition 1 (The Generalized Singular Value Decomposition (GSVD)). *The GSVD of a matrix pair (A, L) , $A \in \mathbb{R}^{m \times n}$, $L \in \mathbb{R}^{p \times n}$, is given by $A = U\Sigma X$, $L = V\Gamma X$, where*

$$\Sigma = \begin{bmatrix} \Sigma_p & 0 \\ 0 & I_{n-p} \\ 0 & 0 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} \Gamma_p & 0 \end{bmatrix} \quad \text{for } m \geq n, p \leq n,$$

and

$$\Sigma = \begin{bmatrix} 0 & \Sigma_m \end{bmatrix}, \quad \Gamma = \begin{bmatrix} I_{n-m} & 0 \\ 0 & \Gamma_m \\ 0 & 0 \end{bmatrix} \quad \text{for } m < n, p > n.$$

The matrices Σ_r and Γ_r (where $r = p$ or $r = m$) are $r \times r$ diagonal matrices satisfying $\Sigma_r^T \Sigma_r + \Gamma_r^T \Gamma_r = I_r$, X is invertible and U and V are orthonormal. Moreover, we have the following ordering of the singular values:

$$\begin{aligned} 0 \leq \gamma_r \leq \dots \leq \gamma_1 \leq 1, \\ 0 \leq \sigma_1 \leq \dots \leq \sigma_r \leq 1. \end{aligned}$$

Plugging the GSVD into (2.1) gives

$$x_\lambda = X^{-1} (\Sigma^T \Sigma + \lambda \Gamma^T \Gamma)^{-1} U^T b. \quad (2.6)$$

The usefulness of the GSVD for generalized Tikhonov regularization lies in the fact that it diagonalizes A and L under a common basis X . The question arises whether there exists an iterative method that approximates the GSVD like LSQR does with the SVD. The answer is yes and this algorithm is called the Joint BiDiagonalization algorithm (JBD) [72]. However, the iterations do not comprise of matrix-vector multiplications, but require the solution to linear systems, which makes the method expensive. There exists no iterative method based solely on matrix-vector multiplications that diagonalizes the matrix pair (A, L) .

The standard-form transformation

There are more or less two options to dealing with general L . The first is to use the so-called *standard-form transformation* [34, 58], which transforms (2.1) to

$$\min_y \|AL_A^\dagger y - b\|_2^2 + \lambda \|y\|_2^2, \quad x = L_A^\dagger y + x_{\mathcal{N}}, \quad x_{\mathcal{N}} = (A(I - L^\dagger L))^\dagger b,$$

where L_A^\dagger is called the *A-weighted pseudoinverse* and is given by

$$L_A^\dagger = (I - (A(I - L^\dagger L))^\dagger A)L^\dagger.$$

The standard-form transformation may be derived as follows. Apply the substitution $y = Lx$. The trick is to split the solution into two parts $x = x_{\mathcal{R}} + x_{\mathcal{N}}$ by a projection, where $x_{\mathcal{N}} \in \mathcal{N}(L)$. The component $x_{\mathcal{R}}$ has to satisfy $y = Lx_{\mathcal{R}}$, and should thus be obtained by an operator L^\dagger that has to satisfy $LL^\dagger L = L$. Note that we do not care about $L^\dagger y$ containing components of $\mathcal{N}(L)$, because this is accounted for by $x_{\mathcal{N}}$. We have

$$\min_x \|Ax_{\mathcal{R}} + Ax_{\mathcal{N}} - b\|_2^2 + \lambda \|Lx_{\mathcal{R}}\|_2^2 \quad (2.7)$$

If $Ax_{\mathcal{R}}$ and $Ax_{\mathcal{N}}$ are orthogonal, then the optimization problem splits into two separate problems, one where the component in the nullspace is determined, $x_{\mathcal{N}}$, and another part in which $x_{\mathcal{R}}$ is determined. Requiring that $Ax_{\mathcal{R}}$ and $Ax_{\mathcal{N}}$ are orthogonal means that $x_{\mathcal{R}}$ and $x_{\mathcal{N}}$ are A -orthogonal. Note that we now impose the following two conditions on obtaining $x_{\mathcal{R}}$:

1. $x_{\mathcal{R}}$ solves $y = Lx_{\mathcal{R}}$, which means $x_{\mathcal{R}}$ is obtained by some generalized inverse L^\dagger satisfying $LL^\dagger L = L$.
2. $x_{\mathcal{R}}$ is A -orthogonal to $x_{\mathcal{N}}$.

Let W be a basis for $\mathcal{N}(L)$. Then AW is a basis for $\mathcal{AN}(L)$, and $AW(AW)^\dagger$ is an orthogonal projector onto this space. Then $I - AW(AW)^\dagger$ is its orthogonal complement, and hence $Ax_{\mathcal{R}}$ has to be projected onto this space, yielding a vector that is orthogonal to $\mathcal{AN}(L)$. This gives

$$(I - AW(AW)^\dagger)Ax = A(I - W(AW)^\dagger A)x.$$

If we now require

$$x_{\mathcal{R}} = (I - W(AW)^\dagger A)L^\dagger y := L_A^\dagger y,$$

$x_{\mathcal{R}}$ is A -orthogonal to $x_{\mathcal{N}}$ and $LL_A^\dagger L = L$. This is an *oblique projector* that projects onto W , orthogonal to $\mathcal{R}(A^T)$, or along $\mathcal{N}(A)$. This splits the problem into the parts

$$x_{\mathcal{R}} = \min_y \|A(I - W(AW)^\dagger A)L^\dagger y - b\|_2^2 + \lambda \|y\|_2^2 \quad (2.8)$$

$$x_{\mathcal{N}} = \min_x \|A(W(AW)^\dagger A)(I - L^\dagger L)x - b\|. \quad (2.9)$$

The regularization term in the second equation vanishes due to the fact that we have chosen $x_{\mathcal{N}}$ in the nullspace of L . Furthermore, Because $I - L^\dagger L = WW^T$, we have

$$\|A(W(AW)^\dagger A)(I - L^\dagger L)x - b\| = \|A(W(AW)^\dagger A)WW^T x - b\| = \|AWW^T x - b\|_2^2. \quad (2.10)$$

It remains to show that $L_A^\dagger = (I - W(AW)^\dagger A)L^\dagger$. We have $W(AW)^\dagger = (AWW^T)^\dagger$ and WW^T is a projector onto $\mathcal{N}(L)$, as is $I - L^\dagger L$, which shows the equivalence.

The standard-form transformation has an elegant representation in terms of the GSVD of (A, L) . In this case the standard-form system is given by

$$\min_y \|U\Sigma\Gamma^\dagger V^T y - b\|_2^2 + \lambda \|y\|_2^2, \quad x_{\mathcal{N}} = X^{-1} \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix} U^T b.$$

It follows that the SVD of the operator AL_A^\dagger is given by $U\Sigma\Gamma^\dagger V^T$.

Krylov methods

A similar approach to Tikhonov regularization is to use *iterative regularization*. A popular class of methods for this purpose are *Krylov methods* [42]. Krylov methods are designed to iteratively solve systems of the form

$$Ax = b,$$

by constructing the Krylov subspace $\mathcal{K}_k(A, b) = \text{span}\{b, Ab, \dots, A^{k-1}b\}$, where k is the iteration number. Well-known examples of Krylov methods are CG [65], LSQR

[97] and GMRES [101]. For inverse problems, LSQR is popular, which, at each iteration, minimizes

$$x_k = \min_{x \in \mathcal{K}_k(A^T A, A^T b)} \|Ax - b\|,$$

without explicitly forming $A^T A$. For ill-posed problems, LSQR exhibits *semiconvergence* [57]. Initially, the error goes down, as does the residual, but after a certain optimal k_* , the error increases whereas the residual keeps decreasing. Therefore, for these regularization methods, the iteration number plays the role of the regularization parameter.

The Krylov subspace \mathcal{K}_k is inherently unstable because the factors $A^j b$ become linearly dependent. Therefore, algorithms based on Krylov subspaces generally construct an orthonormal basis for the Krylov subspace. The LSQR algorithm is based on the Lanczos bidiagonalization algorithm [42]. It constructs two orthonormal bases such that

$$AV_k = U_k B_{k+1,k} \tag{2.11}$$

$$A^T U_k = V_k B_{k+1,k}^T + v e_k^T. \tag{2.12}$$

The columns of V_k form an orthonormal basis for the Krylov subspace $\mathcal{K}_k(A^T A, A^T b)$ and the columns of U_k form an orthonormal basis for the Krylov subspace $\mathcal{K}_k(AA^T, b)$. The matrix $B_{k+1,k} \in \mathbb{R}^{k+1,k}$ is lower bidiagonal. This can be used to solve $\min_{x \in \mathcal{K}_k} \|Ax - b\|_2^2$ as follows:

$$\begin{aligned} \min_{x \in \mathcal{K}_k(A^T A, A^T b)} \|Ax - b\|_2^2 &= \min_{y=V_k x} \|AV_k y - b\|_2^2 \\ &= \min_{y=V_k x} \|U_k B_{k+1,k} y - b\|_2^2 \\ &= \min_{y=V_k x} \|B_{k+1,k} y - \|b\| e_1\|_2^2, \end{aligned}$$

where $U_k^T b = \|b\| e_1$ follows from orthogonality and the fact that $u_1 = b/\|b\|$ by choice. LSQR solves the system $\min_{y=V_k x} \|B_{k+1,k} y - \|b\| e_1\|_2^2$ without storing U_k and V_k and by updating the solution at every iteration using Givens rotations. Similarly, Lanczos bidiagonalization can be used to approximate the solution to (2.1), which yields the following system

$$\min_{y=V_k x} \|B_{k+1,k} y - \|b\| e_1\|_2^2 + \lambda \|y\|_2^2. \tag{2.13}$$

The approximate solution is given by

$$x_{k,\lambda} = V_k (B_{k+1,k}^T B_{k+1,k} + \lambda I_k)^{-1} U_k^T b.$$

This combination of Tikhonov regularization with an iterative method like Lanczos bidiagonalization is called *hybrid regularization* [49, 57, 73]. It has the added benefit of counteracting the effect of semiconvergence. Note that the subspace $\mathcal{K}_k(A^T A, A^T b)$ is independent of the choice of λ , which allows for rapid evaluation of the solution $x_{k,\lambda}$ for all λ . However, it remains unclear which choice of k leads to

a satisfactory solution. Generally, for ill-posed problems, k is small compared to n , but there is no criterion to select k .

There exists an alternative, but closely related approach, to hybrid regularization, based on evaluating criteria for selecting the regularization parameter. Most criteria, on the discrete level, are of the form

$$\min_{\lambda} u^T f_{\lambda}(W)u \quad \text{or} \quad \text{find } \lambda \text{ s.t. } u^T f_{\lambda}(W)u = h(\delta),$$

where δ is the noise level and h is some function, and W is either $A^T A$ or AA^T . The function f_{λ} will be of the form

$$f_{\lambda}(x) = (x + \lambda)^{-p}, p \in \mathbb{N}. \quad (2.14)$$

At the heart of this approach lies the observation that

$$u^T f_{\lambda}(W)u = \int_a^b f(x) d\omega(x),$$

where

$$\omega(\lambda) = \begin{cases} 0 & \text{if } \lambda < a = \lambda_n \\ \sum_{j=i}^n [W^T u]_j^2 & \text{if } \lambda_{i+1} \leq \lambda < \lambda_i \\ \sum_{j=1}^n [W^T u]_j^2 & \text{if } \lambda \geq b = \lambda_1 \end{cases},$$

This integral can be approximated by a *quadrature rule*, i.e.

$$\int_a^b f(x) d\omega(x) \approx \sum_{i=1}^k w_i f(x_i) := I_k(f).$$

The w_i are called the weights and the x_i are called the nodes of the quadrature. The key idea is to use Gauss quadrature to generate the nodes and weights. The reason for this is the deep relation between Gauss quadrature and the Lanczos process, namely that the nodes and weights of Gauss quadrature are obtained from the eigendecomposition of the tridiagonal matrix $T_k = B_{k+1,k}^T B_{k+1,k}$ or $T_k = B_{k+1,k} B_{k+1,k}^T$, obtained from the Lanczos process applied to A [107]. Gauss quadrature is an optimal quadrature rule, in the sense that it is exact for all polynomials up to degree $2n$ for n quadrature nodes, and there exists no quadrature rule that is exact for all polynomials of degree larger than $2n$ [107]. There exist multiple variants of Gauss quadrature that fix one or more nodes and hence lose a degree of freedom. One of them is the Gauss-Radau rule, that fixes precisely one node, and is therefore exact for polynomials of degree $2n - 1$ or less.

The error for these rules has an explicit formula, given by

$$E_{G_k}(f) = \frac{f^{(2k)}(\xi)}{(2k)!} \int_a^b \left[\prod_{i=1}^k (x - x_i) \right]^2 d\omega(x).$$

$$E_{GR_k}(f) = \frac{f^{(2k-1)}(\xi)}{(2k-1)!} \int_a^b (x - a) \left[\prod_{i=1}^{k-1} (x - x_i) \right]^2 d\omega(x).$$

For the functions of the form (2.14) we have $f_\lambda^{(2j)} > 0$ and $f_\lambda^{(2j-1)} < 0$, which means that the Gauss rule and the Gauss-Radau rule are a lower and an upper bound, respectively, for $u^T f_\lambda u$. These bounds decrease/increase monotonically, as was shown in [50]. This approach was first presented in [43], and later extended in [44]. This allows to design a stopping criterion to find a k_* such that the bounds are close to within a user specified tolerance. From this an approximate solution x_{λ, k_*} may be constructed, using the already computed $B_{k+1, k}$. The solution and the quadrature bounds are related in the following way:

Theorem 1 ([22]). *Let $\lambda > 0$ be a desired value of the regularization parameter and let $x_{\lambda, k}$ be an associated approximate solution to (2.1) of the form $x_{\lambda, k} = V_k y$ determined by the Galerkin equation*

$$V_k(A^T A + \lambda I_n)V_k y = V_k^T A^T b.$$

Then

$$\|x_{\lambda, k}\| = \sqrt{\hat{I}_{G_k}}$$

and

$$\|Ax_{\lambda, k} - b\| = \sqrt{I_{GR_{k+1}}}.$$

Generalized Krylov methods

The operator L_A^\dagger is expensive to compute and therefore applying the standard-form transformation is not suitable for large-scale problems. There exist a few other iterative approaches to solve (2.1) with a general linear operator that we will briefly outline here.

In [67], the authors propose an approach similar to Lanczos bidiagonalization applied to both A and L that constructs the Krylov subspaces consisting of all the terms of the binomial products arising from $(A^T A + L^T L)^j, j = 0, \dots, k - 1$. This builds matrices H and K that satisfy the relations

$$AV_k = U_{k+1}H_{k+1, k}, \quad LV_k = W_k K_{k, k}$$

$$A^T U_k = V_{2k-2} H_{k, 2k-2}^T, \quad L^T W_k = V_{2k+1} K_{k, 2k+1}^T$$

where the matrix H is upper Hessenberg and K is upper triangular, but both with a particular sparsity pattern. The projected problem now becomes

$$\min_y \|H_{k+1, k} y - b\|_2^2 + \lambda \|K_{k, k} y\|_2^2.$$

It should be noted that, unlike with Lanczos bidiagonalization, the amount of work done at each iteration increases.

Another approach, presented in [66], is based on the Krylov subspace generated by A , after which the QR decomposition

$$LV_k = QR$$

is obtained. Generally, the product LV_k can be calculated efficiently because L is the first order finite difference operator in most applications, an operator which is highly sparse. The QR decomposition can then be computed in reasonable time if k is (relatively) small.

An approach similar to this idea was introduced in [76] where a subspace method is used. At each iteration, the subspace is enlarged by computing a new vector that is orthogonal to the current subspace. This method was extended to include multiple regularizers in [133].

2.3 Non-smooth regularization

In this section we describe the mathematical tools that are used to solve composite optimization problems of the form

$$\min_x h(x) := \min_x (f(x) + g(x)). \quad (2.15)$$

We will focus on the specific case $f(x) = \frac{1}{2}\|Ax - b\|_2^2$ and $g(x) = \lambda\|Lx\|_1$, but the theory will hold for more general f and g , specifically any convex, smooth f and convex but nonsmooth g . An extensive overview of the material here may be found in [18, 31], on which this section is based.

Since g is non-differentiable, gradient based methods are not applicable. We will assume that at least the subdifferential p of g exists, where the subdifferential is defined as

$$\partial g(x) = \{(x, p) \mid \forall y \in \text{dom}(g), g(y) \geq g(x) + \langle p, y - x \rangle\}. \quad (2.16)$$

Hence, if g is at differentiable at x then $\{\partial g(x)\} = \nabla g(x)$. The subdifferential of $h(x) := f(x) + g(x)$ where f is differentiable is

$$\partial h(x) = \nabla f(x) + \partial g(x).$$

Similar to gradient descent, we could use subgradient descent to find the minimizer of the function h . However, the convergence rate of subgradient descent is considerably lower than the convergence rate of gradient descent, making it an unattractive option. An alternative method is based on the proximal operator, defined as:

$$\text{prox}_f(v) = \underset{x}{\text{argmin}} \left(f(x) + \frac{1}{2}c\|x - v\|_2^2 \right).$$

The minimizers of f are related to the fixed point of prox_{cf} in the following way:

Proposition 1. $x_* = \text{prox}_f(x_*)$ if and only if x_* minimizes f .

Given that the subdifferential of f exists, prox_{c_f} can be evaluated as follows:

$$\begin{aligned} \text{prox}_f(v) &= \underset{x}{\operatorname{argmin}} \left(f(x) + \frac{1}{2} \|x - v\|_2^2 \right) \\ &\Rightarrow 0 \in \partial f(x) + 1/c(x - v) \\ &\Rightarrow v \in x + c\partial f(x) = (I + c\partial f)(x) \\ &\Rightarrow x = (I + c\partial f)^{-1}(v). \end{aligned}$$

The operator $I + c\partial f$ is called the *resolvent*, and will be denoted by R_f . Note that evaluating the proximal operator requires solving an entire optimization problem by itself. However, in some cases, this optimization problem has a simple closed form solution, which allows for efficient evaluation.

An algorithm for $g(x) = \lambda \|x\|_1$: Iterative Soft-Thresholding Algorithm

We now turn to the case (2.15) where we have to minimize the sum of two functions. Instead of finding the proximal operator of the composed function $h(x)$, the general strategy is to apply the proximal operators of f and g separately. Algorithms based on this strategy are called splitting algorithms. The simplest splitting is called *forward-backward splitting* and can be derived as follows:

$$\begin{aligned} 0 &\in c\nabla f(x) + c\partial g(x) \\ 0 &\in c\nabla f(x) + x - x + c\partial g(x) \\ (I - c\nabla f)(x) &\in (I + c\partial g)(x) \\ x^* &= (I + c\partial g)^{-1}(I - c\nabla f)(x^*) \\ x^* &= R_g(I - c\nabla f)(x^*) \end{aligned}$$

If $g = \lambda \|\cdot\|_1$ then the proximal operator acts component wise, and the elements are given by

$$\text{prox}_g(v_i) = \begin{cases} v_i - \lambda & \text{if } v_i > \lambda \\ 0 & \text{if } |v_i| < \lambda \\ v_i + \lambda & \text{if } v_i < -\lambda \end{cases},$$

an operation which is called *soft-thresholding*. The above algorithm is called the Iterative Shrinkage Thresholding Algorithm, or ISTA. Note that the operator $I - c\nabla f$ is the gradient step.

There is a version of this algorithm called Fast Iterative Shrinkage Thresholding Algorithm, FISTA, which chooses a particular combination of the previous two iterates to form the new iterate. This trick is called *Nesterov acceleration*, and leads to optimal convergence for first-order methods.

An algorithm for $g(x) = \lambda \|Lx\|_1$: the Alternating Direction Method of Multipliers

If $g = \lambda \|Lx\|_1$ then prox_g no longer has a closed form expression and we can no longer apply FISTA. For this class of problems there exist different splitting

algorithms that can cope with the presence of the operator L , resulting in the Alternating Direction Method of Multipliers (ADMM). The splitting method used to derive the ADMM is called *Douglas-Rachford splitting*. Instead of applying the resolvent, it applies a fixed point operator called the *Cayley operator*, defined as

$$C_f := 2R_f - I = (I - c\partial f)(I + c\partial f)^{-1}. \quad (2.17)$$

Douglas-Rachford iteratively applies

$$z_{k+1} = \left(\frac{1}{2}I + \frac{1}{2}C_f C_g \right) z_k, \quad x_{k+1} = R_g z_{k+1}. \quad (2.18)$$

Using the definition of the Cayley operator this can be written as follows:

$$\begin{aligned} x_{k+1} &= R_g(z_k) \\ y_{k+1} &= R_f(2x_{k+1} - z_k) \\ z_{k+1} &= z_k + y_{k+1} - x_{k+1}. \end{aligned}$$

ADMM is based on a splitting strategy for the following reformulation of (2.15)

$$\min_{x,z} f(x) + g(z) \quad (2.19)$$

$$\text{s.t. } Ax + Bz = b. \quad (2.20)$$

The choice $A = I$, $B = -L$ and $b = 0$ leads to (2.15). The *Lagrangian* is given by

$$\begin{aligned} \mathcal{L}(x, z, \lambda) &= f(x) + g(z) + \lambda^T (Ax + Bz - c) \\ &= \{f(x) + \lambda^T Ax\} + \{g(z) + \lambda^T Bz\} - \lambda^T c \\ &:= \mathcal{L}_1(x, \lambda) + \mathcal{L}_2(z, \lambda) - \lambda^T c. \end{aligned}$$

The Karush-Kuhn-Tucker conditions then state that the optimal solution is found by evaluating

$$\begin{aligned} \max_{\lambda} \min_{x,z} \mathcal{L}(x, z, \lambda) &= \max_{\lambda} \left(\left\{ \min_x \mathcal{L}_1(x, \lambda) - \lambda^T c \right\} + \left\{ \min_z \mathcal{L}_2(z, \lambda) \right\} \right) \\ &= \max_{\lambda} -f^*(-A^T \lambda) - \lambda^T c - g^*(-B^T \lambda) =: \max_y h(y), \end{aligned}$$

where f^* denotes the conjugate function, defined as

$$f^*(v) = \sup_x (v^T x - f(x)). \quad (2.21)$$

Applying Douglas-Rachford splitting to the dual problem yields the ADMM. We have

$$\partial h(y) = \{A\partial f^*(-A^T \lambda) - c\} + \{B\partial g^*(-B^T \lambda)\} := k_1(\lambda) + k_2(\lambda), \quad (2.22)$$

which, by applying Douglas-Rachford splitting, yields the algorithm

$$\begin{aligned} x_{k+1} &= R_{k_1}(z_k) \\ y_{k+1} &= R_{k_2}(2x_{k+1} - z_k) \\ z_{k+1} &= z_k + y_{k+1} - x_{k+1}. \end{aligned}$$

The ADMM is generally not presented in this form. To obtain the conventional form, one has to rewrite the iterations by writing the resolvent as an explicit minimization problem, which we will describe now. Consider the problem

$$\begin{aligned} \min_x & f(x) \\ \text{s.t.} & Ax = b. \end{aligned}$$

The Lagrangian is given by

$$\mathcal{L}(x, y) = f(x) + y^T(Ax - b),$$

and the dual problem is given by

$$\max_y h(y) := \max_y (-f^*(-A^T y) - y^T b).$$

The subdifferential of h is given by

$$\partial h(y) = A\partial f^*(-A^T y) - b.$$

Using the optimality criteria for the Lagrangian we know

$$\begin{aligned} \mathcal{L}_x &= \partial f(x) + A^T y \ni 0 \\ \iff &x \in (\partial f)^{-1}(-A^T y) \\ \iff &x \in \partial f^*(-A^T y), \end{aligned}$$

This means that

$$\partial h(y) = Ax - b.$$

To find the point that maximizes h , or equivalently, minimizes $-h$, we apply the proximal operator of h . Recall that evaluating the proximal operator is equivalent to applying the resolvent. The resolvent $R_h(y)$ yields

$$w = R_h(y) = (I + c\partial h)^{-1}y \iff w + c(\partial h)(w) = y \iff w + c(Ax - b) = y.$$

Substituting the last expression in \mathcal{L}_x yields

$$\mathcal{L}_x = \partial f(x) + A^T w + cA^T(Ax - b).$$

Hence, w is a fixed point of $R_h(y)$ if the following two conditions are satisfied

$$\begin{aligned} 0 &\in \partial f(x) + A^T y + cA^T(Ax - b) \\ w &= y - c(Ax - b). \end{aligned}$$

This means that x minimizes

$$\mathcal{L}_c(x, y) := f(x) + y^T(Ax - b) + \frac{c}{2}\|Ax - b\|_2^2, \quad (2.23)$$

which is called the *augmented Lagrangian*. This shows that the proximal operator, and hence the resolvent, for the dual functions obtained in (2.22) are equivalent to

the minimization problem (2.23). Using this equivalence between the resolvent and the augmented Lagrangian, note that the first step of Douglas-Rachford splitting can be rewritten as

$$\begin{aligned}\tilde{z}_{k+1} &= \operatorname{argmin} \{g(z) + y_k^T Bz + c\|Bz\|^2\} \\ z_{k+1} &= y_k - cB\tilde{z}_{k+1}.\end{aligned}$$

Equivalently, the second step can be rewritten as

$$\begin{aligned}\tilde{x}_{k+1} &= \operatorname{argmin} \{f(x) + z_{k+1}^T (Ax - b) + c\|Ax - b\|^2\} \\ x_{k+1} &= z_{k+1} - c(A\tilde{x}_{k+1} - b).\end{aligned}$$

Finally, the last step simply yields

$$y_{k+1} = y_k + A\tilde{x}_{k+1} - c + B\tilde{z}_{k+1}.$$

Note that z_{k+1} and x_{k+1} are not explicitly needed. After the substitution

$$y_k = cu_k + c(Ax_k - b),$$

writing $z_{k+1} = \tilde{z}_{k+1}$ and $x_{k+1} = \tilde{x}_{k+1}$ and rearranging the order, we get

$$x_{k+1} = \operatorname{argmin}_x \left\{ f(x) + \frac{c}{2} \|Ax + Bz_k - b + u_k\|_2^2 \right\} \quad (2.24)$$

$$z_{k+1} = \operatorname{argmin}_z \left\{ g(z) + \frac{c}{2} \|Ax_{k+1} + Bz - b + u_k\|_2^2 \right\} \quad (2.25)$$

$$u_{k+1} = u_k + Ax_{k+1} + Bz_{k+1} - b. \quad (2.26)$$

2.3.1 Sparse Relaxed Regularized Regression (SR3)

ADMM can handle any regularizer of the form $\lambda\|Lx\|_1$, but convergence may be slow. FISTA achieves the optimal convergence rate for first order methods, but it can only be applied if $L = I$. Sparse Relaxed Regularized Regression is a recently introduced algorithm that, similar to ADMM, introduces an auxiliary variable to solve problems of the form (2.15):

$$\min_{x,y} \frac{1}{2} \|Ax - b\|_2^2 + \frac{\kappa}{2} \|Lx - y\|_2^2 + \mathcal{R}(y). \quad (2.27)$$

Note the similarity to the augmented Lagrangian in (2.23). However, in SR3, the Lagrange parameter is omitted. Rewriting (2.27) as

$$\min_x \frac{1}{2} \|Ax - b\|_2^2 + \left\{ \min_y \frac{\kappa}{2} \|Lx - y\|_2^2 + \mathcal{R}(y) \right\}, \quad (2.28)$$


we see that SR3 essentially replaces the regularizer with its Moreau envelope. If the regularizer is convex, like the ℓ_1 norm, the Moreau envelope has a smoothing effect. By minimizing out the variable x , we obtain a newly formed system

$$\min_x \frac{1}{2} \|F_\kappa y - g_\kappa\|_2^2 + \mathcal{R}(y), \quad (2.29)$$


where

$$\begin{aligned} H_\kappa &= (A^T A + \kappa L^T L), \\ F_\kappa &= \begin{bmatrix} \kappa A H_\kappa^{-1} L^T \\ \sqrt{\kappa} (I - L H_\kappa^{-1} L^T) \end{bmatrix}, \\ g_\kappa &= \begin{bmatrix} I - A H_\kappa A^T \\ \sqrt{\kappa} L H_\kappa^{-1} L^T \end{bmatrix}. \end{aligned}$$

Note that the operator L has been removed from the regularizer in (2.29). SR3 allows us to apply FISTA to problems with regularizers where the proximal operator of $\mathcal{R}(x)$ has a closed form solution, but the proximal operator of $\mathcal{R}(Lx)$ does not. Moreover, the operator F_κ has more desirable spectral properties, as we will show in chapter 5. This leads to faster convergence.



Comparing RSVD and Krylov
methods for linear inverse
problems



Abstract In this work we address regularization parameter estimation for ill-posed linear inverse problems with an ℓ_2 penalty. Regularization parameter selection is of utmost importance for all of inverse problems and estimating it generally relies on the experience of the practitioner. For regularization with an ℓ_2 penalty there exist a lot of parameter selection methods that exploit the fact that the solution and the residual can be written in explicit form. Parameter selection methods are functionals that depend on the regularization parameter where the minimizer is the desired regularization parameter that should lead to a good solution. Evaluation of these parameter selection methods still requires solving the inverse problem multiple times. Efficient evaluation of the parameter selection methods can be done through model order reduction. Two popular model order reduction techniques are Lanczos based methods (a Krylov subspace method) and the Randomized Singular Value Decomposition (RSVD). In this work we compare the two approaches. We derive error bounds for the parameter selection methods using the RSVD. We compare the performance of the Lanczos process versus the performance of RSVD for efficient parameter selection. The RSVD algorithm we use is based on the Adaptive Randomized Range Finder algorithm which allows for easy determination of the dimension of the reduced order model. Some parameter selection methods also require the evaluation of the trace of a large matrix. We compare the use of a randomized trace estimator versus the use of the Ritz values from the Lanczos process. The examples we use for our experiments are two model problems from geosciences.

This chapter is partially based on the following publication:

N.A. Luiken and T. van Leeuwen. Comparing RSVD and Krylov methods for linear inverse problems. *Computers & Geosciences*, 137:104427, 2020.

3.1 Introduction

Inverse problems are ubiquitous in the earth-sciences with applications ranging from seismology to seismic exploration. Often, these inverse problems are *ill-posed*, meaning that a unique, stable solution does not exist. Regularization is needed to render the problem well-posed. In this chapter we discuss finite-dimensional, linear inverse problems which can be posed as

$$\min_{\mathbf{m}} \|G\mathbf{m} - \mathbf{d}\|_2^2 + \lambda \|L\mathbf{m}\|_2^2, \quad (3.1)$$

where $G \in \mathbb{R}^{m \times n}$ is the *forward operator*; $\mathbf{d} \in \mathbb{R}^m$ denotes the data; $\mathbf{m} \in \mathbb{R}^n$ are the parameters of interest; $L \in \mathbb{R}^{p \times n}$ is the *regularization operator* and $\lambda \in \mathbb{R}_+$ is the *regularization parameter*. The regularization operator incorporates the prior information needed to make the problem uniquely solvable. Without loss of generality, we assume that $L = I$, as every problem of the form (3.1) can be transformed to this form [58]. The regularization parameter balances the prior information and information from the data. The solution can be written in closed form, given by

$$\hat{\mathbf{m}}_\lambda = G_\lambda \mathbf{d}, \quad (3.2)$$

with

$$G_\lambda = (G^T G + \lambda I)^{-1} G^T. \quad (3.3)$$

Although this expression is convenient for derivations, in practice it is usually not feasible to form the matrix G_λ explicitly. Therefore, the solution $\hat{\mathbf{m}}_\lambda$ is usually approximated using an iterative solver. A major issue in solving (3.1) is the selection of the regularization parameter λ . Methods for selecting the regularization parameter are called *parameter selection methods*. A complete overview and comparison of parameter selection methods is given in [9]. Parameter selection methods, generally, rely on repeatedly solving (3.1) and selecting the value of λ that satisfies some auxiliary criteria. These criteria usually involve minimizing a functional $V(\lambda)$ whose evaluation involves the solution of (3.1). This yields a $\hat{\lambda}$,

$$\hat{\lambda} = \underset{\lambda}{\operatorname{argmin}} V(\lambda). \quad (3.4)$$

Solving the inverse problem even once is costly and therefore, finding the optimal parameter $\hat{\lambda}$ is computationally intensive. In order to overcome this computational drawback, various methods for approximating $V(\lambda)$ have been proposed.

3.1.1 Approach

The general approach is to approximate $V(\lambda)$ in such a way that it is cheaper to evaluate and thus allows for efficient estimation of $\hat{\lambda}$. Evaluating V involves two main tasks; evaluating a weighted norm of a given vector

$$\mathbf{u}^T f_\lambda(A) \mathbf{u}, \quad (3.5)$$

and computing the trace of a matrix function

$$\operatorname{trace}(f_\lambda(A)). \quad (3.6)$$

Here A is a positive semi-definite matrix which is either $G^T G$ or $G G^T$ and $\mathbf{u} = \mathbf{d}$ or $\mathbf{u} = G^T \mathbf{d}$. $G \in \mathbb{R}^{m \times n}$ and for notational simplicity we write $A \in \mathbb{R}^{d \times d}$, where d is either m or n . An obvious approach is to replace G by a low-rank approximation G_k and use this to approximate (3.5) and (3.6). An important aspect is the approximation error, its influence on the approximation of V and ultimately on the estimated $\hat{\lambda}$. For most applications, it is not feasible to explicitly form a (truncated) Singular Value Decomposition (SVD) of G , so we will need to approximate the truncated SVD in order to obtain a reduced order model G_k . Traditionally, Krylov subspace methods have been very popular for this purpose [73]. These methods can be used to find upper and lower bounds for (3.5) as well [43, 44]. Recently, randomized techniques have gained popularity. The Randomized Singular Value Decomposition (RSVD) was used for defining the reduced forward operator G_k [127, 128, 129]. The trace (3.6) can be estimated using randomized trace estimation [68, 110].

3.1.2 Contributions and outline

In this chapter we compare the use of Krylov based subspace methods versus the use of RSVD for solving discrete inverse problems, specifically with regard to selecting the regularization parameter. We briefly present some parameter selection methods and review the Lanczos process and the RSVD and cite the relevant literature. For the Lanczos process we focus on the fact that we can obtain lower and upper bounds for the parameter selection methods. We provide error bounds for the parameter selection methods when approximated using the Truncated Singular Value Decomposition (TSVD) and the RSVD. We also discuss the use of Hutchinson's trace estimator for the parameter selection methods. We present a theorem that provides probabilistic bounds for the trace estimation combined with a low dimensional approximation with the Lanczos process, based on the work by [110]. We also show that obtaining guarantees for the accuracy of the trace estimator is too computationally expensive. We compare the randomized trace estimator to estimating the trace using the Ritz values obtained from the Lanczos process or the RSVD. In our numerical examples we present two examples from geosciences. The first example is a severely ill-posed problem and the second is a mildly ill-posed underdetermined problem. We discuss the performance of the Lanczos process and an adaptive RSVD algorithm for parameter selection and discuss the performance of the randomized trace estimator and the Lanczos/RSVD based trace estimator.

This chapter is organised as follows. In section 3.2 we review the necessary theory on parameter selection methods and the Lanczos process and the RSVD. In section 3.3 we show template algorithms to obtain a lower dimensional approximation for two parameter selection methods. In section 3.4 we discuss the performance of the algorithms for two model problems from geosciences. Lastly, in section 3.5 we draw our conclusions.

3.2 Theory

3.2.1 Parameter Selection Methods

In this section we review the parameter selection methods that we use in this work. Among the parameter selection methods there is an important distinction to be made: methods that require knowledge about the noise level in the data and/or the underlying model and methods that do not. Methods that do not require any knowledge on the model and/or the noise level on the data are sometimes called heuristic methods. We consider a standard deterministic approach, for a Bayesian approach to solving linear inverse problems see, e.g. [89, 92, 132]. The main method we cover that requires knowledge on the noise level is the Discrepancy Principle (DP). We use a stochastic Gaussian noise model of the form

$$\mathbf{d} = \mathbf{d}_{\text{true}} + \boldsymbol{\xi}, \quad (3.7)$$

where $\boldsymbol{\xi} \sim \mathcal{N}(0, \delta^2 I)$ is uncorrelated Gaussian noise with mean zero and variance δ^2 and \mathbf{d}_{true} is defined as $\mathbf{d}_{\text{true}} := G\mathbf{m}$, i.e., the true noiseless data. The methods we cover that do not require any knowledge on the noise level or the model are Generalised Cross Validation (GCV), Reginska's rule and the Quasi Optimality Criterion (QO). An overview of the functionals V corresponding to each method is shown in table (3.1).

Method	$V(\lambda)$	Noise estimate
DP	$(\lambda^2 \mathbf{d}^T (GG^T + \lambda I)^{-2} \mathbf{d} - \delta^2 m)^2$	Yes
GCV	$\frac{\lambda^2 \mathbf{d}^T (GG^T + \lambda I)^{-2} \mathbf{d}}{(m^{-1} \text{trace}(I - GG_\lambda))^2}$	No
Reginska's rule	$\mathbf{d}^T G (G^T G + \lambda I)^{-2} G^T \mathbf{d} \cdot \lambda^2 \mathbf{d}^T (GG^T + \lambda I)^{-2} \mathbf{d}$	No
QO	$\lambda^2 \mathbf{d}^T G (G^T G + \lambda I)^{-4} G^T \mathbf{d}$	No

Table 3.1: Parameter selection methods.

In order to express these in terms of the weighted norm (3.5) and trace (3.6) we use the following identities:

$$\|\widehat{\mathbf{m}}_\lambda\|^2 = \mathbf{d}^T G (G^T G + \lambda I)^{-2} G^T \mathbf{d}, \quad (3.8)$$

and

$$\|G\widehat{\mathbf{m}}_\lambda - \mathbf{d}\|^2 = \lambda^2 \mathbf{d}^T (GG^T + \lambda I)^{-2} \mathbf{d}. \quad (3.9)$$

A derivation of these identities is included in appendix (3.6). Below, we briefly discuss each method in detail.

Reginska's rule

Reginska's rule [99] is a variant of the well-known L-curve [53]. We choose to use Reginska's rule because it allows for an easier evaluation of the optimal λ by minimizing

$$V_{RR(\alpha)}(\lambda) = \left(\mathbf{d}^T G (G^T G + \lambda I)^{-2} G^T \mathbf{d} \right)^\alpha \cdot \lambda^2 \mathbf{d}^T (GG^T + \lambda I)^{-2} \mathbf{d} \quad (3.10)$$

It has been proven in [99] that if the L-curve has maximal curvature at $\hat{\lambda}$ and has a tangent with slope $\hat{\alpha}$, then $V_{RR(\hat{\alpha})}$ has a minimizer at $\hat{\lambda}$. In practice, α is generally chosen to be 1.

Generalized Cross Validation

Generalized Cross Validation (GCV) was first introduced by [39] as a method for choosing the regularization parameter and is an alternative to UPRE (section 2.4) when the noise level is not known. It is important to note that although the noise level need not be known, there is an underlying assumption of a white Gaussian noise model [122]. The GCV estimates the optimal λ by minimizing

$$V_{GCV}(\lambda) = \frac{\lambda^2 \mathbf{d}^T (GG^T + \lambda I)^{-2} \mathbf{d}}{(m^{-1} \text{trace}(I - GG_\lambda))^2}. \quad (3.11)$$

The idea behind GCV is that it tries to estimate λ in such a way that the data is explained well, while preventing overfitting. It is known the GCV has desirable statistical properties, but that it deals poorly with correlated noise. It also tends to undersmooth solutions. For further issues we refer the reader to [39], [53], [122], [118], [57]. There exist a few variants of the GCV that are in a sense weighted forms of the GCV that overcome some of the drawbacks of the GCV. All variants have been shown to be more stable than the GCV [26], [86], [87]) in the sense that they emphasize the generally flat minimum of the GCV by making it more pronounced. The Unbiased Predictive Risk Estimator (UPRE) [121], also known as Mallow's C_p [88], is based on the predictive risk. It is in a sense the predecessor of the GCV, as the GCV was developed as a noise-free alternative to the UPRE [122].

The Discrepancy Principle

The Discrepancy Principle is an easy to use method that was first introduced by [90]. The optimal λ found by the Discrepancy Principle is the λ for which the residual equals the noise level, i.e.

$$\mathbf{d}^T (GG^T + \lambda I)^{-2} \mathbf{d} = \eta \delta^2 m,$$

where $\eta \geq 1$ is a user-defined constant. The parameter η is introduced to prevent oversmoothing of the solution. We can cast this into the desired form by introducing

$$V_{DP}(\lambda) = \left(\mathbf{d}^T (GG^T + \lambda I)^{-2} \mathbf{d} - \eta \delta^2 m \right)^2.$$

It is known that the Discrepancy Principle generally tends to oversmooth the solution [61], i.e. the value for λ is too large. Another drawback is that the estimate of the noise level has to be accurate, and that small errors in the estimate can lead to large deviations in the solution [55].

Quasi-Optimality criterion

The quasi-optimality criterion is one of the first heuristic parameter choice criteria [8], [77], [78], [90]. The λ estimated by the quasi-optimality criterion is the minimizer of

$$V_{\text{QO}}(\lambda) = \lambda^2 \mathbf{d}^T G (G^T G + \lambda I)^{-4} G^T \mathbf{d}. \quad (3.12)$$

For a derivation of this expression we refer the reader to [35].

3.2.2 Model Order Reduction and trace estimation

In this section we review various methods for approximation of quantities of the form

$$W(A) = \mathbf{w}^T f_\lambda(A) \mathbf{w},$$

and

$$T(A) = \text{trace}(f_\lambda(A)),$$

where $f_\lambda(x) = (x + \lambda)^{-p}$, $p \in \mathbb{N}$ and $A \in \mathbb{R}^{d \times d}$ is a symmetric positive semi-definite (SPSD) matrix. We define a matrix function in the conventional sense. Given the eigenvalue decomposition $A = Q \Lambda Q^T$, the function is defined as $Q f_\lambda(\Lambda) Q^T$, where $f_\lambda(\Lambda)$ is a diagonal matrix with $f_\lambda(\lambda_i)$ as its entries.

Truncated SVD

In this section we provide bounds for the parameter selection rules based on the Truncated SVD [51]. They will be the basis for the error bounds derived for the RSVD which will be presented in section 3.2.3.

Theorem 2. Let $W(A) = \mathbf{w}^T f_\lambda(A) \mathbf{w}$ and $T(A) = \text{trace}(f_\lambda(A))$. Let $A = \sum_{i=1}^d \sigma_i^2 \mathbf{u}_i \mathbf{u}_i^T$ where $(\sigma_i^2, \mathbf{u}_i)$ denotes an eigenpair of A and let $A_k = \sum_{i=1}^k \sigma_i^2 \mathbf{u}_i \mathbf{u}_i^T$. Then the relative errors are bounded by

$$\frac{|W(A) - W(A_k)|}{W(A)} \leq (d - k) \frac{p}{\lambda^{p+1}} \frac{\sigma_{k+1}}{f_\lambda(\sigma_d)}. \quad (3.13)$$

$$\frac{|T(A) - T(A_k)|}{T(A)} \leq \frac{(d - k) f_\lambda(\sigma_{k+1})}{\sum_{i=1}^d f_\lambda(\sigma_i)}. \quad (3.14)$$

Proof. Using a standard error estimate using the Taylor expansion we obtain:

$$\begin{aligned} |W(A) - W(A_k)| &\leq \sup_x |f'_\lambda(x)| |\mathbf{w}^T (A - A_k) \mathbf{w}| \\ &= \sup_x |f'_\lambda(x)| |\mathbf{w}^T U_{d-k+1} \Sigma_{d-k+1} U_{d-k+1}^T \mathbf{w}| \\ &\leq \sup_x |f'_\lambda(x)| (d - k) \|\mathbf{w}\|^2 \sigma_{k+1} \end{aligned}$$

We have an explicit expression for $\sup_{x \geq 0} |f'_\lambda(x)|$, given by:

$$\sup_{x \geq 0} |f'_\lambda(x)| = \sup_x |-p(x + \lambda)^{-p-1}| = \frac{p}{\lambda^{p+1}}.$$

A different bound can be obtained by making the following observation:

$$\begin{aligned}\mathbf{w}^T f_\lambda(A) \mathbf{w} &= \text{trace}(\mathbf{w}^T f_\lambda(A) \mathbf{w}) \\ &= \text{trace}(f_\lambda(A) \mathbf{w} \mathbf{w}^T) \\ &= \text{trace}(U f_\lambda(\Sigma) U^T \mathbf{w} \mathbf{w}^T).\end{aligned}$$

Now using the fact that $\mathbf{w} \mathbf{w}^T$ is a rank 1 matrix with eigenvalue $\|\mathbf{w}\|^2$, we can use von Neumann's trace inequality to obtain:

$$\|\mathbf{w}\|^2 f_\lambda(\sigma_d) \leq \mathbf{w}^T f_\lambda(A) \mathbf{w} \leq \|\mathbf{w}\|^2 f_\lambda(\sigma_1).$$

Putting both inequalities together we obtain (3.13). For $T(A)$ we get

$$\frac{|T(A) - T(A_k)|}{T(A)} = \frac{\sum_{i=k+1}^d f_\lambda(\sigma_i)}{\sum_{i=1}^d f_\lambda(\sigma_i)} \leq \frac{(d-k)f_\lambda(\sigma_{k+1})}{\sum_{i=1}^d f_\lambda(\sigma_i)}$$

□

It is important to note that the above error estimate depends on λ . As $\lambda \rightarrow 0$, $f'_\lambda \rightarrow \infty$. However, given a certain $\lambda > 0$, there exists a bound for the derivative, but it will become large for small λ . This means that there is an inverse relation between k and λ : for large λ , k can be small, whereas for small λ , k has to be large.

Krylov methods and Gauss quadrature

The approach makes use of the fact that the quantity $\mathbf{w}^T f(A) \mathbf{w}$ can be written as an integral with a certain measure, i.e.

$$\mathbf{w}^T f_\lambda(A) \mathbf{w} = \int_a^b f_\lambda(x) d\omega(x), \quad (3.15)$$

where ω is a piecewise constant measure with discontinuities at the eigenvalues of A . A short, intuitive explanation of this equality is given in section 3.6. The integral can be approximated by a quadrature rule of the form

$$\int_a^b f_\lambda(x) d\omega(x) = \sum_{i=1}^k w_i f_\lambda(x_i) + E_k(f_\lambda) := I_k(f_\lambda) + E_k(f_\lambda), \quad (3.16)$$

where $I_k(f)$ denotes the approximation with k nodes and $E_k(f)$ the associated error. The w_i are the weights and the x_i are the nodes. The weights and nodes for the Gauss quadrature rule are chosen such that the quadrature rule is exact for all polynomials of degree $2k$. It can be shown that there is no quadrature rule that is exact for all polynomials of order larger than $2k$. A variant, the Gauss-Radau rule, fixes one node, which means that the Gauss-Radau rule is exact for polynomials up to degree $2k - 1$. The errors for the k -point Gauss rule (E_k) and the

k -point Gauss-Radau rule (\tilde{E}_k) are given by [107], [41]:

$$E_k(f) = \frac{f^{(2k)}(\xi_1)}{(2k)!} \sum_{i=1}^m \mathbf{u}_i^T \mathbf{w} \left[\prod_{j=1}^k (\sigma_i^2 - \theta_j^{(k)}) \right]^2, \quad (3.17)$$

$$\tilde{E}_k(f) = \frac{f^{(2k-1)}(\xi_2)}{(2k-1)!} \sum_{i=1}^m \mathbf{u}_i^T \mathbf{w} (\sigma_i^2 - a)^2 \left[\prod_{j=2}^k (\sigma_i^2 - \theta_j^{(k)}) \right]^2. \quad (3.18)$$

Recall that the parameter selection methods are functions of the form $f_\lambda(x) = (x + \lambda)^{-p}$ where $p \in \mathbb{N}$, typically, $p = 1, 2$ or 4 . The derivatives for this class of functions are:

$$f_\lambda^{(2k)}(x) = (-1)^{(2k)} p(p+1) \cdots (p+2k-1) (x+\lambda)^{-(p+2k)} > 0 \quad (3.19)$$

$$f_\lambda^{(2k-1)}(x) = (-1)^{(2k-1)} p(p+1) \cdots (p+2k-2) (x+\lambda)^{-(p+2k-1)} < 0 \quad (3.20)$$

The nodes and weights for the Gauss quadrature are obtained by the eigendecomposition of the tridiagonal matrix T_k , which can be obtained by Lanczos tridiagonalization with starting vector \mathbf{w} . Let $T_k = Q\Lambda Q^T$, then the nodes of the quadrature are given by the eigenvalues and the weights are given by the first entry of the corresponding eigenvector.

3.2.3 Evaluating the Gauss and Gauss-Radau rule

Let T_k be the tridiagonal matrix obtained by the Lanczos process with starting vector \mathbf{w} . For a general form $\mathbf{w}^T f(A) \mathbf{w}$ the k -point Gauss quadrature rule is given by [21]:

$$I_k(f) = \sum_{i=1}^k w_i f(x_i) = \|\mathbf{w}\|^2 \sum_{i=1}^k f(\lambda_i) (\mathbf{e}_1^T Q \mathbf{e}_i)^2 \quad (3.21)$$

$$= \|\mathbf{w}\|^2 \mathbf{e}_1^T Q^T f(\Lambda) Q \mathbf{e}_1 \quad (3.22)$$

$$= \|\mathbf{w}\|^2 \mathbf{e}_1^T f(T_k) \mathbf{e}_1. \quad (3.23)$$

The functions that have to be evaluated are either functions of the form $\mathbf{w}^T G(G^T G + \lambda I)^{-p} G^T \mathbf{w}$ or $\mathbf{w}^T (GG^T + \lambda I)^{-p} \mathbf{w}$. For functions of the form $\mathbf{w}^T (GG^T + \lambda I)^{-p} \mathbf{w}$ the matrix T_k is obtained by using the Lanczos bidiagonalization process with starting vector \mathbf{b} . Let B_k denote the lower bidiagonal matrix obtained by the Lanczos bidiagonalization algorithm and let \bar{B}_k be B_k with its last column removed. Then the Gauss and Gauss-Radau rules are obtained by [21]:

$$I_k(f) = \|\mathbf{w}\|^2 \mathbf{e}_1^T f(B_k B_k^T) \mathbf{e}_1, \quad (3.24)$$

$$\tilde{I}_k(f) = \|\mathbf{w}\|^2 \mathbf{e}_1^T f(\bar{B}_{k-1} \bar{B}_{k-1}^T) \mathbf{e}_1. \quad (3.25)$$

For functions of the form $\mathbf{w}^T G(G^T G + \lambda I)^{-p} G^T \mathbf{w}$ the matrix T_k can still be obtained by the Lanczos lower bidiagonalization process, however, it has to be

slightly modified. Let B_k be the lower bidiagonal matrix obtained by the Lanczos bidiagonalization process. Let $B_k = Q\tilde{B}_k$ be the QR decomposition of B_k . Then the Gauss and Gauss-Radau rules are obtained by [21]:

$$I_k(f) = \|\mathbf{w}\|^2 \mathbf{e}_1^T f(\tilde{B}_k \tilde{B}_k^T) \mathbf{e}_1, \quad (3.26)$$

$$\tilde{I}_k(f) = \|\mathbf{w}\|^2 \mathbf{e}_1^T f(\overline{\tilde{B}}_{k-1} \overline{\tilde{B}}_{k-1}^T) \mathbf{e}_1. \quad (3.27)$$

The QR decomposition can be carried out in $\mathcal{O}(k)$ steps. Alternatively, the matrix $T_k = \tilde{B}_k \tilde{B}_k^T$ for the Gauss and Gauss-Radau rules for functions of the form $\mathbf{w}^T G(G^T G + \lambda I)^{-p} G^T \mathbf{w}$ may be obtained by the Lanczos upper bidiagonalization algorithm [44].

Randomized SVD

In this section we present the algorithms that are used to compute the RSVD. Moreover, we provide error bounds for the parameter selection methods. Although

Algorithm 1 Randomized Range Finder (Algorithm 4.2 from [48])

Require: General matrix $G \in \mathbb{R}^{m \times n}$, tolerance ϵ and an integer r .

Ensure: Matrix Q_k s.t. $\|G - Q_k Q_k^T G\| < \epsilon$ holds with probability at least $1 - \min\{m, n\}10^{-r}$.

- 1: Draw a standard normally distributed matrix $\Omega \in \mathbb{R}^{n \times r}$.
 - 2: Compute $Y = G\Omega$.
 - 3: Set $j = 0$. Q_0 is empty.
 - 4: **while** $\max\{\|\mathbf{y}^{(j+1)}\|, \dots, \|\mathbf{y}^{(j+r)}\|\} > \delta/(10\sqrt{1/2\pi})$ **do**
 - 5: $j = j + 1$.
 - 6: $\mathbf{y}^{(j)} = \mathbf{y}^{(j)} - Q_{j-1} Q_{j-1}^T \mathbf{y}^{(j)}$.
 - 7: $\mathbf{q}^{(j)} = \mathbf{y}^{(j)} / \|\mathbf{y}^{(j)}\|$.
 - 8: $Q_j = [Q_{j-1} \ \mathbf{q}^{(j)}]$.
 - 9: Draw a random vector $\boldsymbol{\omega}^{(j+r)}$.
 - 10: $\mathbf{y}^{(j+r)} = (I - Q_j Q_j^T) G \boldsymbol{\omega}^{(j+r)}$.
 - 11: Orthogonalize $\mathbf{y}^{(j+1)}, \dots, \mathbf{y}^{(j+r-1)}$ against $\mathbf{q}^{(j)}$.
 - 12: **end while**
-

the RSVD algorithm has been used before for the purpose of solving discrete ill-posed problems, see e.g. [128, 129, 119], the algorithms presented there are fixed rank algorithms in the sense that they return an RSVD given an a-priori target rank. Here, we use an two-step algorithm from [48] which similar to the Lanczos algorithm is iterative in nature. The first step is to extract a good approximation to the range of G , which is done iteratively. The second step is to extract the RSVD. The first step of the algorithm, called the Adaptive Randomized Range Finder [48, algorithm 4.2], is presented in algorithm 1. We show the RSVD algorithm, taken from [48], in algorithm 2. We now present the error bounds for the parameter selection methods for the RSVD.

Algorithm 2 RSVD algorithm (Algorithm 5.1 from [48])

Require: General matrix $G \in \mathbb{R}^{m \times n}$, tolerance ϵ and an integer r .

Ensure: $G \approx U_k \Sigma_k V_k^T$, U and V are orthonormal and Σ_k diagonal.

- 1: Compute Q_k using algorithm (1).
- 2: Compute $B = Q_k^T G$.
- 3: Compute the SVD of B : $B = \tilde{U}_k \Sigma_k V_k^T$.
- 4: Compute $U = Q_k \tilde{U}$.

Theorem 3. (Adapted from [48, Corollary 10.9]) Let $W(A) = \mathbf{w}^T f_\lambda(A) \mathbf{w}$ and $T(A) = \text{trace}(f_\lambda(A))$. Let $A = \sum_{i=1}^n \sigma_i^2 \mathbf{u}_i \mathbf{u}_i^T$ where $(\sigma_i^2, \mathbf{u}_i)$ denotes an eigenpair of A . Let $\tilde{A}_k = \tilde{U}_k \tilde{\Sigma}_k \tilde{V}_k^T$ with $\tilde{\Sigma}_k = \text{diag}(\tilde{\sigma}_1, \dots, \tilde{\sigma}_k)$ be the RSVD of A given by algorithm (2). Then the relative errors are bounded by

$$\frac{|W(A) - W(A_k)|}{W(A)} \leq \frac{p}{\lambda^{p+1}} \frac{\left(1 + 8\sqrt{(k+p)p \log p}\right) \sigma_{k+1} + 3\sqrt{k+p} \left(\sum_{j>k} \sigma_j^2\right)^{1/2}}{f_\lambda(\sigma_m)} \quad (3.28)$$

with failure probability at most $6p^{-p}$,

$$\frac{|T(A) - T(A_k)|}{T(A)} \leq \frac{p}{\lambda^{p+1}} \frac{\left(\sum_{i=1}^k (\sigma_i^2 - \tilde{\sigma}_i^2) + \sum_{i=k+1}^m \sigma_i^2\right)}{\sum_{i=1}^m f_\lambda(\sigma_i)}. \quad (3.29)$$

Proof. The proof is similar to the proof of (2) except that the errors between A and A_k are now determined by the RSVD algorithm. The error bound is directly taken from [48, Corollary 10.9]. \square

Note that the RSVD in this theorem is the RSVD of A , which is either GG^T or $G^T G$. In practice we use the RSVD of G .

Randomized trace estimation

In this section we discuss estimating the trace. The trace of a symmetric positive definite matrix A can be estimated by a randomization approach, using Hutchinson's trace estimator [68]. Equivalently, we can use it to estimate the trace of the function of a matrix:

$$T(A) \approx T_N(A) := \frac{1}{N} \sum_{i=1}^N \mathbf{v}_i^T f_\lambda(A) \mathbf{v}_i, \quad (3.30)$$

This estimator is an unbiased estimator for the trace [43, theorem 1]. The entries of the vector \mathbf{v}_i are chosen according to a uniform distribution on the interval $[0, 1]$. Let t_i denote the i^{th} number drawn from this distribution, then the entries of \mathbf{v} are given by

$$v_i = \begin{cases} +1 & \text{if } t_i \geq 1/2 \\ -1 & \text{if } t_i < 1/2 \end{cases} \quad (3.31)$$

The vectors \mathbf{v} drawn from the distribution (3.31) are referred to as Rademacher vectors. For a matrix function, we can estimate its trace by the quantity $\mathbf{v}^T f_\lambda(A) \mathbf{v}$. To increase the accuracy of the estimator, the trace can be estimated by $V_N^T f_\lambda(A) V_N$, where V_N is a matrix with N columns and each column is of the form (3.31). We can not bound the trace exactly, but there exist probabilistic bounds for the trace estimator. In [110] a probabilistic bound for combined randomized trace estimation and model order reduction through Krylov subspaces is presented. The authors present an a priori bound for the combined randomized trace estimator and Gauss quadrature. In our case, we have to rely on an a priori bound for the randomized trace estimator. However, the accuracy of the Gauss quadrature to the trace estimator is estimated a posteriori based on how close the lower and upper bound are. We state a theorem similar to theorem 4.1 in [110]. Our approximation is either the upper or the lower bound denoted by \tilde{I}_k and I_k . The key to obtaining bounds is to split the error into two parts:

$$|T(A) - T_N(A_k)| \leq |T(A) - T_N(A)| + |T_N(A) - T_N(A_k)|.$$

The first term concerns the accuracy of the trace estimator itself. The second term is approximated using the lower and upper bounds by using the Lanczos process. For the first term there exist standard probabilistic bounds, [110], [68]. The second term is bounded in [110] for general functions using a different error bound for the Gauss quadrature. Here, we present an a posteriori bound based on the lower and upper bounds. We have

$$|T_N(A) - T_N(A_k)| \leq \left| \tilde{I}_k - I_k \right|.$$

To obtain a useful bound we now require

$$\left| \tilde{I}_k - I_k \right| \leq \frac{\epsilon}{2} T(A).$$

Using the fact that $f_\lambda(x) = (x + \lambda)^{-p}$, $p > 0$, we require

$$\left| \tilde{I}_k - I_k \right| \leq d \frac{\epsilon}{2} f_\lambda(\sigma_1) \leq \frac{\epsilon}{2} T(A).$$

This leads to the following theorem.

Theorem 4 (Adapted from [110]). *Choose $N \geq (24/\epsilon^2) \log(2/\eta)$ as the number of starting Rademacher vectors. Carry out k iterations of the Lanczos process such that*

$$\left| \tilde{I}_k - I_k \right| \leq d \frac{\epsilon}{2} f_\lambda(\sigma_1).$$

Then the output $T_N(A_k)$ is such that:

$$\Pr \left[|T(A) - T_N(A_k)| \leq \epsilon |T(A)| \right] \geq 1 - \eta. \quad (3.32)$$

Of course, σ_1 is not available. However, we can estimate σ_1 using the first Ritz value θ_1 . Unfortunately though, there is no estimate available for the quantity $|\sigma_1 - \theta_1|$ that does not depend on the singular values of A . By standard convergence theory for the Lanczos process, we do know that the largest Ritz value converges to the largest singular value first. Therefore, we expect the approximation to be quite accurate and use θ_1 instead of σ_1 to calculate the error bound. It should be noted, however, that a theorem of this form is not particularly useful for this application. In the limit, i.e. $\epsilon, \eta \rightarrow 1$, we have $N \gtrsim 16$ already, which can be prohibitively expensive. Preferably, we would like to use very few random vectors. We will investigate the impact of increasing N for small N (roughly 1 - 10) in our numerical experiments. It has been reported before in [7] that $N = 1$ has the optimal trade-off between computational complexity and accuracy. In our numerical experiments we investigate the influence of increasing N for small N .

Obtaining the solution

It is important be able to evaluate the parameter selection methods quickly in order to obtain a suitable λ . However, we are ultimately interested in the solution to the problem and the question arises whether we can obtain the solution to the problem quickly using the reduced order model for the parameter selection method. For the RSVD this is trivial: we simply use the RSVD we have also used to evaluate the parameter selection method. For the Lanczos process we do the same thing. [73, theorem 3.1] shows that the solution obtained from Lanczos process for the norm of the solution is the same as the solution from Conjugate Gradient applied to $(G^T G + \lambda I)\mathbf{m} = G^T \mathbf{d}$. This can be obtained easily from the B_k obtained from evaluating the norm of the residual. Hence, for every parameter selection method we can easily obtain a solution to the problem with an estimate for λ .

3.2.4 Computational costs

We compare the computational costs for Lanczos bidiagonalization to the presented RSVD algorithm in terms of FLOPs. We start with Lanczos bidiagonalization. The costs for the standard Lanczos bidiagonalization algorithm for a matrix $G \in \mathbb{R}^{m \times n}$ are

$$k \cdot \text{nnz}(A)(m + n) + 5(m + n), \quad (3.33)$$

where the first term is for the matrix-vector multiplication and the second term is for various subtractions, divisions and taking the norm of vectors. It should be noted that the Lanczos bidiagonalization algorithm is known to be unstable, i.e. the orthogonal bases lose orthogonality, and may require reorthogonalization [42]. It has been shown that a loss of orthogonalization has a strong influence on the estimated eigenvalues, but the effect on the solution of a linear system is small ([55], page 158). The costs for algorithm (1) are reported in [48], section 6.2, and are

$$kmR + k \cdot \text{nnz}(A)n + k^2 m, \quad (3.34)$$

where the first term is the cost for generating Ω , the second term is the cost of matrix-vector multiplication and the third term is the cost for the orthogonalization of Q , in this case done by the Gram-Schmidt algorithm. If, for numerical stability, we

use Householder reflectors, this cost would increase to roughly $2k^2m - \frac{2}{3}k^3$ ([42], section 5.2.2). The costs for extracting the SVD are $\mathcal{O}(mk^2)$ with the addition of $2mnk$ FLOPs for the multiplications with Q . The big advantage of the RSVD is the possibility to easily parallelize the computation and the fact that only one pass over the data is needed. The ability to parallelize makes that, although the number of matrix-vector multiplications may be similar, the RSVD algorithm is faster in terms of computational time. When access to the matrix A is prohibitively expensive the RSVD is certainly the desired option. For an in depth discussion on this topic see [48], section 6.2.

3.3 Algorithms

In this section we show a blueprint for an algorithm based on either Lanczos quadrature or the RSVD for selecting the regularization parameter. We wish to make some small clarifying notes. We use the sampling for λ in order to be able to detect if there is a minimizer and to easily check how close the upper and lower bounds are. If we find a minimizer where the upper and lower bounds are not close enough, we resample around the minimizer. We always check whether the minimizer is not at the boundary of the sampled λ . In this case we simply resample again. Minimizing only the upper bound halves the number of evaluations we have to do for the sampled λ . Evaluating for a given λ is cheap, as it involves solving systems involving B_k .

Algorithm 3 Quadrature bounds for Reginska's rule.

Require: The data \mathbf{d} , matrix G and tolerance ϵ and a range of $\lambda \in [\lambda_{\min}, \lambda_{\max}]$.

Ensure: $\lambda_{\text{Reginska}}^U$ and $\lambda_{\text{Reginska}}^L$ with relative error ϵ .

- 1: **while** Not converged **do**
- 2: Carry out a step of Lanczos bidiagonalization yielding $B_{k+1,k}$ and $B_{k,k}$.
- 3: Compute the matrices $\tilde{B}_{k,k}$ and $\tilde{B}_{k,k-1}$.
- 4: Compute the upper bound for the norm of the solution and the norm of the residual:

$$\text{ub_s}(\lambda) = \|\mathbf{b}\|^2 \mathbf{e}_1^T \left(\tilde{B}_{k,k-1} \tilde{B}_{k,k-1}^T + \lambda I \right)^{-2} \mathbf{e}_1$$

$$\text{ub_r}(\lambda) = \lambda^2 \|\mathbf{b}\|^2 \mathbf{e}_1^T \left(B_{k,k} B_{k,k}^T + \lambda I \right)^{-2} \mathbf{e}_1$$

- 5: **if** upper bound yields a minimizer **then**
 - 6: Calculate lower bound at the minimizer.
 - 7: **if** relative error is smaller than ϵ **then**
 - 8: Compute $\lambda_{\text{Reginska}}^U := \min_{\lambda} \text{ub}_{\text{Reginska}}^{(k)}(\lambda)$ and $\lambda_{\text{Reginska}}^L := \min_{\lambda} \text{lb}_{\text{Reginska}}^{(k)}(\lambda)$.
 - 9: **if** $|\text{ub}_{\text{Reginska}}(\lambda_{\text{Reginska}}^U) - \text{lb}_{\text{Reginska}}(\lambda_{\text{Reginska}}^U)| < \epsilon$ and $|\lambda_{\text{Reginska}}^U - \lambda_{\text{Reginska}}^L| < \epsilon$ **then**
 - 10: **break**
 - 11: **end if**
 - 12: **else**
 - 13: Resample λ around the minimizer.
 - 14: **end if**
 - 15: **end if**
 - 16: $k \rightarrow k + 1$.
 - 17: **end while**
-

Algorithm 4 Quadrature bounds for GCV with randomized trace estimator.

Require: Data \mathbf{d} and $U \in \mathbb{R}^{n \times N}$, matrix G and tolerance ϵ and a range of $\lambda \in [\lambda_{\min}, \lambda_{\max}]$.

Ensure: λ_{GCV}^U and λ_{GCV}^L with relative error ϵ .

- 1: **while** Not converged **do**
- 2: Carry out a step of Lanczos bidiagonalization for starting vectors $\mathbf{b}, \mathbf{u}_1, \dots, \mathbf{u}_k$ yielding $B_{k+1,k}$ and $B_{k,k}$ for every starting vector.
- 3: Sample λ . We choose 10 values on a log scale between λ_{\min} and λ_{\max} .
- 4: Compute the lower and upper bound for the norm of the residual and the trace estimators:

$$\begin{aligned} \text{ub}_r(\lambda) &= \lambda^2 \|\mathbf{b}\|^2 \mathbf{e}_1^T (B_{k,k} B_{k,k}^T + \lambda I)^{-2} \mathbf{e}_1 \\ \text{ub}_{\mathbf{u}_i}(\lambda) &= \|\mathbf{u}_i\|^2 \mathbf{e}_1^T (\tilde{B}_{k,k} \tilde{B}_{k,k}^T + \lambda I)^{-1} \mathbf{e}_1 \end{aligned}$$

- 5: **if** upper bound yields a minimizer **then**
 - 6: Calculate lower bound at the minimizer.
 - 7: **if** relative error is smaller than ϵ **then**
 - 8: Compute $\lambda_{\text{GCV}}^U := \min_{\lambda} \text{ub}_{\text{GCV}}^{(k)}(\lambda)$ and $\lambda_{\text{GCV}}^L := \min_{\lambda} \text{lb}_{\text{GCV}}^{(k)}(\lambda)$.
 - 9: **if** $|\text{ub}_{\text{GCV}}(\lambda_{\text{GCV}}^U) - \text{lb}_{\text{GCV}}(\lambda_{\text{GCV}}^U)| < \epsilon$ and $|\lambda_{\text{GCV}}^U - \lambda_{\text{GCV}}^L| < \epsilon$ **then**
 - 10: break
 - 11: **end if**
 - 12: **else**
 - 13: Resample λ around the minimizer.
 - 14: **end if**
 - 15: **end if**
 - 16: $k \rightarrow k + 1$.
 - 17: **end while**
-

Algorithm 5 Quadrature bounds for GCV with Ritz value based trace estimator.

Require: Data \mathbf{d} and $U \in \mathbb{R}^{n \times N}$, matrix G and tolerance ϵ and a range of $\lambda \in [\lambda_{\min}, \lambda_{\max}]$.

Ensure: λ_{GCV}^U and λ_{GCV}^L with relative error ϵ .

- 1: **while** Not converged **do**
 - 2: Carry out a step of Lanczos bidiagonalization for starting vector \mathbf{d} yielding $B_{k+1,k}$ and $B_{k,k}$.
 - 3: Sample λ . We choose 10 values on a log scale between λ_{\min} and λ_{\max} .
 - 4: Compute the upper bound for the norm of the residual: $\text{ub}_r(\lambda) = \lambda^2 \|\mathbf{b}\|^2 \mathbf{e}_1^T (B_{k,k} B_{k,k}^T + \lambda I)^{-2} \mathbf{e}_1$
 - 5: Estimate the trace using the Ritz values $B_k B_k^T = U \Theta V^T$, $\Theta = \text{diag}(\theta_1, \dots, \theta_k)$. $\text{trace}(I - A(A^T A + \lambda I)^{-1} A^T) \approx \lambda \sum_{i=1}^k \frac{1}{\theta_i + \lambda} + n - k := T(A_k)$
 - 6: **if** upper bound yields a minimizer **then**
 - 7: Calculate error with upper bound from previous iteration.
 - 8: **if** relative error is smaller than ϵ **then**
 - 9: Compute $\lambda_{\text{GCV}}^U := \min_{\lambda} \text{ub}_{\text{GCV}}^{(k)}(\lambda)$ and $\lambda_{\text{GCV}}^L := \min_{\lambda} \text{lb}_{\text{GCV}}^{(k)}(\lambda)$.
 - 10: **if** $|\text{ub}_{\text{GCV}}^{(k)}(\lambda_{\text{GCV}}^U) - \text{ub}_{\text{GCV}}^{(k-1)}(\lambda_{\text{GCV}}^U)| < \epsilon$ and $|\lambda_{\text{GCV}}^U - \lambda_{\text{GCV}}^L| < \epsilon$ **then**
 - 11: break
 - 12: **end if**
 - 13: **else**
 - 14: Resample λ around the minimizer.
 - 15: **end if**
 - 16: **end if**
 - 17: $k \rightarrow k + 1$.
 - 18: **end while**
-

Algorithm 6 RSVD for any parameter selection method.

Require: Data \mathbf{d} , matrix G , a tolerance ϵ and an integer r .

Ensure: $\hat{\lambda}$ such that the inequality (3.28) holds.

- 1: Compute Q_k using the Adaptive Randomized Range Finder algorithm.
 - 2: Compute the RSVD of $Q_k^T A = U_k \Sigma_k V_k^T$. Obtain $A \approx \tilde{A}_k = Q_k U_k \Sigma_k V_k^T$.
 - 3: Use \tilde{A}_k to evaluate the parameter selection methods.
-

3.4 Numerical experiments

3.4.1 Gravity surveying

We consider the classical example of gravity surveying, see e.g. [57]. Let $m(t)$ be the mass at location t and $d(s)$ be the measured force at the surface at location s . Let h denote the depth of the gravity field. We then have the following relation

$$d(s) = \int_0^1 \frac{h}{(h^2 + (s - t)^2)^{3/2}} m(t) dt.$$

The problem is to retrieve $m(t)$ from measurements $d(s)$. We show the setup in figure (3.1). Because this is a Fredholm integral operator of the first kind, the

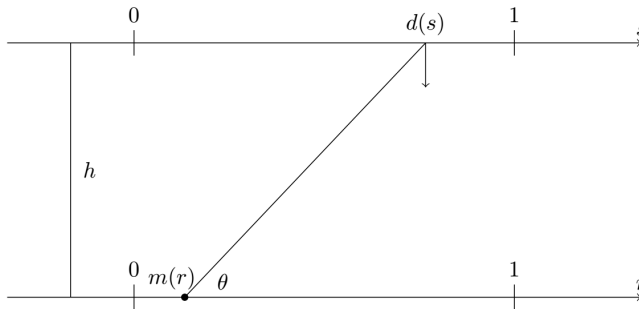


Figure 3.1: Setup for the gravity problem. Figure is taken from [57].

problem of retrieving $m(t)$ is ill-posed. Specifically, the gravity surveying problem is severely ill-posed due to the severe decay of the singular values, as can be observed from figure (3.2). We regularize the problem using standard form Tikhonov regularization. We show the approximation error and dimension of the lower

Table 3.2: Comparison of quadrature bounds versus the RSVD for $\epsilon = 10^{-1}$. $m = n = 1000$. Results are averages plus-minus one standard deviation over 10 different noise realizations.

Method	Lanczos		RSVD	
	$\frac{\ \lambda - \hat{\lambda}\ }{\ \lambda\ }$	k	$\frac{\ \lambda - \hat{\lambda}\ }{\ \lambda\ }$	k
GCV	$1.3 \cdot 10^{-2} \pm 1.2 \cdot 10^{-2}$	11.9 ± 1	$1.1 \cdot 10^{-2} \pm 1.2 \cdot 10^{-2}$	15
Reginska	$2.4 \cdot 10^{-3} \pm 2.5 \cdot 10^{-3}$	9.5 ± 0.7	$4.0 \cdot 10^{-4} \pm 9.0 \cdot 10^{-5}$	15
QO	$8.3 \cdot 10^{-3} \pm 8.8 \cdot 10^{-3}$	9 ± 0	$8.5 \cdot 10^{-3} \pm 5.1 \cdot 10^{-3}$	15
DP	$6.1 \cdot 10^{-4} \pm 1.6 \cdot 10^{-3}$	7.5 ± 1	$1.4 \cdot 10^{-2} \pm 1.3 \cdot 10^{-2}$	15

dimensional space for $\epsilon = 10^{-1}, 10^{-2}$ and 10^{-3} in tables (3.2), (3.3) and (3.4) respectively. Because we use a probabilistic measure to check for convergence the

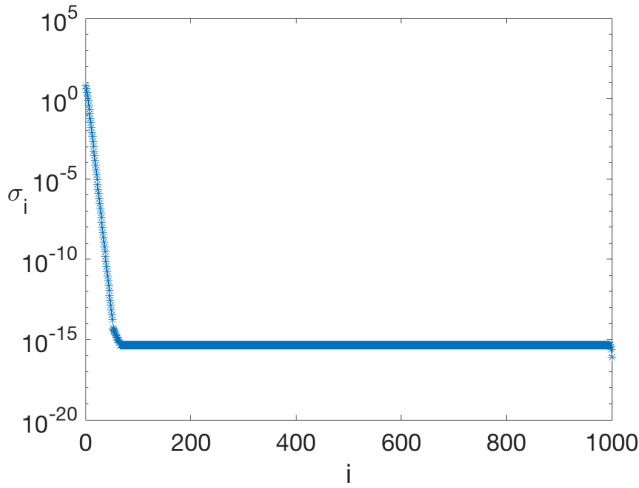


Figure 3.2: Singular values of the matrix G for the gravity problem where $m = n = 1000$.

Table 3.3: Comparison of quadrature bounds versus the RSVD for $\epsilon = 10^{-2}$. $m = n = 1000$. Results are averages plus-minus one standard deviation over 10 different noise realizations.

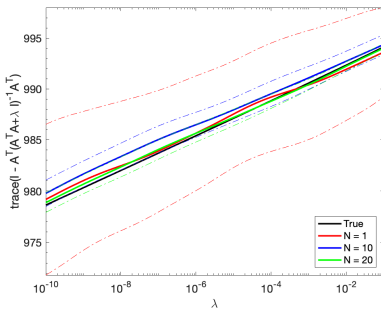
Method	Lanczos		RSVD	
	$\frac{\ \lambda - \hat{\lambda}\ }{\ \lambda\ }$	k	$\frac{\ \lambda - \hat{\lambda}\ }{\ \lambda\ }$	k
GCV	$1.2 \cdot 10^{-3} \pm 1.1 \cdot 10^{-3}$	14.2 ± 1.4	$1.3 \cdot 10^{-3} \pm 1.8 \cdot 10^{-3}$	18
Reginska	$9.5 \cdot 10^{-4} \pm 1.4 \cdot 10^{-3}$	9.5 ± 0.5	$1.6 \cdot 10^{-5} \pm 1.1 \cdot 10^{-5}$	18
QO	$1.1 \cdot 10^{-3} \pm 2.2 \cdot 10^{-3}$	9.6 ± 0.5	$3.2 \cdot 10^{-5} \pm 1.7 \cdot 10^{-5}$	18
DP	$7.5 \cdot 10^{-6} \pm 1.4 \cdot 10^{-5}$	8.4 ± 1	$1.6 \cdot 10^{-4} \pm 4.7 \cdot 10^{-5}$	18

size of Q_k may vary with different realizations.

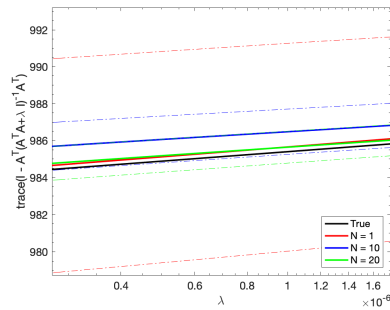
In figure (3.3) we show the performance of the randomized trace estimator for varying N , where we have averaged over 10 realizations of the random vectors \mathbf{u} . We show the average of the 10 realizations and the dotted lines indicate one standard deviation. It is clear that with increasing N we obtain a better approximation on average. Moreover, the standard deviation drastically decreases. However, twenty random vectors is generally too computationally expensive and with regard to theorem (4), does not give us any strong guarantees on how close it will be to the true trace. In figure (3.4) we show the trace estimator using the Ritz values. The accuracy of the trace estimator using the Ritz values rapidly increases as k increases. For large values of λ the trace is well approximated early, but for small λ we need more iterations. It is important to note that we are interested in approximating the trace well around the optimal λ , which is unlikely to be very small. For $k = 30$ we have already obtained a near perfect approximation of the

Table 3.4: Comparison of quadrature bounds versus the RSVD for $\epsilon = 10^{-3}$. $m = n = 1000$. Results are averages plus-minus one standard deviation over 10 different noise realizations.

Method	Lanczos		RSVD	
	$\frac{\ \lambda - \hat{\lambda}\ }{\ \lambda\ }$	k	$\frac{\ \lambda - \hat{\lambda}\ }{\ \lambda\ }$	k
GCV	$2.5 \cdot 10^{-4} \pm 2.1 \cdot 10^{-4}$	15.5 ± 1.7	$7.8 \cdot 10^{-4} \pm 2.4 \cdot 10^{-3}$	21
Reginska	$2.4 \cdot 10^{-5} \pm 2.5 \cdot 10^{-5}$	10 ± 0	$6.4 \cdot 10^{-7} \pm 7.9 \cdot 10^{-7}$	21
QO	$9.0 \cdot 10^{-5} \pm 1.2 \cdot 10^{-4}$	9.9 ± 0.7	$3.1 \cdot 10^{-5} \pm 1.9 \cdot 10^{-5}$	21
DP	$2.5 \cdot 10^{-8} \pm 1.8 \cdot 10^{-7}$	9.9 ± 0.6	$4.3 \cdot 10^{-5} \pm 5.3 \cdot 10^{-5}$	21



(a) Full view.



(b) Zoom.

Figure 3.3: Randomized trace estimator for increasing N for the gravity problem. We show the average for 10 random realizations and the dashed-dotted lines are plus minus one standard deviation.

trace. The extra work needed in calculating the Ritz values is small, as we are computing the SVD of a $k \times k$ symmetric tridiagonal matrix. For $k = 14$ we have already obtained a satisfactory approximation of the trace, as the trace around the optimal λ is well approximated. This is due to the fact that the spectrum decays very quickly, as can be seen from figure (3.2).

3.4.2 Cross-well tomography

We consider classical linear cross-well tomography, an example taken from the AIR tools package [59]. We show the setup of the problem in figure (3.5). On the right are the sources and on the left are the receivers. We show the rays travelling from one source to all receivers. The data are the traveltimes from source i to receiver j and the goal is to reconstruct the well. We show the data and the well in figure (3.6). Typically, for this setup we have far less sources and receivers than gridpoints. This means that the problem is underdetermined. The well has a smooth structure, and since the problem is underdetermined, we use general form Tikhonov regularization where L is the discrete Laplace operator: this enforces a

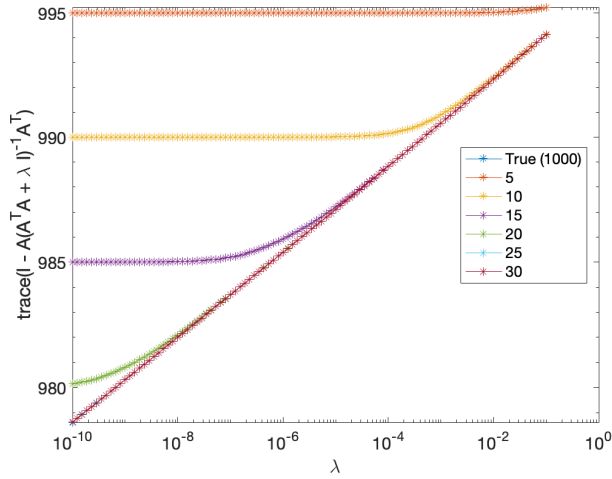


Figure 3.4: Approximation of the trace using the Ritz values for increasing k .

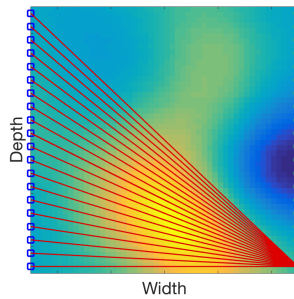
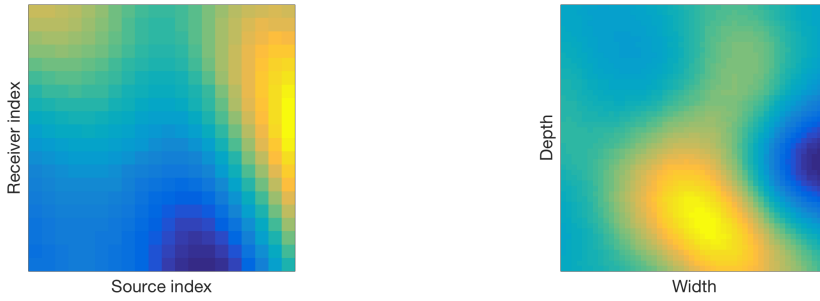


Figure 3.5: Linear cross-well tomography.

smooth reconstruction. The problem is mildly ill-posed due to the fact that the singular values decay mildly, as can be observed from figure (3.8). For the noise level we use $\delta = 10^{-1}$. For the Adaptive Randomized Range Finder we use a modified scheme based on the RSVD for underdetermined problems from [129]. Instead of using $A\Omega$ we use ΩA , or equivalently, $A^T \Omega$, to obtain the RSVD. For the Adaptive Randomized Range Finder we use the same parameters as for the gravity problem. Interestingly, the Adaptive Randomized Range Finder does not converge until we have obtained the full QR decomposition. We show the true errors $\|G - Q_k Q_k^T G\|_F$ for all k in figure (3.7). The performance of the Lanczos method is shown in table 3.5. The Quasi-Optimality Criterion did not yield a minimizer, hence we have omitted this rule from the results. Notice that although the trace estimator seems rather accurate, there is still a considerable error compared to the optimal λ . This does not mean that the solution will necessarily be bad though.

We compare the randomized trace estimator versus the approximation based on



(a) Traveltimes: entry (i,j) indicates the traveltime from source i to receiver j .

(b) The ground truth.

Figure 3.6: Traveltimes and the well for the tomography problem.

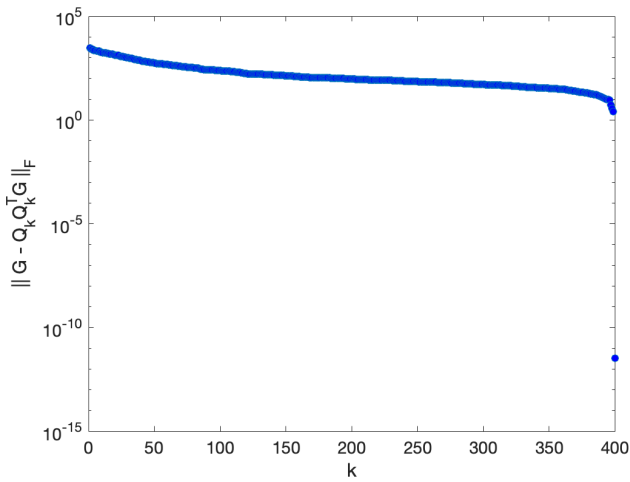


Figure 3.7: $\|G - Q_k Q_k^T G\|_F$ for all k .

the Ritz values in figure (3.9). We show the accuracy for varying N in figure (3.10).

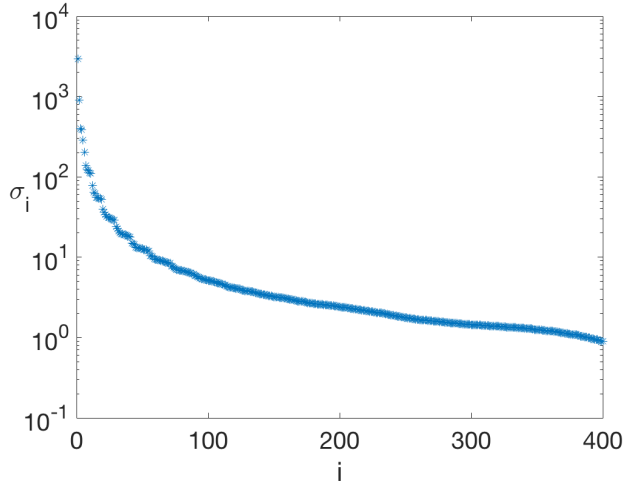
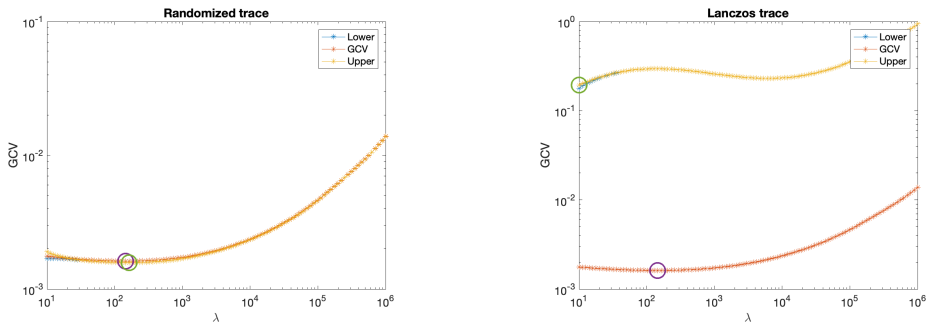


Figure 3.8: Singular values of the matrix GL^{-1} for the tomography problem.

Table 3.5: Results for parameter selection using the Lanczos procedure. $m = 400$ and $n = 2500$. Results are averages plus-minus one standard deviation over 10 different noise realizations.

Method	Lanczos	
	$\frac{\ \lambda - \hat{\lambda}\ }{\ \lambda\ }$	k
GCV	$2.4 \cdot 10^{-1} \pm 1.7 \cdot 10^{-1}$	15 ± 2.5
Reginska	$3.1 \cdot 10^{-3} \pm 6.5 \cdot 10^{-3}$	9.4 ± 0.8
DP	$1.4 \cdot 10^{-3} \pm 7.6 \cdot 10^{-4}$	20.7 ± 4



(a) GCV approximation using the trace estimator.

(b) GCV approximation using the Ritz values.

Figure 3.9: Comparison of trace estimators for the GCV for $k = 30$. The circles denote the minimizers.

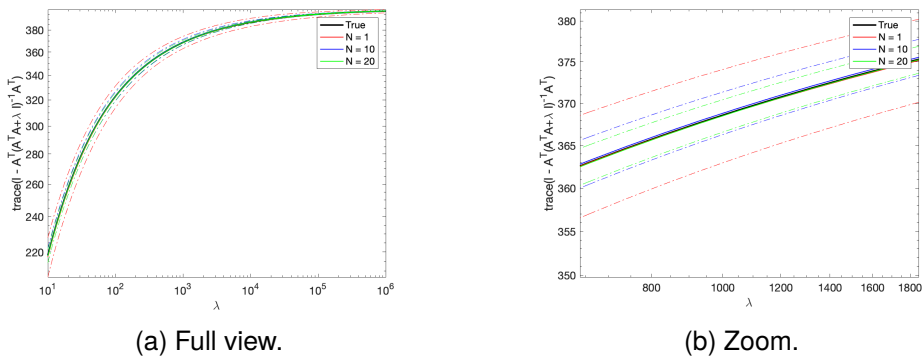


Figure 3.10: Randomized trace estimator for increasing N for the tomography problem. We show the average for 10 random realizations and the dotted line are plus minus one standard deviation.

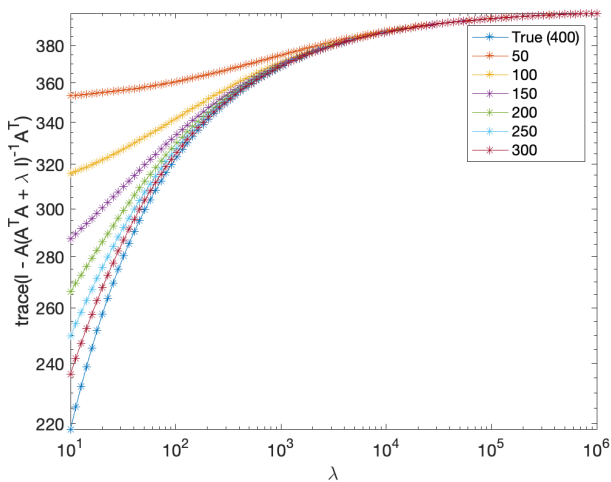


Figure 3.11: Approximation of the trace using the Ritz values for increasing k .

3.5 Conclusion

In this chapter we have compared the use of the Lanczos process for parameter selection methods versus the use of the RSVD. We have derived bounds for the parameter selection methods when estimated using the RSVD. We have presented a theorem that provides probabilistic bounds for trace estimation combined with a low dimensional approximation obtained by the Lanczos method. This theorem provides us with certain guarantees in terms of accuracy. However, these guarantees require too many computations. We have compared the use of Lanczos quadrature and RSVD for two model problems from geosciences: gravity surveying and linearized cross-well tomography. We have also compared the use of Hutchinson's trace estimator versus the trace estimator based on the estimates for the singular values from the Lanczos process and the RSVD. The gravity surveying problem is severely ill-posed and we have shown that, for this problem, the Lanczos quadrature method and the RSVD yield comparable results. We have also shown that the trace estimator based on the Ritz values of the Lanczos process (or the estimated singular values of the RSVD) outperforms the randomized trace estimator for the GCV. For the tomography problem, which is a mildly ill-posed underdetermined problem, we have shown that the RSVD failed to provide a satisfactory low dimensional approximation to evaluate the parameter selection methods. The Lanczos quadrature method was able to provide a lower dimensional approximation. The key difference is that, due to the fact that we obtain lower and upper bounds for the parameter selection methods, we obtain a lower dimensional model given the λ estimated by the parameter selection method. For the tomography problem this is a great advantage, because the optimal λ is quite large. A large λ allows for a lower dimensional approximation than a small λ , something which is reflected by the error bounds for the Lanczos quadrature method, and the bounds derived by us for the RSVD. For the tomography problem, we have shown that Hutchinson's trace estimator gives a far better approximation of the trace for small k than using the estimates obtained by the Lanczos procedure.

3.6 Appendix

Relations

We give a short derivation of (3.9). We have

$$\|G\hat{\mathbf{m}}_\lambda - \mathbf{d}\| = \|(G(G^T G + \lambda I)^{-1} G^T - I) \mathbf{d}\|. \quad (3.35)$$

We now use the following relation:

$$(G^T G + \lambda I)^{-1} (G^T G + \lambda I) G^T = G^T \quad (3.36)$$

$$\iff (G^T G + \lambda I)^{-1} G^T (G G^T + \lambda I) = G^T \quad (3.37)$$

$$\iff (G^T G + \lambda I)^{-1} G^T = G^T (G G^T + \lambda I)^{-1} \quad (3.38)$$

Plugging this into (3.35) gives

$$\|G\hat{\mathbf{m}}_\lambda - \mathbf{d}\| = \|(G G^T (G G^T + \lambda I)^{-1} - I) \mathbf{d}\|. \quad (3.39)$$

Using the relation

$$(G G^T + \lambda I)(G G^T + \lambda I)^{-1} = I \quad (3.40)$$

$$\iff G G^T (G G^T + \lambda I)^{-1} = I - \lambda (G G^T + \lambda I)^{-1} \quad (3.41)$$

Plugging this into (3.39) yields the desired result

$$\|G\hat{\mathbf{m}}_\lambda - \mathbf{d}\|^2 = \lambda^2 \mathbf{d}^T (G G^T + \lambda I)^{-2} \mathbf{d}. \quad (3.42)$$

Measure

In this section we describe the relation (3.15). Our aim is to give an explanation of how the piece-wise measure works for the reader that has no experience with measure theory, without any mathematical rigor, but simply to give an intuitive idea. To understand the measure in (3.15), it suffices to think of a measure as a weighted integral. Consider the following integral where $g(x)$ is a continuously differentiable function:

$$\int_{\Omega} f(x) \mathrm{d}g(x) = \int_{\Omega} f(x) \frac{\mathrm{d}g(x)}{\mathrm{d}x} \mathrm{d}x = \int_{\Omega} f(x) g'(x) \mathrm{d}x. \quad (3.43)$$


Hence, if the measure is a continuously differentiable function we can regard the measure as a weighted integral, where $g'(x)$ is the weight. Now if the function $g(x)$ is piecewise constant it is no longer differentiable. Consider the following function:

$$f(x) = \begin{cases} 0 & \text{if } 0 \leq x \leq 1 \\ 1 & \text{if } 0 < x \leq 2 \end{cases} \quad (3.44)$$


The function is everywhere differentiable except at $x = 1$. Now, for $h > 0$

$$\frac{f(1+h) - f(1)}{h} = \frac{1}{h}. \quad (3.45)$$

If $h \rightarrow 0$ the function will go to infinity. We can regard the derivative of this function at $x = 1$ as a delta function. Hence, when integrating a function with a piecewise constant measure we can regard this as taking point evaluations. If we let the discontinuities be at the eigenvalues of a matrix, we get (3.15).



Block-Krylov methods for Multi-Dimensional Deconvolution



Abstract We address the estimation of the impulse response at a reflector by deconvolving the up- and downgoing waves. Deconvolution in the time domain can be written as a linear system with multiple right-hand sides in the frequency domain. A straightforward way of solving these systems is by applying an iterative method like LSQR. However, these solvers are dependent on the right-hand side and for every right-hand side we have to use LSQR again. This can be a costly process. We propose to solve the linear systems using block Krylov methods. We show that these methods give comparable accuracy compared to standard Krylov methods, but at a much lower computational cost. This is due to the fact that block methods are able to exploit similarities in the data and are able to construct solutions in a much richer subspace. We also show that it is hard to solve the MDD problem in the frequency domain alone, and that additional optimization in the time domain is most likely required.

This chapter is partially based on the following publication:

N.A. Luiken and A. Garg. Block-Krylov methods for multi-dimensional deconvolution. *Society of Exploration Geophysicists*, pages 5070–5074, 2019.

4.1 Introduction

In this work we address Multi-Dimensional Deconvolution (MDD), specifically in the context of seismic wavefield redatuming. The objective is to estimate an impulse response, or Green's function, of a reflector in the subsurface from the downgoing and upgoing wavefields at the reflector. The upgoing wavefield is a convolution of the downgoing wavefield with the impulse response in the time domain. Due to the large size of the matrices, the problem is often transformed to the frequency domain. In the frequency domain the convolution is simply a multiplication, which means that deconvolving amounts to solving systems of the form $XP^+ = P^-$ for all frequencies, where P^+ and P^- denote the downgoing and upgoing wavefields respectively for a number of experiments, and X is the unknown multi-dimensional impulse response. This is a linear system of equations with multiple right-hand sides. Usually, these systems are solved with a standard solver like LSQR, which seeks an approximation in a Krylov subspace. However, LSQR depends on the right-hand side and one has to solve for each right-hand side individually. There are solvers that are specifically designed to handle multiple right-hand sides and exploit redundancy in the data. These solvers are based on so-called block Krylov methods [40], that find a solution in a larger Krylov subspace than standard solvers do. In certain cases, this can lead to a considerable speed-up in computations. Through an example, we show how these solvers can be used to obtain a considerable speed-up in computations in the MDD process.

4.2 Block Krylov methods

In this section we shortly outline the basic theory behind block Krylov methods. For an in depth overview we refer the reader to [46]. We start with some notation. We are interested in solving linear systems of the form

$$GX = D, \quad (4.1)$$

where $G \in \mathbb{C}^{m \times n}$, $X \in \mathbb{C}^{n \times s}$, $D \in \mathbb{C}^{m \times s}$. G is the model, D is the data and X is the variable of interest. Equivalently, the system is solved by the following minimization problem:

$$\min_X \|GX - D\|_F. \quad (4.2)$$

In case $s = 1$ we have a system with one right-hand side. Solutions to these systems can be obtained through standard iterative methods such as LSQR, CG or GMRES. These solvers are based on Krylov subspaces, which are defined as

$$\mathcal{K}_k(G, \mathbf{d}) := \text{span} \{ \mathbf{d}, G\mathbf{d}, \dots, G^{k-1}\mathbf{d} \}.$$

The iterative methods build an approximate solution by projecting onto the Krylov subspace and the number of iterations corresponds to the size of the subspace. The methods terminate, in exact arithmetic, after n steps and find the exact solution (provided G is nonsingular). The aim is to obtain a good approximation to the true solution for $k \ll n$. Note that the Krylov subspaces are based on a starting vector \mathbf{d} . In the context of solving linear systems, this is usually the data. When $s > 1$ in (4.1), we have to solve multiple linear systems with the same model. This raises

the question whether there is a more efficient way to solve this system. Intuitively, it is clear that when two columns of D , \mathbf{d}_i and \mathbf{d}_j , are similar that they can be approximated in the same Krylov subspace. In the extreme case where they are orthogonal they have to be approximated in two completely different spaces. It is reasonable to expect that for low rank D we can re-use Krylov subspaces generated for nearly linearly dependent vectors. This idea is exploited by block methods. The block Krylov subspace is defined as

$$\begin{aligned}\mathcal{K}_k^\square(G, D) &:= \text{block span} \{D, GD, \dots, G^{k-1}D\} \\ &:= \sum_{i=0}^{k-1} G^i DC_i, \quad C_i \in \mathbb{C}^{s \times s}.\end{aligned}$$

It is important to note that the coefficients C_i are not real numbers, but matrices. It can be shown that we have the following equivalence

$$\begin{aligned}X = [\mathbf{x}_1 | \dots | \mathbf{x}_r] \in \mathcal{K}_k^\square(G, D) &\Leftrightarrow \mathbf{x}_l = \sum_{j=1}^r \sum_{i=0}^{k-1} G^i \mathbf{b}_j \beta_{i,j}^{(l)}, \\ l = 1, \dots, r &\quad \beta_{i,j}^{(l)} \in \mathbb{C}.\end{aligned}$$

If we solve (4.1) using a non-block method by projecting onto a Krylov subspace depending on \mathbf{d}_l we have

$$\mathbf{x}_l = \sum_{i=0}^{k-1} G^i \mathbf{d}_l \alpha_i^{(l)}.$$

Hence, we see that if we solve (4.1) using a block method, the columns \mathbf{x}_l of X are in a much richer subspace than if we solve with a standard method. We now show the block Arnoldi algorithm from [46], which is the basis for block-GMRES and state the conditions under which block methods are preferred. The algorithm is shown in 7.

Algorithm 7 Block Arnoldi process

Require: Matrix $G \in \mathbb{C}^{m \times m}$ and starting vector $B \in \mathbb{C}^{m \times s}$.

Ensure: Matrices $Q_k = [q_1, \dots, q_s] \in \mathbb{C}^{m \times ks}$, with $q_i^H q_j = 0$, $q_i^H q_i = I$, and upper block Hessenberg $H_k \in \mathbb{C}^{(k+1)s \times ks}$ with entries $h_{ij} \in \mathbb{C}^{s \times s}$.

- 1: $q_1 h_{00} = B$. (QR decomposition of B)
 - 2: **for** $i = 1$ to k **do**
 - 3: $r_i = Gq_{i-1}$
 - 4: **for** $j = 1$ to i **do**
 - 5: $h_{ji} = q_j^H r_i$
 - 6: $r_i = r_i - q_j h_{ji}$
 - 7: **end for**
 - 8: $q_i h_{i+1,i} = r_i$. (QR decomposition of r_i)
 - 9: **end for**
-

After k steps of the algorithm we have the following relations

$$GQ_k = Q_{k+1}H_k.$$

Making the substitution $X = Q_k Y$ and inserting this into (4.2) we obtain the reduced system

$$\min_X \|GX - D\|_F = \min_Y \|H_k Y - \tilde{D}\|_F.$$

If $s = 1$ then solving this reduced system is done by the GMRES algorithm. Note that the difference between the standard Arnoldi algorithm and the block Arnoldi algorithm is that instead of multiplying with a vector, we multiply with matrices. It is clear that the QR factorizations in algorithm 7 can become prohibitively expensive for large s . For solving linear systems of the form (4.1) s is equal to the size of D . Hence if the data matrix is large, block methods become unattractive. This drawback can be overcome through a process called *deflation*. There are two types of deflation: initial deflation and Arnoldi deflation. Initial deflation concerns the data. The linear dependencies in the data are removed which leaves a linearly independent set of vectors. This set of vectors is then used as the start matrix for the block method. It is clear that the larger the start matrix is, the more expensive the block method becomes. Therefore, it is essential that for large right-hand sides we are able to deflate the data and work with small blocks that yield the same amount of information. Initial deflation can be done via the QR factorization of the data. Let

$$D = [Q_1, Q_2] \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix}.$$

In exact arithmetic, the rank of D is the size of the block R_{11} and R_{22} will be 0. In applications we typically choose a certain tolerance for the rank of the matrix. R_{11} will then be the block corresponding to the tolerance and $\|R_{22}\|_F$ will be small. $Q_1 R_{11}$ will now be the QR decomposition of the starting matrix for the Arnoldi process. From this process we obtain an approximate solution X_1 . The full solution is now given by

$$X = [X_1, X_2] := [X_1, X_1 R_{11}^{-1} R_{12}].$$

For a detailed derivation we refer the reader to [46].

Arnoldi deflation occurs when one of the $h_{i,j}$ becomes rank deficient. We have not encountered this issue and therefore we do not elaborate on it. To use initial deflation we have to determine the rank of D . Determining the rank of a matrix is a costly procedure though, and can be as costly as factorizing the matrix G , which again can make block methods unattractive.

4.3 Analysis via SVD

Before we attempt to solve the problem we would first like to make an analysis of the linear systems we are solving. We will show that the nature of the problem changes as the frequency increases. We first start by making an important distinction between the rank of a matrix and the *numerical rank* of a matrix. The rank is defined as the number of nonzero singular values. However, in finite precision arithmetic, matrices can often have very small singular values that are not precisely 0, but are so small that the matrix can be regarded as rank deficient. If a matrix has singular values around machine precision these are often considered to be 0. The presence of small singular values tells us something about the ill-posedness of our problem. A problem is called ill-posed if it violates one of the following three conditions:

1. there exists a solution,
2. the solution is unique,
3. the solution is stable with respect to perturbations.

Small singular values lead to a violation of the third condition. The small singular values amplify small changes in the data in the inversion which makes the solution unstable. If the matrix is rank deficient we only obtain a solution if $\mathbf{b}_i \in \mathcal{R}(A)$ for $i = 1, \dots, r$. Otherwise, the second property is violated. Note that in finite precision arithmetic we use the numerical rank instead of the rank to determine rank deficiency. We show the rank of G for all frequencies in figure 4.1. We see that

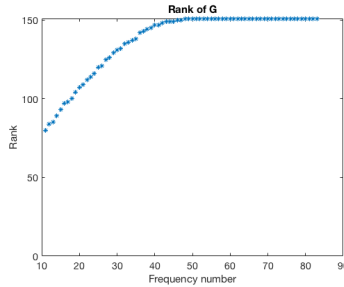


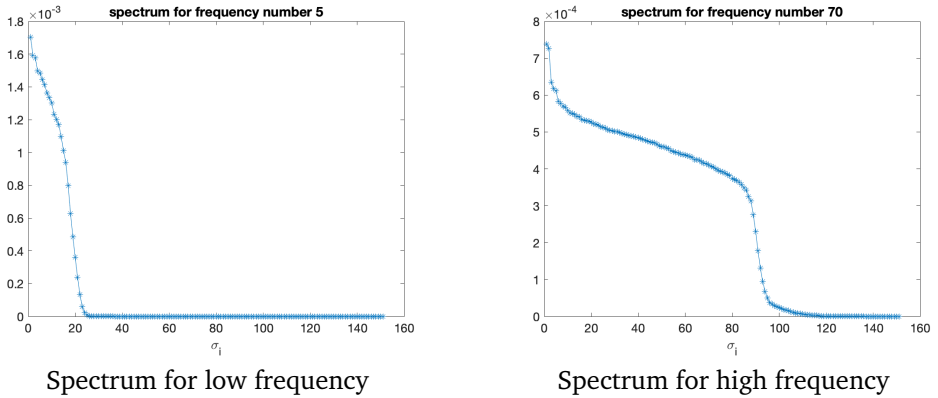
Figure 4.1: Rank of G for varying frequency

for the medium and high frequencies G is full rank, but for the low frequencies G is rank deficient. To illustrate the ill-posedness of the problem, we show the spectrum of G in figure 4.2 for two different frequencies: one for a low frequency and one for a high frequency. It is clear that for the low frequencies the problem is much more ill-posed. For the high frequencies, there are only a small number of very small singular values. For iterative solvers like LSQR and GMRES, ill-posedness of the matrix G (in the sense that G has small singular values), will lead to *semiconvergence* [57]. Semiconvergence is the phenomenon where the error between the reconstruction and the true X goes down initially, but increases as the iterations go on. This is due to the fact that noise, amplified by the small singular values, starts to enter the solution. Therefore, the method has to be terminated early to obtain a good solution. Hence, the number of iterations k takes the role of the regularization parameter. Due to the different nature of the spectra, it is important to regularize the low and high frequencies differently. Moreover, the rank deficiency of G for the low frequencies may lead to instabilities when using LSQR or CG. We have not observed any issues for this particular example.

4.4 Multi-Dimensional Deconvolution

As stated in the introduction, we investigate the usefulness of block methods on an MDD example in the context of redatuming. The aim is to estimate the impulse response $X(x_r, x_r)$ from the down- and upgoing wavefields $P^+(x_r, x_s)$ and $P^-(x_r, x_s)$ which satisfy the relations

$$XP^+ = P^-. \quad (4.3)$$

Figure 4.2: Spectrum of G for two different frequencies.

Here x_r denotes the receiver spacing and x_s denotes the source spacing. If we transpose (4.3) it is of the form (4.1) where P^{+H} is the model and P^{-H} is the data. We show our subsurface model in figure 4.3. For a description of the model used

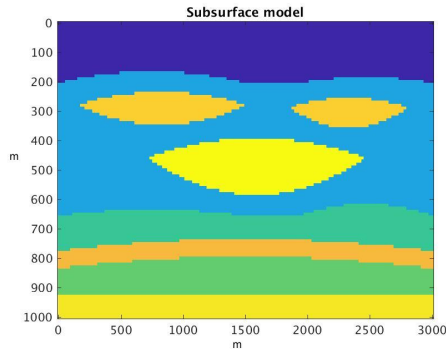


Figure 4.3: Subsurface model. We estimate the impulse response at 680 m.

above we refer the reader to [36]. We estimate the impulse response at depth level 680 m. For this example we have chosen equal source and receiver sampling and hence all matrices are square. In a more realistic case, there would be less source sampling and hence the problem will become underdetermined. In this case, we can resort to the normal equations $GG^H = D$. In our setting, where we have the same number of sources and receivers and in principle we don't have to resort to the normal equations. Interestingly, we have observed that the normal equations seem to have a regularizing effect in the sense that it seems to filter out high frequency components from the data. If we do not solve the normal equations but attempt to solve $GX = D$ instead, we get a noisy reconstruction. This is due to the fact that we are not able to solve the system $GX = D$ accurately enough for the higher frequencies. The filtering is due to the product $G^H D$ and we conjecture that

therefore it is better even for underdetermined systems to solve $G^H G = D$ instead of $GG^H = D$. Note that by resorting to the normal equations, the Arnoldi process becomes mathematically equivalent to Lanczos tridiagonalization and hence block GMRES becomes equivalent to block CG.

We now turn to the possible use of block methods. As we have stated earlier, in order for block methods to be efficient we need to be able to use deflation. The initial deflation is dependent on the rank of the data. We show the rank of D for all frequencies in figure 4.4. The y-axis is scaled to the size of the matrix. We can

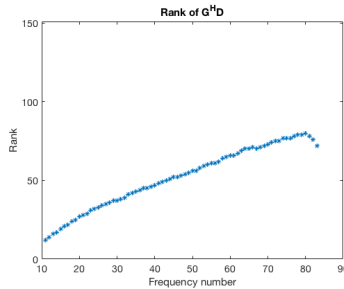


Figure 4.4: Rank of the data matrix

clearly see that the data is rank deficient and that the rank of the matrices increases with the frequency number. This suggests that the block methods are efficient for the lower frequencies, whereas for high frequencies the extra cost of the block methods starts to dominate. We compare the results of our deconvolution to the results obtained using the methodology in [36]. There, the deconvolution process has three additional constraints in the time domain. The first is reciprocity, which means that X should be symmetric. The second is a hyperbolic time window which suppresses non causal events. The third is a sparsity constraint which suppresses so called ringing artifacts. The deconvolution process is then solved using steepest descent.

We show the reconstruction for the block GMRES method versus the standard GMRES method for one source in figure 4.5. We see that the results for the standard GMRES versus the block-GMRES are comparable. However, near the edges the impulse response from GMRES shows a small tail that curves upwards, which is not present in the impulse response from block-GMRES. We see that both impulse responses show non-causal effects near the top and show some "ringing artifacts". However, in terms of computational time, the block-GMRES method is much faster. We've clocked the computational time to solve the linear systems for all frequencies over 10 runs. Block-GMRES averaged 1.1 seconds whereas GMRES averaged 23.1 seconds. We compare the impulse response from block-GMRES to the reference impulse response generated by the algorithm from [36] in figure 4.6. The impulse response obtained by deconvolution in the frequency domain is pretty close to the reference impulse response. Note that we do not impose any constraints: we are simply solving the linear systems. We see that if we regularize, i.e. terminate the iterative method, properly, we get a decent reconstruction. We

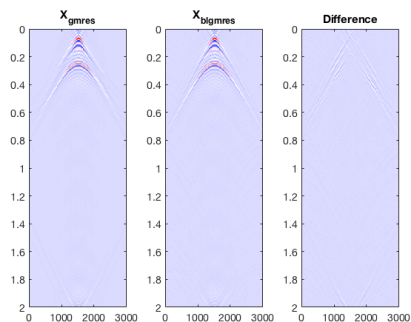


Figure 4.5: GMRES versus Block-GMRES for 1 source position.

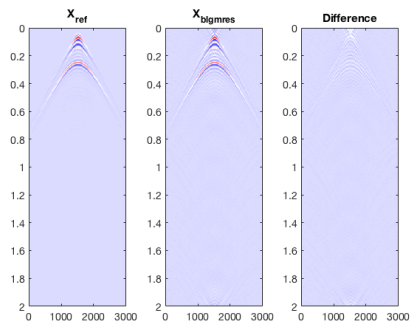


Figure 4.6: Reference versus Block-GMRES for 1 source position.

show the relative error for GMRES and block-GMRES versus the reference impulse response in figure 4.7. We see that the errors for GMRES and block-GMRES are

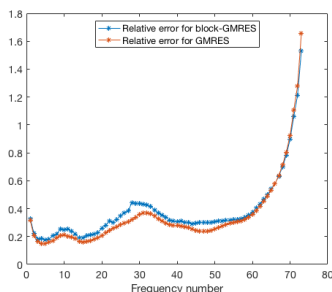


Figure 4.7: Error per frequency measured versus the reference impulse response.

comparable. Interestingly, we see that the error goes up with the frequency although we have shown that the spectra for the higher frequencies are relatively flat and have a small number of small singular values. We conjecture that there is

noise, i.e. unexplained physical events, in the data that affects the higher frequencies more than the low frequencies.

4.5 Discussion

We have shown that block methods can be used to efficiently solve the MDD problem in the frequency domain. We have shown that the quality of the reconstruction of the block-GMRES method is comparable to the quality of GMRES. It is however much faster. One of the benefits of the block method is that it is easier to regularize than normal GMRES, because the number of iterations one has to take is limited. For this problem, block-GMRES could be terminated after one iteration. For the low frequencies this is due to the fact that G is low-rank, whereas for the high frequencies the number of iterations is limited due to the large size of the input blocks. The most expensive computation of the block method is computing the rank of the data matrix D via the QR decomposition. Due to this overhead cost, we think that direct methods are competitive alternatives to iterative methods for MDD, e.g. via the use of an SVD. One important factor is that the matrix G is generally full, and due the fact that we have a large number of right-hand sides, a direct method may be more efficient. We have also seen that we are not able to solve the MDD problem to a satisfactory precision in the frequency domain. The block method can however be used as an initialization, after which the problem can be solved via a gradient based scheme with appropriate constraints in the time domain. We suspect that due to a good initial guess the gradient scheme will have converged rapidly. We do not see any sensible regularization in the frequency domain that could lead to a better solution. Lastly, we would like to point out the interesting work [71], where the authors compress the data using the Randomized Singular Value Decomposition. To overcome the problem of high-rank matrices in the high frequency regime, the authors propose the use of Hierarchical Semiseparable matrices (HSS matrices), the exploit the structure of seismic data. This seems to be a promising alternative to our work. For future research, we would like to solve the MDD problem in the time domain.



Seismic wavefield redatuming
with regularized
Multi-Dimensional Deconvolution



Abstract In seismic imaging the aim is to obtain an image of the subsurface using reflection data. The reflection data are generated using sound waves and the sources and receivers are placed at the surface. The target zone, for example an oil or gas reservoir, lies relatively deep in the subsurface below several layers. The area above the target zone is called the overburden. This overburden will have an imprint on the image. Wavefield redatuming is an approach that removes the imprint of the overburden on the image by creating so-called virtual sources and receivers above the target zone. The virtual sources are obtained by determining the impulse response, or Green's function, in the subsurface. The impulse response is obtained by deconvolving all up- and downgoing wavefields at the desired location. In this chapter, we pose this deconvolution problem as a constrained least-squares problem. We describe the constraints that are involved in the deconvolution and show that they are associated with orthogonal projection operators. We show different optimization strategies to solve the constrained least-squares problem and provide an explicit relation between them, showing that they are in a sense equivalent. We show that the constrained least-squares problem remains ill-posed and that additional regularization has to be provided. We show that Tikhonov regularization leads to improved resolution and a stable optimization procedure, but that we cannot estimate the correct regularization parameter using standard parameter selection methods. We also show that the constrained least-squares can be posed in such a way that additional nonlinear regularization is possible.

This chapter is partially based on the following publication:

N.A. Luiken and T. van Leeuwen. Seismic wavefield redatuming with regularized multi-dimensional deconvolution. *Inverse Problems*, 36:095010, 2020.

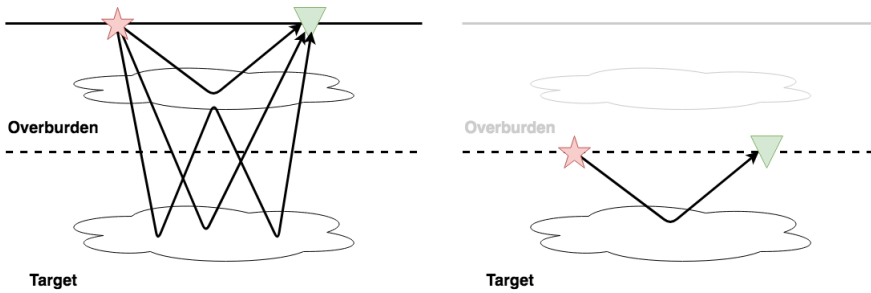


Figure 5.1: Schematic depiction of the redatuming procedure. A seismic survey consists of emitting waves into the subsurface from an impulsive source (red star) and recording the reflected response (green triangle). The goal is to transform recorded reflection data that includes the response of both the overburden and target zone to the response of the target zone only.

5.1 Introduction

In seismic imaging, one aims to obtain an image of the subsurface from reflection data. Here, an impulse source sends waves into the subsurface and the reflected response is recorded by an array of receivers. Reconstructing an image from the reflected data is an inverse problem that has been studied extensively (for an elaborate review see [108]). Wavefield redatuming is an inverse problem that appears in the same context and is often considered a pre-processing procedure. The goal of wavefield redatuming is to remove the effects of a part of the medium that is not of primary interest for imaging purposes (called the *overburden*), thereby making subsequent imaging of the *target zone* (e.g., an oil reservoir) easier. Redatuming transforms the response of the medium (overburden plus target zone) to the responses of the target zone only. This situation is depicted schematically in figure 5.1.

For an extensive overview of the redatuming problem we refer to [12, 104, 124, 126, 114, 113, 112, 98]. Imaging methods based on redatuming are, for example, Marchenko imaging [125] and JMI-res [36]. A key ingredient in all redatuming methods is *Multi-Dimensional Deconvolution*. Here, the data and unknown response are related through Multi-Dimensional Convolution with a given kernel. This leads to a linear, ill-posed inverse problem

$$p(t, x, x') = \iint g(t - s, x, y) q(s, y, x') ds dy,$$

where $p : \mathbb{R}^3 \rightarrow \mathbb{R}$ and $q : \mathbb{R}^3 \rightarrow \mathbb{R}$ are two given wavefields and $g : \mathbb{R}^3 \rightarrow \mathbb{R}$ is the unknown impulse response. After discretisation this yields a linear system of matrix equations

$$QG = P,$$

where Q is a block Toeplitz matrix, P contains the measurements and G represents the impulse response. The Toeplitz structure of the matrix is due to the convolution

operation, see, e.g. [60, page 35]. The block structure is due to the fact that a convolution in the time domain is multiplication in the frequency domain, and the block sizes are $n_r \times n_s$, where n_r is the number of receivers and n_s is the number of sources. It should be noted here that the usual distinction between model and data does not hold as both Q and P are contaminated with noise. This is because they are either measured directly or derived from measured data by some preprocessing steps. Nevertheless, we may attempt to solve the inverse problem by posing it as a regularized least-squares problem:

$$\min_G \|QG - P\|_F^2 + \lambda \|G\|_F^2, \quad (5.1)$$

where $\|\cdot\|_F$ denotes the Frobenius norm and $\lambda > 0$ is the regularization parameter. Because of the block-Toeplitz structure of Q this can be efficiently solved block-by-block in the Fourier domain. Typical difficulties that are encountered when solving such Multi-Dimensional Deconvolution problems are illustrated in the following example.

Example 5.1.1. *To illustrate the idea we show an example of source redatuming after the one in [104]. Here, $q(t, r, s)$ is the transmitted (downgoing) wavefield generated by a point source at location $(0, s)$ (indicated by red stars in figure 5.2 (a)) as recorded by the receivers (indicated by green triangles) at location $(500, r)$ and $p(t, r, s)$ is the reflected (upgoing) wavefield recorded at the same receivers. The impulse response g corresponds to a virtual experiment where sources are placed at $z = 500$ (depicted in 5.2 (b)). The corresponding wavefields q and p for a source at $(0, 0)$ as generated by a finite-difference modelling code are shown in figure 5.3 (a). The resulting estimate of g is depicted in figure 5.3 (b), alongside the true response (also generated by a finite-difference modelling code). Although the main features are reconstructed, we see some notable artifacts in the solution. In particular, we see non-physical events arriving before the first arrival. To counter such artifacts, we need a regularization method that takes into account such prior knowledge of the underlying physics.*

5.1.1 Approach and challenges

To include prior physical constraints in the reconstruction, we pose the inverse problem as a constrained least-squares problem:

$$\min_G \|QG - P\|_F^2 \quad \text{such that} \quad G \in \mathcal{A}, \quad (5.2)$$

where \mathcal{A} denotes a (convex) set of admissible solutions. Typical constraints include *causality*: $g(t, s, r) = 0$ when $t < \tau(s, r)$ for some given function τ , and *reciprocity*: $g(t, s, r) = g(t, r, s)$. Note that even though p and q should obey this constraint as well, noise or modelling errors may cause the unregularized solution to violate this constraint. Moreover, if Q has nullspace we may add components from the nullspace that violate the constraints. Enforcing reciprocity is nothing new, see e.g. [70, 69], and is frequently used by practitioners. However, here, we analyze the constraint from a more mathematical point of view. The difficulty in solving the inverse problem is that it is typically underdetermined and rank deficient. This is due to the fact that a seismic survey usually has fewer sources than receivers and

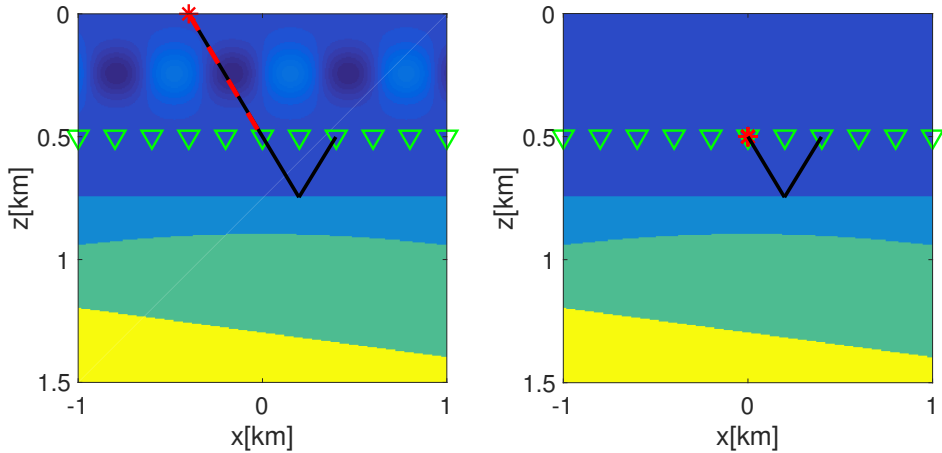


Figure 5.2: An example of source redatuming. The dashed red line indicates a wave traveling from source to receiver. The black line indicates a wave traveling the same path, but passing through the receiver after which it reflects and goes to another receiver. The difference between the two waves is the Green's function, shown in the figure on the right.

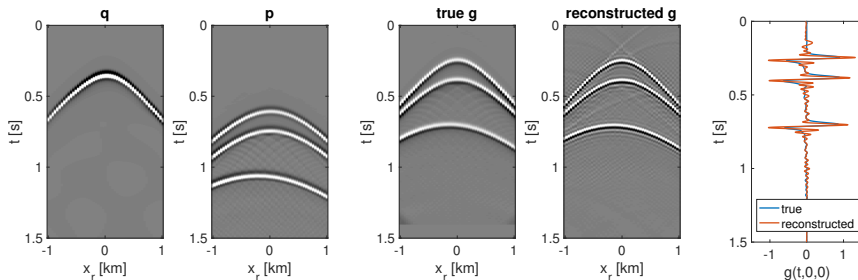


Figure 5.3: Wavefields $q(t, r, s = 0)$, $p(t, r, s = 0)$ associated with the subsurface in the example from figure (5.2) and the corresponding true and reconstructed response $g(t, r, s = 0)$.

the measurements are bandlimited. Therefore, further regularization besides the constraints is needed to stabilize the solution.

5.1.2 Contribution

In this chapter we pose Multi-Dimensional Deconvolution as a constrained least-squares problem. In particular, we treat the source-redatuming problem with causality and reciprocity constraints. We show that these constraints are associated with orthogonal projection operators. We describe different optimization methods to incorporate the constraints in the optimization and show explicit relations

between the methods. We show that the optimization methods are in a certain way equivalent, but that solving them numerically leads to different solutions. We show that even with incorporating the constraints, the problem exhibits semiconvergence, which means that the optimization scheme is still not stable. This means that the iterations have to be stopped at the appropriate point. Finally, we show that the addition of a Tikhonov penalty can further improve the reconstruction, but that standard parameter selection methods do not yield a good estimate for the regularization parameter. This makes the addition of a Tikhonov penalty impractical.

5.1.3 Outline

The chapter is organized as follows. In section (5.2) we describe the Multi-Dimensional Deconvolution (MDD) problem and set up the discretized linear system. In section (5.3) we describe how the reciprocity constraint and the causality constraint can be incorporated in the optimization. In section (5.4) we compare different optimization strategies and detail the difficulties in solving the optimization problem. Finally, in section (5.5), we draw our conclusions and add a short discussion and outlook.

5.2 Source redatuming

We start from the scalar wave equation in \mathbb{R}^n :

$$(c(x)^{-2}\partial_t^2 - \nabla^2) u(t, x, x') = f(t)\delta(x - x'), \quad (5.3)$$

where c is the soundspeed in the medium, and f is the time-signature of the source. The wave equation is furnished with appropriate boundary and initial conditions to ensure causal, outward propagating solutions. We split the medium in two parts; the *overburden* where $c(x) = c_0(x)$ and the *target zone*, where $c(x) = c_1(x)$. In essence, the inverse problem is as follows; given measurements of u at depth level \bar{z} we want to retrieve the impulse response of the target zone that excludes any effects from the overburden. In absence of horizontally propagating waves, we can split the solution to (5.3) into an upgoing and downgoing part, u_- and u_+ [20]. The upgoing and downgoing constituents can be obtained by solving a system of equations involving u and its vertical derivative [27, 123]. We now consider measurements of u at $x_r = (\bar{z}, r)$ originating from a source at $x_s = (z_0, s)$ and set $q(t, r, s) = u_+(t, x_r, x_s)$ and $p(t, r, s) = u_-(t, x_r, x_s)$. These two quantities are related to the upgoing response at (\bar{z}, r) to a downward radiating source at (\bar{z}, s) via convolution:

$$p(t, r, s) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(t - t', r, r') q(t', r', s) dt' dr'. \quad (5.4)$$

The goal is to recover g from (noisy) samples $p_{ijk} := p(t_i, r_j, s_k) + \epsilon_{ijk}$ and $q_{ijk} := q(t_i, r_j, s_k) + \delta_{ijk}$, with ϵ_{ijk} and δ_{ijk} representing the noise terms. For more details regarding the derivation of this relation we refer to [124, 126, 98]. A concrete example illustrating the ill-posedness of the problem is given below.

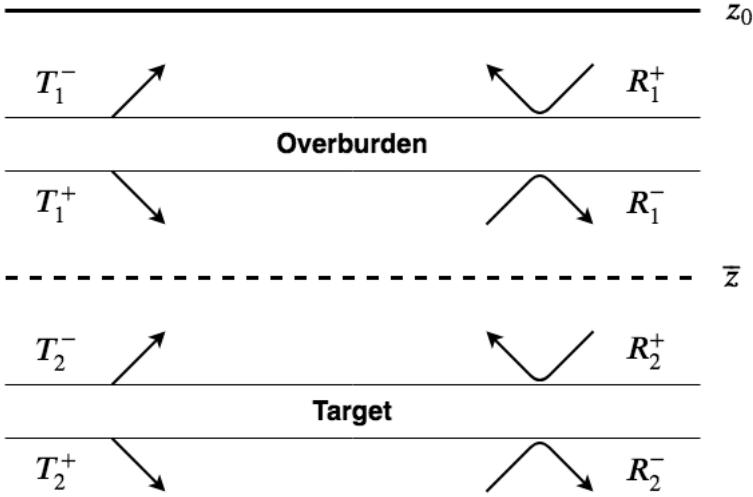


Figure 5.4: Schematic depiction of the up- and downgoing responses of the overburden and target. A superscript $-$ indicates the responses of the medium to an upgoing plane wave, while a superscript $+$ indicates the response to a downgoing plane wave.

5.2.1 Analysis for layered media

For horizontally layered media, the wavefields and impulse response can be expressed as

$$p(t, r, s) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{p}(\omega, \xi) \exp(-i(\omega t + \xi(r - s))) d\omega d\xi,$$

and similarly for q and g . The forward relation (5.4) can be expressed in terms of the Fourier transformed quantities as

$$\hat{p}(\omega, \xi) = \hat{q}(\omega, \xi) \cdot \hat{g}(\omega, \xi).$$

We can get an explicit expression for the up and downgoing solutions as follows. For a horizontally layered medium we can explicitly factorize the wave equation in the Fourier domain:

$$\hat{u}'_{\pm}(z) \mp k(z)\hat{u}_{\pm}(z) = 0,$$

with $k(z) = \sqrt{c(z)^{-2}\omega^2 - |\xi|^2}$ where ω is the temporal frequency and ξ the horizontal wavenumber. We can now think of the response of the overburden and target in terms of incoming plane waves. We denote the transmitted and reflected response of the overburden to an up/downgoing plane wave by T_1^{\pm} , R_1^{\pm} and likewise we denote the responses of the target zone by T_2^{\pm} , R_2^{\pm} . Figure 5.4 illustrates the situation. The downgoing response of the entire medium to a downgoing plane wave $e^{ik_0 z}$ at \bar{z} can now be expressed as

$$\hat{q} = T_1^+ + R_1^- R_2^+ T_1^+ + \dots = (1 - R_1^- R_2^+)^{-1} T_1^+.$$

Likewise, the upgoing response at \bar{z} can be described by

$$\hat{p} = R_2^+ T_1^+ + R_2^+ R_1^- R_2^+ T_1^+ + \dots = (1 - R_1^- R_2^+)^{-1} R_2^+ T_1^+.$$

We immediately see that

$$\hat{p}/\hat{q} = R_2^+,$$

which is the impulse response of the target zone to a downgoing planewave measured at $z = \bar{z}$. The main cause of the ill-posedness of the inverse problem is the bandlimited nature of the measured responses; they only contain propagating modes for which $|\xi| < \omega/c$. In practice, the measured response contains the imprint of the source wavelet $\hat{f}(\omega)$ and is further band-limited in ξ because it is measured with a finite array of receivers. The estimated response is then given by

$$\hat{g} = \frac{(\hat{q} + \hat{\epsilon})^* (\hat{p} + \hat{\delta})}{|\hat{q} + \hat{\epsilon}|^2 + \lambda},$$

where $\hat{\epsilon}$ and $\hat{\delta}$ represent the measurement noise and $\lambda > 0$ is the regularization parameter (cf. (5.1)). This can be decomposed as

$$\hat{g} = \frac{|\hat{q} + \hat{\epsilon}|^2}{|\hat{q} + \hat{\epsilon}|^2 + \lambda} R_2^+ - \frac{(\hat{q} + \hat{\epsilon})^* \hat{\epsilon}}{|\hat{q} + \hat{\epsilon}|^2 + \lambda} R_2^+ + \frac{(\hat{q} + \hat{\epsilon})^* \hat{\delta}}{|\hat{q} + \hat{\epsilon}|^2 + \lambda}.$$

Thus the error can be bounded as

$$|\hat{g} - R_2^+| \leq \frac{\lambda}{|\hat{q} + \hat{\epsilon}|^2 + \lambda} \cdot |R_2^+| + \frac{|\hat{q} + \hat{\epsilon}|}{|\hat{q} + \hat{\epsilon}|^2 + \lambda} \cdot (|\hat{\epsilon}| \cdot |R_2^+| + |\hat{\delta}|),$$

from which we recognize a bias and variance term. Notably, we see the regularizing effect that λ has on both sources of noise.

Example 5.2.1. *As a concrete example we consider a medium with three horizontal layers with sound speed c_i for $z \in [z_i, z_{i+1})$. The redatuming level is set at $\bar{z} = (z_1 + z_2)/2$. The corresponding transmission and reflection responses can then be explicitly expressed in terms of $k_i = \sqrt{(\omega/c_i)^2 - \xi^2}$:*

$$T_1^+ = \frac{2k_0}{k_1 + k_0} e^{-ik_1 h_1/2}, \quad R_1^- = \frac{k_1 - k_0}{k_1 + k_0} e^{-ik_1 h_1}, \quad R_2^+ = \frac{k_1 - k_2}{k_1 + k_2} e^{-ik_1 h_1},$$

with $h_1 = z_2 - z_1$. Two typical examples of the corresponding spectra \hat{q} , \hat{p} and \hat{g} are illustrated in figure 5.5. We see that when $c_0 > c_1$, part of the impulse response, \hat{g} , is in the null-space of \hat{q} – the modes for which $|\xi| > \omega/c_0$. A properly regularized inversion will thus at best give an estimate of \hat{g} that is restricted to the support of \hat{q} in the (ω, ξ) domain.

5.2.2 Discretization

Assuming the signals are regularly sampled and the spatial samples are co-located, i.e. $t_i = i \cdot \Delta t$ for $i = 0 \dots n_t - 1$, $r_j = j \cdot \Delta r$ for $j = 0 \dots n_r - 1$ and $r_k = k \cdot \Delta s$ for $k = 0 \dots n_s - 1$, we can represent the signals in terms of their samples as

$$p(t, r, s) = \sum_{ijk} p_{ijk} \operatorname{sinc} \left(\frac{t - t_i}{\Delta t} \right) \operatorname{sinc} \left(\frac{r - r_j}{\Delta r} \right) \operatorname{sinc} \left(\frac{s - s_k}{\Delta s} \right),$$

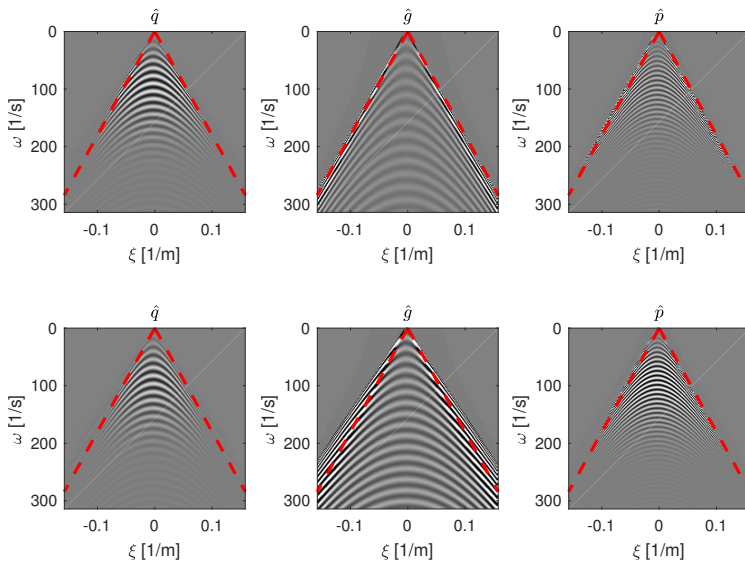


Figure 5.5: Spectrum of the responses \hat{q} , \hat{g} and \hat{p} for a layered medium with $z_0 = 0$ m, $z_1 = 500$ m, $z_2 = 1000$ m and $c_0 = 1500$ m/s, $c_1 = 1800$ m/s, $c_2 = 2000$ m/s (top), $c_0 = 1800$ m/s, $c_1 = 1500$ m/s, $c_2 = 2000$ m/s (bottom). In the top scenario, we see that the spectrum of \hat{q} has the same support as that of \hat{g} , making it in principle possible to retrieve \hat{g} completely from \hat{q} . In the bottom scenario, the spectrum of \hat{q} has a narrower support than that of \hat{g} , making it impossible to recover the complete spectrum of \hat{g} from \hat{p} .

$$q(t, r, s) = \sum_{ijk} q_{ijk} \operatorname{sinc}\left(\frac{t-t_i}{\Delta t}\right) \operatorname{sinc}\left(\frac{r-r_j}{\Delta r}\right) \operatorname{sinc}\left(\frac{s-s_k}{\Delta s}\right),$$

and

$$g(t, r, s) = \sum_{ijk} g_{ijk} \operatorname{sinc}\left(\frac{t-t_i}{\Delta t}\right) \operatorname{sinc}\left(\frac{r-r_j}{\Delta r}\right) \operatorname{sinc}\left(\frac{r-r_k}{\Delta r}\right).$$

Using the orthogonality relations of the normalized sinc function, we can re-write this as a system of matrix-equations with a block-circulant structure¹:

$$P = QG,$$

with

$$Q = \begin{pmatrix} Q_0 & Q_{n_t-1} & \dots & Q_1 \\ Q_1 & Q_0 & \dots & Q_2 \\ \vdots & \ddots & \ddots & \vdots \\ Q_{n_t-1} & \dots & Q_1 & Q_0 \end{pmatrix}, \quad G = \begin{pmatrix} G_0 \\ G_1 \\ \vdots \\ G_{n_t-1} \end{pmatrix}, \quad P = \begin{pmatrix} P_0 \\ P_1 \\ \vdots \\ P_{n_t-1} \end{pmatrix}.$$

Here, $Q_i \in \mathbb{R}^{n_s \times n_r}$ is a matrix with elements q_{ijk} , P is a block matrix with n_t blocks $P_i \in \mathbb{R}^{n_s \times n_r}$ with elements p_{ijk} and G is a block matrix with n_t blocks $G_i \in \mathbb{R}^{n_r \times n_r}$. Since a circulant matrix diagonalizes under the Discrete Fourier Transform, F_{n_t} (with entries $\exp\left(i\frac{2\pi ij}{n_t}\right)$), we can express Q as

$$Q = (F_{n_t} \otimes I_{n_r \cdot n_s}) \widehat{Q} (F_{n_t}^{-1} \otimes I_{n_r \cdot n_s}),$$

where

$$\widehat{Q} = \operatorname{blockdiag}\left(\widehat{Q}_0, \widehat{Q}_1, \dots, \widehat{Q}_{n_t}\right),$$

and

$$\widehat{Q}_i = \sum_{j=0}^{n_t-1} \exp\left(i\frac{2\pi ij}{n_t}\right) Q_j.$$

This means we can decouple the system into n_t matrix equations

$$\widehat{Q}_i \widehat{G}_i = \widehat{P}_i.$$

The constraints, however, may not decouple in this fashion. Moreover, it is not very attractive to have to estimate a separate regularization parameter for each frequency separately. We therefore stick with a time-domain formulation. Matrix-vector multiplication with Q are carried out in the frequency-domain, however, for computational efficiency.

¹assuming that we are looking for a solution that is periodic in time

5.3 Constrained least squares

In this section we describe how to solve the constrained least-squares problem (5.2) and describe two relevant constraints, causality and reciprocity, in detail. For each constraint we describe an orthogonal projection operator $P_{\mathcal{A}} : \mathbb{R}^{n_t \times n_r \times n_r} \rightarrow \mathbb{R}^{n_t \times n_r \times n_r}$ i.e., the operator that solves

$$P_{\mathcal{A}}(G) = \operatorname{argmin}_{G' \in \mathbb{R}^{n_t \times n_r \times n_r}} \|G' - G\|_F^2 \quad \text{s.t. } G' \in \mathcal{A}.$$

Related to the projection operator is the penalty operator, $L_{\mathcal{A}} : \mathbb{R}^{n_t \times n_r \times n_r} \rightarrow \mathbb{R}^{n_t \times n_r \times n_r}$, defined by

$$P_{\mathcal{A}}(G) + L_{\mathcal{A}}(G) = G \quad \forall G \in \mathbb{R}^{n_t \times n_r \times n_r}.$$

5.3.1 The constraints

The system has to satisfy two binding constraints. The first constraint is source-receiver reciprocity. This prior requires that a wave travels from source location to receiver location in the same time as a wave traveling from receiver location to source location. This means that the impulse response G has to satisfy $g_{ijk} = g_{ikj}$, for all i, j, k . The second constraint is causality which means that $g_{ijk} = 0$ for $i < \tau_{jk}$ for some given symmetric matrix τ . Below, we formulate the projection and penalty operators for each constraint.

Causality

The set of causal solutions is given by

$$\mathcal{C} = \{G \in \mathbb{R}^{n_t \times n_r \times n_r} \mid g_{ijk} = 0 \text{ for } 1 \leq i < \tau_{jk}, 1 \leq j \leq n_r, 1 \leq k \leq n_r\},$$

where $\tau_{jk} > 1$ are given. The corresponding projection and penalty operators are given by

$$P_{\mathcal{C}}(G)_{ijk} = \begin{cases} g_{ijk} & \text{if } i \geq \tau_{jk} \\ 0 & \text{otherwise} \end{cases}$$

$$L_{\mathcal{C}}(G)_{ijk} = \begin{cases} g_{ijk} & \text{if } i < \tau_{jk} \\ 0 & \text{otherwise} \end{cases}$$

Reciprocity

The set of solutions satisfying reciprocity is given by

$$\mathcal{R} = \{G \in \mathbb{R}^{n_t \times n_r \times n_r} \mid g_{ijk} = g_{ikj} \text{ for } 1 \leq i \leq n_t, 1 \leq j \leq n_r, 1 \leq k \leq n_r\}.$$

The dimension of \mathcal{R} is $\frac{n_t \cdot n_r \cdot (n_r + 1)}{2}$ and an orthogonal basis for \mathcal{R} and its complement can be easily constructed. An example for $n_t = 1$, $n_r = 3$ is shown in figure 5.6. The corresponding projection and penalty operators are given by

$$P_{\mathcal{R}}(G)_{ijk} = \frac{1}{2} (g_{ijk} + g_{ikj}),$$

$$L_{\mathcal{R}}(G)_{ijk} = \frac{1}{2} (g_{ijk} - g_{ikj}).$$

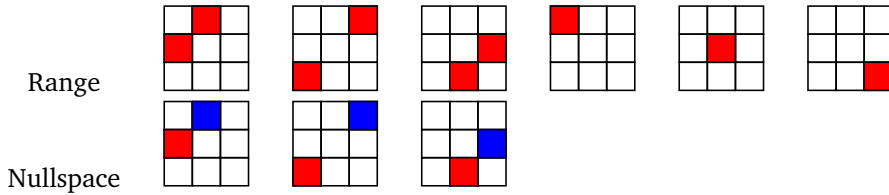


Figure 5.6: Basis elements for the symmetric and anti-symmetric 3×3 matrices. Blue is -1 and red is 1.

Intersection

To include both constraints we need the projection and penalty operators for the set $\mathcal{R} \cap \mathcal{C}$. In general, one cannot naively project on to the intersection by concatenating the individual projection operators. However, because the causality constraint will not violate reciprocity and vice versa, the constraints are consistent here. Therefore, we may project onto the intersection of the two sets. The orthogonal projection on to $\mathcal{R} \cap \mathcal{C}$ is thus simply given by

$$P_{\mathcal{R} \cap \mathcal{C}}(G)_{ijk} = \begin{cases} \frac{1}{2}(g_{ijk} + g_{ikj}) & \text{if } i \geq \tau_{jk} \\ 0 & \text{otherwise} \end{cases}, \quad (5.5)$$

and the corresponding penalty operator by

$$L_{\mathcal{R} \cap \mathcal{C}}(G)_{ijk} = \begin{cases} \frac{1}{2}(g_{ijk} - g_{ikj}) & \text{if } i \geq \tau_{jk} \\ g_{ijk} & \text{otherwise} \end{cases}, \quad (5.6)$$

An example of the orthogonal basis of $\mathcal{R} \cap \mathcal{C}$ and its complement for $n_t = 1$, $n_r = 3$ is shown in figure 5.7.

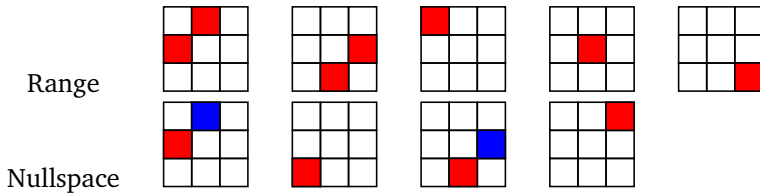


Figure 5.7: Basis elements for the symmetric matrices with $G_{31} = G_{13} = 0$. Note that the second element for the symmetric and anti-symmetric matrices have been replaced by basis elements making the corresponding entries 0.

5.3.2 Solving the constrained least-squares problem

Before continuing we first express (5.2) in a more convenient form

$$\min_{\mathbf{g}} \|(I \otimes Q)\mathbf{g} - \mathbf{p}\|_2^2 \quad \text{s.t.} \quad L\mathbf{g} = \mathbf{0}, \quad (5.7)$$

where $\mathbf{g} = \text{vec}(G) \in \mathbb{R}^n$, $\mathbf{p} = \text{vec}(P) \in \mathbb{R}^m$ with $n = n_t \cdot n_r^2$ and $m = n_t \cdot n_r \cdot n_s$, and $L \in \mathbb{R}^{n \times n}$ is the penalty operator corresponding to the admissible set \mathcal{A} . In the remainder of the chapter we will refer to $I \otimes Q$ as Q for ease of notation. Given an orthogonal basis, A , for \mathcal{A} we have $P = AA^T$ and $L = I - AA^T$.

The solution to (5.7) can be expressed as $\mathbf{g} = A\mathbf{y}$ where \mathbf{y} solves

$$\min_{\mathbf{y}} \|QA\mathbf{y} - \mathbf{p}\|_2^2.$$

Hence, the (minimum-norm) solution to (5.7) is given by

$$\mathbf{g} = A(A^T Q^T Q A)^\dagger A^T Q^T \mathbf{p}. \quad (5.8)$$

We note that in practical applications, Q , P and L are never formed explicitly; their action is computed on-the-fly in a matrix-free fashion. Next, we discuss three approaches for finding a solution to (5.7).

All-at-once

The first-order Karush-Kuhn-Tucker optimality conditions corresponding to (5.7) lead to a saddle-point problem

$$\begin{pmatrix} 0 & Q^T & L \\ Q & -I & 0 \\ L & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{g} \\ \mathbf{r} \\ \boldsymbol{\mu} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{p} \\ \mathbf{0} \end{pmatrix}, \quad (5.9)$$

from which the residual \mathbf{r} can be eliminated to yield

$$\begin{pmatrix} Q^T Q & L \\ L & 0 \end{pmatrix} \begin{pmatrix} \mathbf{g} \\ \boldsymbol{\mu} \end{pmatrix} = \begin{pmatrix} Q^T \mathbf{p} \\ \mathbf{0} \end{pmatrix}. \quad (5.10)$$

This system of equations can be readily solved with an iterative method like MINRES to yield the desired solution [95]. Applying the substitution $\mathbf{g} = A\mathbf{y}$ the system reduces to

$$Q^T Q A \mathbf{y} + L \boldsymbol{\mu} = Q^T \mathbf{p}. \quad (5.11)$$

Projecting onto A^T yields $A^T Q^T Q A \mathbf{y} = A^T Q^T \mathbf{p}$, whose solution is given by $\mathbf{y} = (A^T Q^T Q A)^\dagger A^T Q^T \mathbf{p}$, and hence $\mathbf{g} = A(A^T Q^T Q A)^\dagger A^T Q^T \mathbf{p}$, which coincides with (5.8).

Note that we do not need to explicitly form the matrix in order to do so; we can easily compute matrix-vector multiplications with the system matrix on-the-fly.

Right preconditioning

We can explicitly eliminate the constraint in (5.7) via a substitution $\mathbf{g} = P\tilde{\mathbf{g}}$:

$$\min_{\tilde{\mathbf{g}}} \|QP\tilde{\mathbf{g}} - \mathbf{p}\|_2^2, \quad (5.12)$$

where P is the projection operator in (5.5). Here, the operator P makes the solution satisfy reciprocity and causality by projecting onto the space of causal and reciprocal solutions. The resulting least-squares problem may not have a unique solution due to overlap of the null-spaces of Q and P . A minimum norm solution can be readily

obtained using LSQR. To see that this is indeed equivalent to (5.7), we note that the minimum norm solution to (5.12) is given by

$$\mathbf{g} = P(PQ^TQP)^\dagger PQ^T\mathbf{p},$$

which is indeed equivalent to (5.8). To show this, use that $(AHA^T)^\dagger = AH^\dagger A^T$ where $A^T A = I$ and H is symmetric.

Quadratic penalty

We may incorporate the constraints via a quadratic penalty and solve

$$\min_{\mathbf{g}} \|Q\mathbf{g} - \mathbf{p}\|_2^2 + \rho \|L\mathbf{g}\|_2^2. \quad (5.13)$$

We can solve this in a straightforward fashion using LSQR by introducing the augmented matrix $[Q^T, \sqrt{\rho}L]^T$. In [74] and [29] this method is used to analyze symmetric solutions to matrix equations with $\rho = 1$. For ill-posed inverse problems, the choice of ρ may be slightly more involved. If the system admits a symmetric solution then this approach works for any ρ . However, if \mathbf{p} and possibly also \mathbf{g} are perturbed, the solution may no longer admit a symmetric solution. In this case, we have to find the best symmetric solution and in this case there is a trade-off between symmetry and data misfit. To ensure a symmetric solution, we have to choose ρ large enough. Consider the following example.

Example 5.3.1. Let $A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$, $X = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $AX = B = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$.

One can consider the matrices A and B as perturbed such that the solution is now X , which is not symmetric. The solution using (5.10) or (5.12) is

$$X = \begin{bmatrix} 1/2 & 1/3 & 1/2 \\ 1/3 & 1 & 1/3 \\ 1/2 & 1/3 & 1/2 \end{bmatrix}, \text{ and } A^\dagger B = \begin{bmatrix} 1/2 & 1/2 & 1/2 \\ 0 & 1 & 0 \\ 1/2 & 1/2 & 1/2 \end{bmatrix}. \text{ Plugging in } \rho = 1 \text{ in (5.13)}$$

gives the solution $\begin{bmatrix} 1/2 & 3/8 & 1/2 \\ 1/4 & 1 & 3/8 \\ 1/2 & 1/4 & 1/2 \end{bmatrix}$, which is not symmetric. Plugging in $\rho = 10000$

gives the desired result. Increasing the value of ρ will gradually make the solution more symmetric.

Equivalence between the solutions

We now show the relation between the quadratic penalty and the preconditioning approach using the standard form transformation after [34], see also [58]. First, we split the solution in two parts;

$$\mathbf{g} = \mathbf{g}_A + \mathbf{g}_B,$$

where $\mathbf{g}_A \in \mathcal{A}$ represents the admissible part with $L\mathbf{g}_A = 0$ and $\mathbf{g}_B \in \mathcal{B}$ is the remainder. Here, we take \mathcal{B} to be the Q -orthogonal complement of \mathcal{A} , meaning that $\mathbf{g}_A^T Q^T Q \mathbf{g}_B = 0$. This leads to two different optimization in terms of \mathbf{g}_A and \mathbf{g}_B . We introduce the corresponding oblique projection operators L_Q and P_Q . Given an orthonormal basis, A , for \mathcal{A} , these are explicitly given by

$$P_Q = A(QA)^\dagger Q, \quad L_Q = I - P_Q.$$

The problem then decomposes in two parts

$$\mathbf{g}_A = \underset{\mathbf{g}}{\operatorname{argmin}} \quad \|Q\mathbf{A}\mathbf{g} - \mathbf{p}_A\|_2^2, \quad (5.14)$$

$$\mathbf{g}_B = \underset{\mathbf{g}}{\operatorname{argmin}} \quad \|QL_QL^\dagger\mathbf{g} - \mathbf{p}_B\|_2^2 + \rho\|\mathbf{g}\|_2^2, \quad (5.15)$$

where $\mathbf{p}_A \in \mathcal{R}(QP_Q)$ and $\mathbf{p}_B \in \mathcal{R}(QL_Q)$. First note that we may replace \mathbf{p}_A in (5.14) by \mathbf{p} without changing the solution. The solution to (5.14) is given by $\mathbf{g} = (A^TQ^TQA)^\dagger A^TQ^T\mathbf{p}$ and hence $\mathbf{g}_A = A(A^TQ^TQA)^\dagger A^TQ^T\mathbf{p}$. Thus, \mathbf{g}_A coincides with the solution of (5.12). If the system $Q\mathbf{g} = \mathbf{p}$ has an admissible solution we have $\mathbf{p}_B = 0$ and any non-zero value of ρ will suffice to suppress non-admissible solutions. The same argument holds, when all non-admissible solutions are in the null-space of Q . If, like in our example, there is no symmetric solution, then ρ has to be chosen large enough to make $\mathbf{g}_B = 0$. This means that $\rho > \sigma_1(QL_QL^\dagger)$, where $\sigma_1(\cdot)$ denotes the largest singular value. In practice, we can not calculate this singular value due to the size of the system. Moreover, the singular values of QL_QL^\dagger are not related to the singular values of Q and L_Q or L . Even an iterative scheme like the power method does not work, because we can not compute the matrix L_Q , nor are matrix-vector multiplications available.

Additional regularization

Note that incorporating the reciprocity and causality constraints still allows for additional, possibly nonlinear regularization. This excludes the use of the KKT system, as it can only deal with linear (in)equality constraints and we can not use it in combination with nonlinear regularizers. However, the systems (5.12) and (5.13) allow for additional nonlinear regularization such that the constraints are still satisfied, where the system (5.12) is preferred for simplicity.

We see two possible nonlinear regularizers. The first is an ℓ_1 penalty on the impulse response in the curvelet domain, which has shown to be a good basis to represent seismic data [24], [62]. This approach has been used in the specific context of separating primaries and multiples via deconvolution, in the works [64, 81, 80, 79]. Another possibility is low rank minimization. Seismic data is shown to have low rank in the midpoint-offset domain [4], [75], which can be enforced by penalizing the nuclear norm of the impulse response in the midpoint-offset domain.

5.4 Numerical experiments

Our numerical experiments are carried on wavefields generated for the subsurface model presented in [36]. We show the subsurface model in figure (5.8). We discuss the deconvolution problem for two different source to receiver sampling ratios, namely 1:1 and 1:4. The 1:4 source to receiver ratio is most realistic, but we have added the other setup to investigate the effects of undersampling. For our experiments we always have 151 receivers spread 20 meters apart and we have 512 time samples. Depending on the source to receiver ratio, we either have 151 or 38 sources. The free-surface multiples have been removed by a pre-processing algorithm.

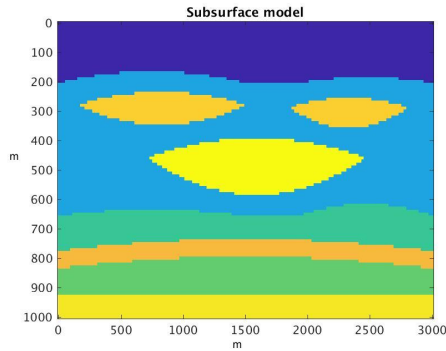


Figure 5.8: The subsurface model. The sources and receivers are redatumed at 680 m.

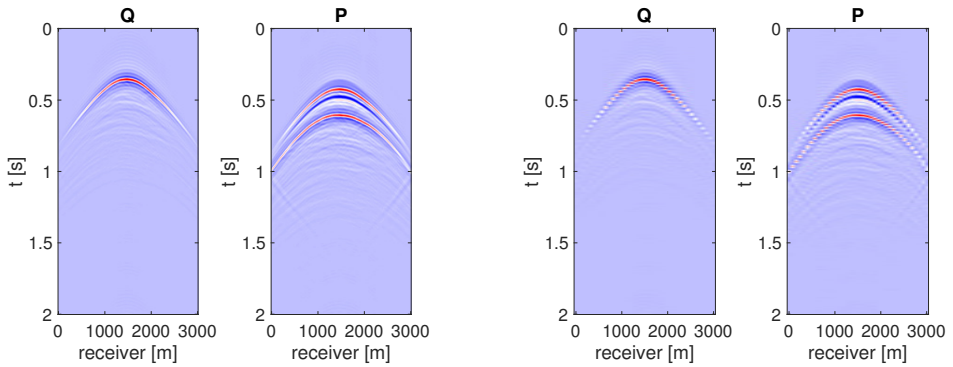


Figure 5.9: The wavefields for the different sampling scenarios. The left panel shows the impulse response for the 1:1 sampling ratio and the right panel shows the impulse response for the 1:4 sampling ratio. Both impulse responses correspond to one source, which is located at the center of the domain.

5.4.1 The wavefields and the impulse response

In figures (5.9) and (5.10) we show the wavefields and the impulse response that has to be inverted for, both for the center source. Both wavefields contain modeling errors that have to be accounted for. We clearly see some ringing artifacts and the wavefield P clearly contains some modeling errors near the boundary. We also see that the wavefields for the 1:4 sampling clearly suffers from limited source sampling. The data have been modeled by the Full Wavefield Modeling scheme (FWMod) [13]. We benchmark our results against the impulse response obtained in [36], which is obtained by nonlinear inversion. We will consider this the true impulse response. For further details we refer to [36].

Firstly, it is important to investigate whether the problem is actually ill-posed. It is clear that the system for the 1:4 source to receiver ratio is ill-posed, due to the fact that they lead to underdetermined systems. The 1:1 source to receiver ratio leads

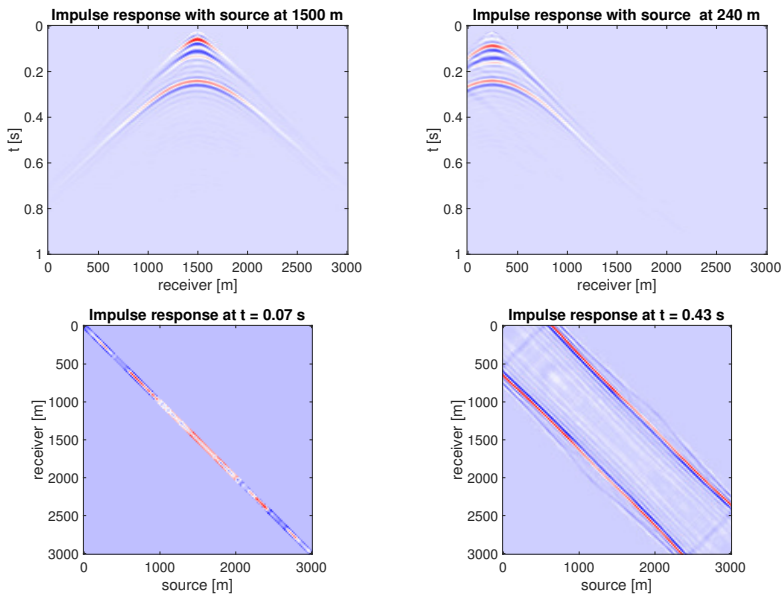
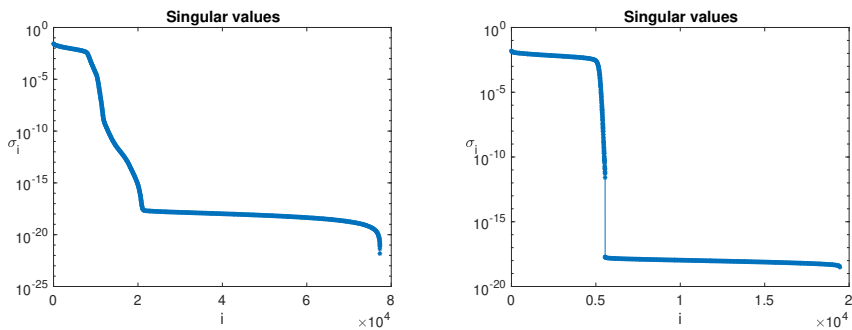


Figure 5.10: The impulse response in two different domains. The top panel shows the impulse response corresponding to a fixed source at different locations of the domain. The bottom panel shows the impulse response for a fixed time at different moments.



(a) 1:1 sampling. Rank of the matrix is 16912.

(b) 1:4 sampling. Rank of the matrix is 5548.

Figure 5.11: Singular values for different sampling scenarios.

to a square system which may be well-posed. It turns out it is not, as can be seen from the singular values, which we show in figure (5.11a). From figure (5.11a) we see that for the 1:1 source to receiver sampling the system is rank deficient, which means that additional regularization is required. As we have stated before, it is not given that the symmetry constraint and the time window will be sufficient prior information to regularize the problem. From figure (5.11b) we see that even in the

undersampled case the problem is rank deficient. However, in this case the very small singular values correspond to frequencies that have very little impact anyway.

	1:4 sampling	1:1 sampling
Preconditioned	0.17	0.14
Tikhonov	0.17	0.14
KKT	0.13	0.10

Table 5.1: Relative error for three different optimization strategies.

5.4.2 Results for different optimization strategies

In this section we discuss the results for the different optimization strategies for the two different sampling scenarios. We compare the results to the true solution using the relative error, given by

$$\text{err} = \frac{\|\mathbf{g} - \mathbf{g}_{\text{true}}\|_2}{\|\mathbf{g}_{\text{true}}\|_2}.$$

We present the errors in table (5.1). We solve the Tikhonov system and the preconditioned system using LSQR. The KKT system is solved using MINRES. The Tikhonov approach and the preconditioned system have almost identical solutions but the KKT solution is slightly better. Interestingly, although the wavefields for the 1:4 sampling ratio contain much less information, the error does not increase much. The reported results are optimal in the sense that we have chosen the number of iterations that gives the lowest error, by comparing the solution with the true solution.

5.4.3 Semiconvergence

The projection constraints do not stabilize the solution and semiconvergence may still be observed. In figure (5.12) we show the semiconvergence for the three different optimization strategies for the 1:4 sampling ratio. The Tikhonov and KKT approach show interesting convergence behavior where the error goes down in stages. The convergence is much slower than for the preconditioned system, but the area around the minimizer is flat as opposed to the minimizer of the preconditioned system. The Tikhonov approach requires half the iterations of the KKT system. Due to the slower convergence and the plateau the solution of the Tikhonov approach and the KKT system seem more stable.

In figure (5.13) we show the impulse response for different iteration counts. What we see is that the effects of semiconvergence are large amplitudes and strong ringing artifacts. Therefore the presence of ringing artifacts could potentially be used as a stopping criterion, although that would be difficult to automate, because it is a visual criterion. Note that the ringing artifacts are still present for the optimal number of iterations, but only at a minimum.

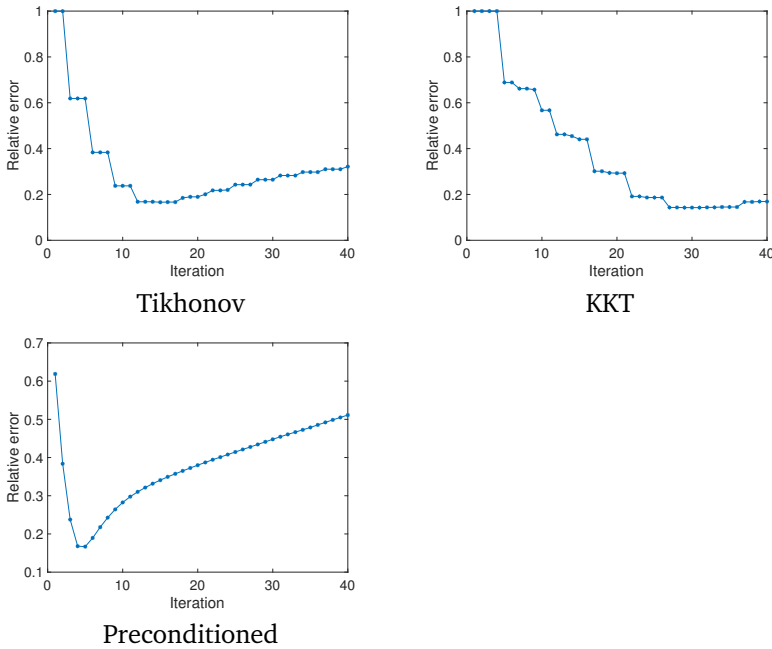


Figure 5.12: Semiconvergence for three different optimization strategies for the 1:4 sampling ratio.

5.4.4 Additional Tikhonov regularization

In order to stabilize the solution, and possibly even improve accuracy, we can use an additional Tikhonov penalty. This leads to the following systems:

$$\min_{\mathbf{g}} \|QP\mathbf{g} - \mathbf{p}\|_2^2 + \lambda\|\mathbf{g}\|_2^2 \tag{5.16}$$

$$\min_{\mathbf{g}} \|Q\mathbf{g} - \mathbf{p}\|_2^2 + \|L\mathbf{g}\|_2^2 + \lambda\|\mathbf{g}\|_2^2 \tag{5.17}$$

$$\begin{bmatrix} Q^T Q + \lambda I & L \\ L & 0 \end{bmatrix} \begin{bmatrix} \mathbf{g} \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} Q^T \mathbf{p} \\ 0 \end{bmatrix} \tag{5.18}$$

To check whether this approach is effective we first determine the optimal λ_{opt} , given by

$$\lambda_{\text{opt}} = \underset{\lambda}{\operatorname{argmin}} \|\mathbf{g}_\lambda - \mathbf{g}_{\text{true}}\|,$$

where \mathbf{g}_λ is a solution of a given system for a given λ . The optimal λ and the associated error are reported in table (5.2). The errors decrease for both sampling scenarios and all three optimization strategies, although the improvement for the KKT system is not as big as for the other two. In figure (5.14) we show the stabilizing effect of Tikhonov regularization, which is in line with the results shown in [49] and [26]. Interestingly, the optimal λ is not necessarily the one that leads to a stable solution. The Tikhonov system and the KKT system show similar behavior.

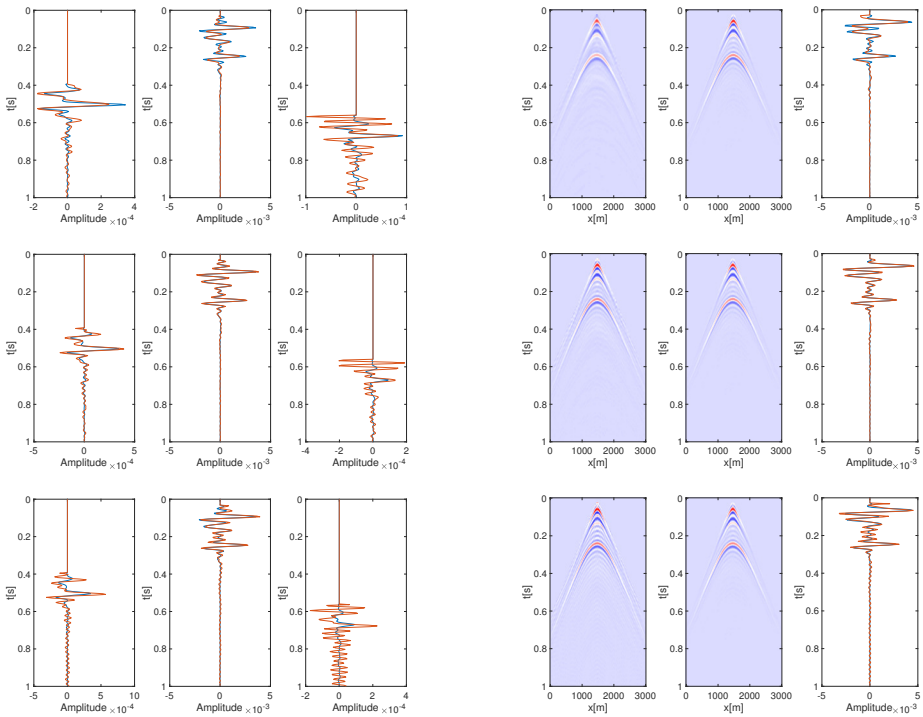


Figure 5.13: Some images of the impulse response as a function of the iterations of LSQR, all for the preconditioned form. The top panel corresponds to 3 iterations (overregularized), the middle panel to 5 iterations (optimal), and the bottom panel to 20 iterations (underregularized). The left column shows cross section corresponding to source-receiver pairs at (1500 m, 580 m), (180 m, 180 m), and (1500 m, 180 m), from left to right. The blue line shows the true impulse response and the orange line shows the reconstructed impulse response. The right column shows the reconstructed impulse response for a source at 1500 m, the true impulse response, and a cross section for a source-receiver pair at (1500 m, 1500 m).

5.4.5 Parameter selection methods

In order to make the regularization algorithm useful in practice we have to provide a parameter selection rule. There are two types of parameter selection rules: methods that require an estimate of the noise level and methods that do not. The latter are sometimes referred to as heuristic methods. Here, noise is identically, independently distributed white noise. Such noise is not present in our data. We are dealing with noise, but the noise comes from modeling errors. Therefore, we have to rely on heuristic parameter selection methods to give an estimate of the regularization parameter. Parameter selection rules are designed for the preconditioned system and the Tikhonov system, but not for the KKT system. The Tikhonov system has to be modified, by defining the operator $\begin{bmatrix} Q \\ L_{ST} \end{bmatrix}$ and the data $\begin{bmatrix} \mathbf{P} \\ 0 \end{bmatrix}$. We will use the Lanczos

	1:1 sampling		1:4 sampling	
	Relative error	λ_{opt}	Relative error	λ_{opt}
Preconditioned	0.08	$1.1 \cdot 10^{-5}$	0.12	$3.1 \cdot 10^{-6}$
Tikhonov	0.08	$1.1 \cdot 10^{-5}$	0.12	$3.1 \cdot 10^{-6}$
KKT	0.08	$1 \cdot 10^{-5}$	0.12	$2.6 \cdot 10^{-6}$

Table 5.2: Reconstruction for three different optimization strategies using additional Tikhonov regularization.

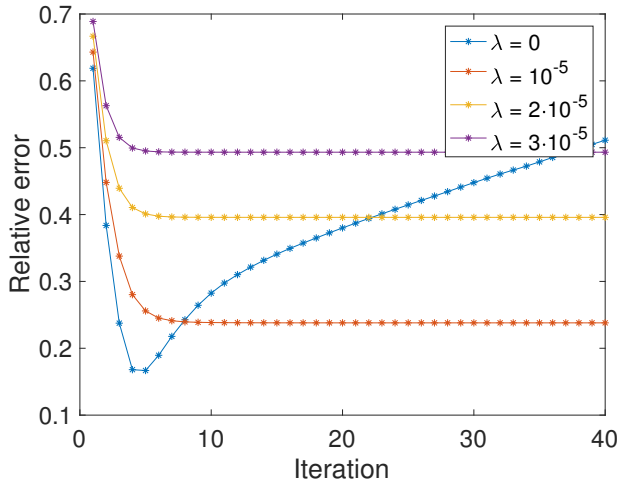


Figure 5.14: Reconstruction for different λ .

process to obtain a low dimensional subspace that is invariant with respect to λ and which allows us to evaluate the parameter selection methods efficiently. We briefly describe some heuristic parameter selection methods. For clarity of presentation we change notation, and now assume that we are solving a system

$$\mathbf{g}_\lambda := \min_{\mathbf{g}} \|\mathbf{Q}\mathbf{g} - \mathbf{p}\|_2^2 + \lambda \|\mathbf{g}\|_2^2.$$

We define

$$\mathbf{r}_\lambda := \|\mathbf{Q}\mathbf{g}_\lambda - \mathbf{p}\|_2.$$

Reginska's rule

Reginska's rule [99] is a variant of the L-curve [53], also known as the Pareto curve. It estimates the optimal λ as the minimizer of

$$\lambda_{\text{Reginska}} = \min_{\lambda} \|\mathbf{g}_\lambda\|^2 \|\mathbf{r}_\lambda\|^2.$$

We choose Reginska's rule over the L-curve because it is easier to evaluate. A relation between the λ estimated by Reginska's rule and the λ estimated by the L-curve can be found in [99].

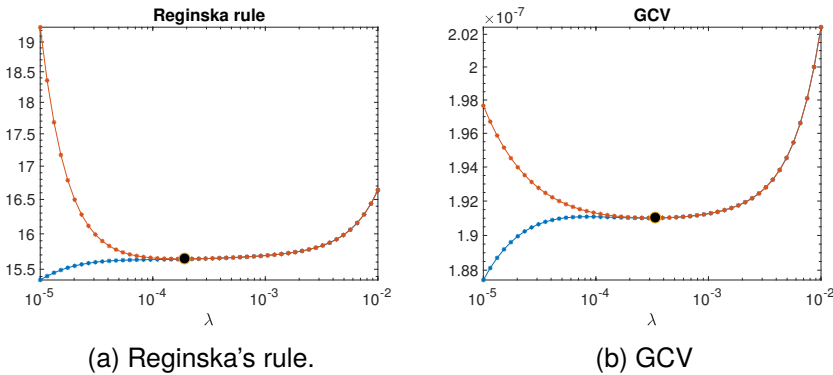


Figure 5.15: Lower and upper bounds for Reginska's rule and GCV for a standard problem. The black dot indicates the minimizer which is an estimate for λ .

GCV

The Generalized Cross Validation [122], [39] is a parameter selection method that estimates the optimal λ as the minimizer of

$$\lambda_{\text{GCV}} = \min_{\lambda} \frac{\|\mathbf{r}_{\lambda}\|^2}{\text{trace}(Q(Q^T Q + \lambda I)^{-1} Q^T)}.$$

The denominator of the GCV can be seen as a measure for the degrees of freedom of the system. If A is large it is costly to evaluate the trace. Therefore, it has been proposed in [39] to use a randomized trace estimator instead. The trace is estimated by

$$\mathbf{u}^T Q(Q^T Q + \lambda I)^{-1} Q^T \mathbf{u},$$

where \mathbf{u} is a vector whose entries are drawn from the Rademacher distribution. For more information on randomized trace estimation we refer to [68].

Lower and upper bounds

Using the Lanczos bidiagonalization process, which is also the basis for LSQR, we can obtain lower and upper bounds for these parameter selection methods. The upper and lower bounds are calculated using a low dimensional approximation to Q , which makes them cheap to evaluate. For details we refer to [43, 44, 21, 23]. The difference between the upper and lower bound indicates how close we are to the true value of the parameter selection method. To show what the lower and upper bounds should look like we show them for a standard problem in figure (5.15). We can see from figure (5.16) that the parameter selection methods fail to give an approximation to the regularization parameter. Although Tikhonov regularization does improve the accuracy of the reconstruction, we can not estimate the correct regularization parameter.

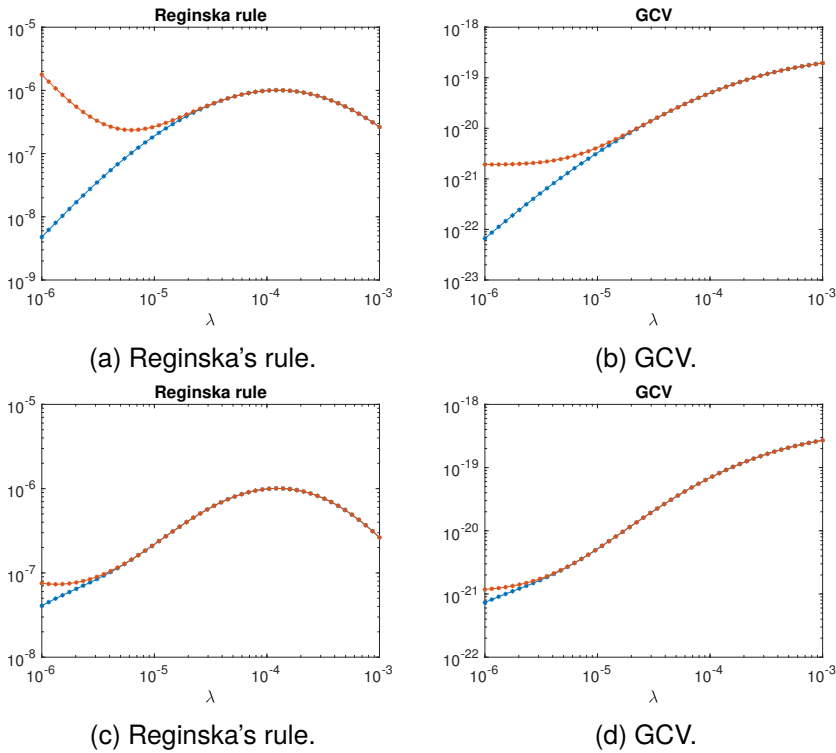


Figure 5.16: Both parameter selection methods fail to have a minimizer estimating the optimal λ for the 1:4 source sampling. The red line shows the upper bound and the blue line shows the lower bound. The top row is for the Tikhonov system and the bottom row for the preconditioned system.

5.5 Conclusion, discussion and outlook


In this chapter we have discussed deconvolving two wavefields to obtain the impulse response in the context of redatuming. We have discussed two constraints that the impulse response has to satisfy and we have shown that they are associated with orthogonal projection operators and a related penalty operator. We have shown three different optimization methods that incorporate the constraints and have shown that they are equivalent in a certain sense. Incorporating the constraints as a penalty using a generalized Tikhonov approach should be avoided, as it is inferior to the preconditioned system in terms of computational time and simplicity. The KKT system was superior in our numerical experiments. However, it is computationally more expensive and it may be difficult or impossible to incorporate additional regularization. Lastly, we have shown that the constraints for the impulse response do not have a stabilizing effect on the solution and that additional regularization is necessary.

We have shown that Tikhonov regularization can be used to improve the


reconstruction. However, we have seen that the parameter selection methods Reginska's rule and GCV are not able to estimate the optimal λ . Moreover, we have to use heuristic parameter selection methods because the noise level is not known and, perhaps more importantly, the noise consists of modeling errors and is not Gaussian and independently and identically distributed.

To solve the MDD problem we have applied standard techniques and theory from the inverse problem literature. The theory is developed for linear systems where the data have been generated by a known forward model. For the MDD problem we deal with two datasets in the form of wavefields that are related by a convolution with the impulse response. Now, both datasets contain noise and a strict model and data separation is artificial. An approach like Total Least Squares (TLS) [115], where both model and data are assumed to be noisy, seems more natural. Specifically, Restricted Total Least Squares [116], where equality constraints can be taken into account, seems the most natural approach for the MDD problem. However, the solution to this problem is given by the Restricted Singular Value Decomposition, which can not be computed for the large matrices arising in MDD problems.

Finally, we could add a regularization filter that filters out the non-recoverable modes. As shown in our example, see figure (5.5), the support of \hat{q} determines how much of \hat{g} can be recovered. We can stabilize the reconstruction by restricting the support of \hat{g} to the support of \hat{q} . Anything outside of this support can not be recovered and can be considered as noise.



Relaxed regularization for linear inverse problems

**Abstract**

We consider regularized least-squares problems of the form $\min_x \frac{1}{2} \|Ax - b\|_2^2 + \mathcal{R}(Lx)$. Recently, Zheng et al. [130] proposed an algorithm called Sparse Relaxed Regularized Regression (SR3) that employs a splitting strategy by introducing an auxiliary variable y and solves $\min_{x,y} \frac{1}{2} \|Ax - b\|_2^2 + \frac{\kappa}{2} \|Lx - y\|_2^2 + \mathcal{R}(x)$. By minimizing out the variable x , we obtain an equivalent optimization problem $\min_y \frac{1}{2} \|F_\kappa y - g_\kappa\|_2^2 + \mathcal{R}(y)$. In our work, we view the SR3 method as a way to approximately solve the regularized problem. We analyze the conditioning of the relaxed problem in general and give an expression for the SVD of F_κ as a function of κ . Furthermore, we relate the Pareto curve of the original problem to the relaxed problem and we quantify the error incurred by relaxation in terms of κ . Finally, we propose an efficient iterative method for solving the relaxed problem with inexact inner iterations. Numerical examples illustrate the approach.

This chapter is partially based on the following publication:

N.A. Luiken and T. van Leeuwen. Relaxed regularization for linear inverse problems. *SIAM J. Sci. Comp.*, Accepted for publication.

6.1 Introduction

Inverse problems are problems where a certain quantity of interest has to be determined from indirect measurements. In medicine, well-known examples include MRI [131], CT [63], and ultrasound imaging [16] where the objective is to obtain images of the interior of the human body. In the geosciences, inverse problems arise in seismic exploration and seismology [120], where the interest lies in exploring the elastic properties of the different layers of our planet. Other examples include tomography [6, 91, 15], radar imaging [17], remote sensing [109, 93], astrophysics [106], and more recently, machine learning [45].

Inverse problems are challenging for a number of reasons. There may be limited data available, or the data may be corrupted by noise. The datasets are generally very large, and the underlying model is generally not well-defined for retrieving the quantity of interest. Therefore, inverse problems often have to be *regularized*, meaning prior information has to be added. They can be posed in the following way:

$$\min_x \frac{1}{2} \|Ax - b\|_2^2 + \mathcal{R}(Lx), \quad (6.1)$$

where $A \in \mathbb{R}^{m \times n}$ is the linear forward operator, $\mathcal{R}(\cdot)$ is the regularization term and $L \in \mathbb{R}^{p \times n}$ the regularization operator. The latter two encode the prior information about x . In our work, we focus on $\mathcal{R}(\cdot) = \lambda \|\cdot\|_p^p$, or, equivalently, $\mathcal{R}(\cdot) = \delta_{\|\cdot\|_p \leq \tau}(\cdot)$, which is the indicator function of the set $\|\cdot\|_p \leq \tau$ ¹. By equivalent we mean that for every τ there is a λ such that the solutions of the two problems coincide [5]. A direct solution to the problem above is generally not possible, either because a closed-form solution does not exist, or because evaluating the direct solution is too computationally expensive. Therefore, we have to resort to iterative methods to solve the problem, with most algorithms being designed for specific choices of p and L .

Traditionally, $p = 2$, called Tikhonov regularization, is a popular choice, because the objective function is differentiable and allows for a closed-form expression of the solution of eq. (6.1) in terms of A, L and λ . For this class of problems, Krylov based algorithms have been proven very effective [22, 23, 38, 44, 57, 66, 67, 72, 73, 133]. These methods generally exploit the fact that a closed-form solution exists by constructing a low dimensional subspace from which an approximate solution is extracted.

The choice $p = 1$ has gained popularity in recent years because it gives sparse solutions while still yielding a convex objective. Sparsity is important in a number of applications, like compressed sensing [25], seismic imaging [62], image restoration [100], and tomography [58]. However, the objective is no longer differentiable and the aforementioned Krylov methods do not apply. If $L = I$, a proximal gradient method (sometimes referred to as Iterative Soft Thresholding – ISTA) [28] can be

¹In our work we use p for both the size of the matrix L as the norm of the regularizer. The meaning of p is always clear from the context.

applied, iteratively updating the solution via

$$x_{k+1} = \text{prox}_{\alpha\lambda\|\cdot\|_1}(x_k - \alpha A^T(Ax_k - b)),$$

where $\alpha \in (0, \|A\|_2^{-2})$ is the stepsize and the proximal operator is the soft thresholding operator, which can be efficiently evaluated. Generally, ISTA achieves a sub-linear rate of convergence of $\mathcal{O}(1/k)$ (unless $m \geq n$ and A has full rank, in which case we have a linear rate of convergence). FISTA (Fast Iterative Soft Thresholding Algorithm) [10] is a faster version of ISTA that generally achieves a sublinear rate of $\mathcal{O}(1/k^2)$.

If $L = I$ the optimization problem is said to be in standard-form and for any other L the algorithm is in general form. If L is full-rank and has no nullspace, the optimization problem can be put into standard-form via the change of variables $y = Lx$. Instead of the matrix A , we get AL^\dagger . In such cases we can apply the (F)ISTA method directly at the expense of having to evaluate L^\dagger . In some applications, we have $L^\dagger = L^T$ (e.g., when L is a tight frame). If L has a non-trivial nullspace the algorithm can still be put in standard-form by the standard-form transformation [34, 58], but this is nontrivial, because the nullspace has to be accounted for.

If $L \neq I$, and we cannot easily transform the problem to standard form, the proximal operator is no longer easy to evaluate in general and FISTA may no longer be attractive. An example of this class of problems is Total Variation (TV) regularization, where L is the discretization of the gradient, which gives blocky solutions. A popular algorithm for this class of problems is the Alternating Direction Method of Multipliers, ADMM [18]. ADMM solves eq. (6.1) by forming the *augmented Lagrangian*

$$\min_{x,y} \max_z \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|y\|_p^p + z^T (Lx - y) + \frac{\rho}{2} \|Lx - y\|_2^2,$$

and alternately minimizing over the variables x and y , and the Lagrange multiplier z . The strength of ADMM is that it can closely approximate the solution of any convex sparse optimization problem. However, convergence can be slow [18].

If $p < 1$, the emphasis on sparsity of the solution is stronger than for the case $p = 1$. However, the objective function is no longer convex which makes it more difficult to solve.

Recently, a unifying algorithm was proposed that allows the efficient approximation of the solution of any problem of the form eq. (6.1), called Sparse Relaxed Regularized Regression (SR3) [130]. This algorithm makes use of a splitting strategy by introducing an auxiliary variable y and yields:

$$\min_{x,y} \frac{1}{2} \|Ax - b\|_2^2 + \frac{\kappa}{2} \|Lx - y\|_2^2 + \mathcal{R}(y). \quad (6.2)$$

By minimizing out x , we obtain a new optimization problem of the form:

$$\bar{y}_\kappa = \operatorname{argmin}_y \frac{1}{2} \|F_\kappa y - g_\kappa\|_2^2 + \mathcal{R}(y), \quad (6.3)$$

where $F_\kappa = \begin{pmatrix} \kappa^{1/2} (I - \kappa L H_\kappa^{-1} L^T) \\ \kappa A H_\kappa^{-1} L^T \end{pmatrix}$ and $g_\kappa = \begin{pmatrix} \kappa^{1/2} L H_\kappa^{-1} A^T b \\ b - A H_\kappa^{-1} A^T b \end{pmatrix}$, $H_\kappa = A^T A + \kappa L^T L$. The solution to (6.2) is then given by

$$\bar{x}_\kappa = H_\kappa^{-1} (\kappa L^T \bar{y}_\kappa + A^T b). \quad (6.4)$$

This solution is then used as an approximation of the solution of (6.1). In [130] the particular case with $L^T L = I$ is analyzed. Using the SVD of A , the singular values of F_κ were calculated, showing a relation between the condition number of F_κ and A depending on κ . In short, the result shows that a small κ improves the conditioning of F_κ and as $\kappa \rightarrow \infty$ the condition numbers are the same, because the original optimization problem is obtained.

For the implementation of SR3, it is not necessary to form the operator F_κ , as was shown in [130]. The authors propose the following algorithm for solving the relaxed problem

$$x_{k+1} \leftarrow (A^T A + \kappa L^T L)^{-1} (A^T b + \kappa L^T y_k) \quad (6.5)$$

$$y_{k+1} \leftarrow \operatorname{prox}_{\alpha \mathcal{R}}(y_k - \alpha \kappa (y_k - L x_{k+1})), \quad (6.6)$$

which for the particular choice $\alpha = 1/\kappa$ simplifies to

$$x_{k+1} \leftarrow (A^T A + \kappa L^T L)^{-1} (A^T b + \kappa L^T y_k) \quad (6.7)$$

$$y_{k+1} \leftarrow \operatorname{prox}_{1/\kappa \mathcal{R}}(L x_{k+1}). \quad (6.8)$$

This method has several advantages when applied to solving inverse problems that we highlight in the examples below.

6.1.1 Motivating examples

Below we show some typical examples encountered in various areas of science to which SR3 can be applied. The problems we tackle are of the form

$$\min_x \frac{1}{2} \|Ax - b\|_2^2 \quad \text{s.t.} \quad \|Lx\|_1 \leq \tau. \quad (6.9)$$

The main tasks are to solve this for a given value of τ and to find an appropriate value of τ . The latter is achieved by picking the corner of the Pareto curve (sometimes called the L-curve) $\phi(\tau) = \min_{\|x\|_p \leq \tau} \|Ax - b\|_2$. Comparing a proximal gradient method to SR3, we show the residual as a function of τ , the optimal reconstruction, and the convergence history in terms of the primal-dual gap. These examples show two favourable aspects of SR3 over the conventional proximal gradient method: *i*) SR3 converges (much) faster for any fixed value of τ and *ii*) the corners of both Pareto-curves coincide, allowing us to effectively use SR3 to estimate τ .

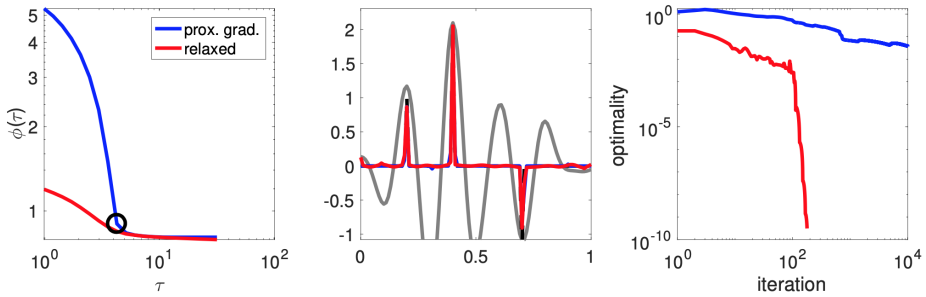


Figure 6.1: Spiky deconvolution example. The left figure shows the Pareto curve, the middle figure shows the solution and the right figure shows the primal-dual gap as a function of the number of iterations. The grey line in the middle figure shows the minimum norm solution.

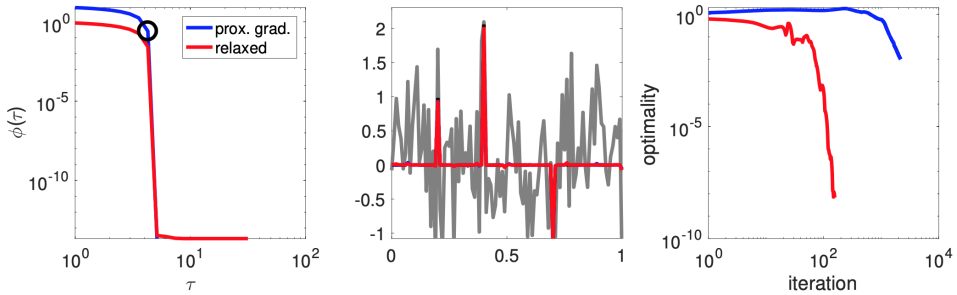


Figure 6.2: Compressed sensing. A signal is reconstructed from very few samples, which requires sparse regularization. The left figure shows the Pareto curve, the middle figure shows the solution and the right figure shows the primal-dual gap as a function of the number of iterations. The grey line in the middle figure shows the minimum norm solution.

Spiky deconvolution ($m = n$, $L = I$)

Consider a deconvolution problem where A is a Toeplitz-matrix that convolves the input with a bandlimited function;

$$a_{ij} = w(t_i - t_j),$$

where $w(t) = (1 - (t/\sigma)^2)e^{-(t/\sigma)^2}$ and $t_i = i \cdot h$. We take $n = 101$, $h = 1/n$ and $\sigma = 0.05$. The results are shown infig. 6.1.

Compressed sensing ($m < n$, $L = I$)

Here, the goal is to recover a sparse signal from compressive samples. The forward operator is a random matrix with i.i.d. normally distributed entries. We take $n = 101$ and $m = 20$. The results are shown in fig. 6.2.

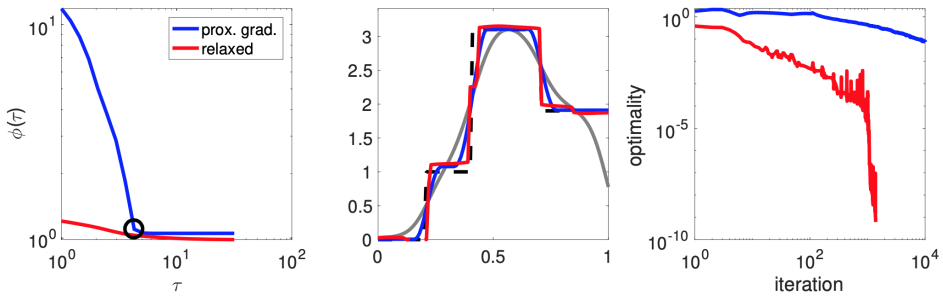


Figure 6.3: Total variation example. Here, the solution has a blocky structure. The left figure shows the Pareto curve, the middle figure shows the solution and the right figure shows the primal-dual gap as a function of the number of iterations. The grey line in the middle figure shows the minimum norm solution.

Total variation ($m = n$, $L = D$)

Consider a deconvolution problem where A is a Toeplitz-matrix that convolves the input with a bandlimited function;

$$a_{ij} = w(t_i - t_j),$$

where $w(t) = e^{-(t/\sigma)^2}$ and $t_i = i \cdot h$. L is a finite-difference discretization of the first-order derivative with Neumann boundary conditions. We take $n = 101$, $h = 1/n$ and $\sigma = 0.05$. The results are shown in fig. 6.3.

6.1.2 Contributions

In this chapter we set out to further analyze the SR3 method proposed in [130] and analyze in detail the observations made in the above examples. Our contributions are:

Conditioning of F_κ for general L . We extend the analysis of [130] and derive the SVD of F_κ for general L . We show how the singular values and the condition number of F_κ are related to the generalized singular values of (A, L) . As a by-product, we show that SR3 implicitly makes a standard-form transformation [34] of eq. (6.1).

Approximation of the Pareto-curve. We show that the Pareto curve corresponding to the relaxed problem (6.2) always underestimates the Pareto curve of the original problem (6.1) and that the error is of order $\mathcal{O}(\kappa^{-2})$. A by-product of this result is a better understanding of the Pareto curve for general p and an intuitive explanation of the observation that the corners of the relaxed original Pareto curves coincide.

Inexact solves. We propose an inexact inner-outer iterative version of the SR3 algorithm where the regularized least-squares problem eq. (6.7) is solved approximately using a Krylov-subspace method. In particular, we propose an automated adaptive stopping criterion for the inner iterations.

6.1.3 Outline

In section 6.2 we analyze the operator F_κ . We derive the SVD of F_κ and analyse the limiting cases $\kappa \rightarrow \infty$ and $\kappa \rightarrow 0$. Our main results are a characterization of the singular values of F_κ and showing that SR3 implicitly applies a standard-form transformation. In section 6.3, we relate the Pareto curve of SR3 to the Pareto curve of the original problem and derive an error bound in terms of κ . Next, section 6.4 is concerned with the implementation of SR3. We propose two ingredients that make SR3 suitable for large-scale applications. In section 6.5, we conduct our numerical experiments and verify the theoretical results from section 6.2. Moreover, we numerically investigate the influence of κ on the convergence rate. Finally, in section 6.6, we draw our conclusions.

6.2 Analysis of SR3

In this section we analyze some of the properties of the operator F_κ . We will characterize the singular values of F_κ for general L and analyse the limits $\kappa \rightarrow 0$ and $\kappa \rightarrow \infty$. First, we will treat some preliminaries needed for understanding what happens in the limit $\kappa \rightarrow \infty$.

6.2.1 The Generalized Singular Value Decomposition

The central tool in our analysis is the Generalized Singular Value Decomposition (GSVD) of (A, L) . The definition of the GSVD depends on the size of the matrices and the dimensions of the matrices relative to each other. We use the definitions for the case $A \in \mathbb{R}^{m \times n}$ and $L \in \mathbb{R}^{p \times n}$ where $m \geq n$, $p < n$ or $m < n$, $p > n$ because this corresponds to the examples we use in our experiments.

Definition 2 (GSVD). *Let $A \in \mathbb{R}^{m \times n}$ and $L \in \mathbb{R}^{p \times n}$. The Generalized Singular Value Decomposition (GSVD) of (A, L) is given by $A = U\Sigma X$, $L = V\Gamma X$, where*

$$\Sigma = \begin{bmatrix} \Sigma_p & 0 \\ 0 & I_{n-p} \\ 0 & 0 \end{bmatrix}, \quad \Gamma = [\Gamma_p \quad 0] \quad \text{for } m \geq n, p \leq n,$$

and

$$\Sigma = [0 \quad \Sigma_m], \quad \Gamma = \begin{bmatrix} I_{n-m} & 0 \\ 0 & \Gamma_m \\ 0 & 0 \end{bmatrix} \quad \text{for } m < n, p > n.$$

The matrices Σ_r and Γ_r (where $r = p$ or $r = m$) are $r \times r$ diagonal matrices satisfying $\Sigma_r^T \Sigma_r + \Gamma_r^T \Gamma_r = I_r$, X is invertible and U and V are orthonormal. Moreover, we have the following ordering of the diagonal elements σ_i of Σ and γ_i of Γ :

$$\begin{aligned} 0 &\leq \gamma_r \leq \dots \leq \gamma_1 \leq 1, \\ 0 &\leq \sigma_1 \leq \dots \leq \sigma_r \leq 1. \end{aligned}$$

The decomposition of A and L in the GSVD share similar properties to the SVD. The number of nonzero entries of Σ and Γ give the rank of A and L respectively. If r_A

is the rank of A and r_L is the rank of L then the last $r - r_A$ columns, corresponding to Σ_r , of U form a basis for the range of A and the first r_L columns, corresponding to Γ_r , of V form a basis for the range of L . The first $r - r_A$ columns, corresponding to Σ_r , of X^{-1} form a basis for the nullspace of A and the last $r - r_L$ columns, corresponding to Γ_r , of X^{-1} form a basis for the nullspace of L .

6.2.2 Standard-form transformation

The standard-form transformation, see e.g. [34, 55], makes a substitution $y = Lx$ such that $x = x_{\mathcal{M}} + x_{\mathcal{N}}$, where

$$\bar{x}_{\mathcal{M}} = L_A^\dagger \bar{y}, \quad \bar{y} = \operatorname{argmin}_y \frac{1}{2} \|AL_A^\dagger y - b\|_2^2 + \mathcal{R}(y), \quad L_A^\dagger = (I - (A(I - L^\dagger L))^\dagger A) L^\dagger. \quad (6.10)$$

and

$$\bar{x}_{\mathcal{N}} = (A(I - L^\dagger L))^\dagger b. \quad (6.11)$$

The operator L_A^\dagger is called the *A-weighted pseudo-inverse*. The transformation splits the solution into two parts: one part in the range of L , $L_A^\dagger y$, and one part in the nullspace of L , $x_{\mathcal{N}}$. The operator L_A^\dagger makes the two parts A -orthogonal. The parts $L_A^\dagger y$ and $x_{\mathcal{N}}$ are then obtained by two independent optimization problems. If L is invertible $L_A^\dagger = L^{-1}$ and if $p > n$ and L has full rank we have $L_A^\dagger = L^\dagger$. Hence, if $L^T L = I$, the standard-form is achieved by simply applying L^T .

In terms of the GSVD of (A, L) , the standard-form transformation has a much simpler form. The operator L_A^\dagger can be written in terms of the GSVD as

$$L_A^\dagger = X^{-1} \Gamma^\dagger V^T,$$

and hence eq. (6.10) can be written as

$$\bar{x}_{\mathcal{M}} = X^{-1} \Gamma^\dagger V^T \bar{y}, \quad \bar{y} = \operatorname{argmin}_y \frac{1}{2} \|U \Sigma \Gamma^\dagger V^T y - b\|_2^2 + \mathcal{R}(y). \quad (6.12)$$

Similarly, eq. (6.11) can be written in terms of the GSVD as

$$\bar{x}_{\mathcal{N}} = X^{-1} \begin{bmatrix} 0 & 0 \\ 0 & I_{p-r_L} \end{bmatrix} U^T b. \quad (6.13)$$

6.2.3 The SVD of F_κ

In this section we derive the SVD of F_κ in terms of the GSVD of (A, L) .

Theorem 5. Let $F_\kappa = Y \Lambda Z^T$ be the SVD of F_κ . Let the GSVD of $\begin{bmatrix} A \\ L \end{bmatrix} = \begin{bmatrix} U \Sigma \\ V \Gamma \end{bmatrix} X$.

Then

$$\begin{aligned} Y &= \begin{bmatrix} \kappa^{1/2} V \tilde{\Sigma}_{\kappa, I}^{1/2} & \kappa V \tilde{\Sigma}_{\kappa, I}^{-1/2} \Gamma (\Sigma^T \Sigma + \kappa \Gamma^T \Gamma)^{-1} \Sigma^T \\ \kappa U \Sigma (\Sigma^T \Sigma + \kappa \Gamma^T \Gamma)^{-1} \Gamma^T \tilde{\Sigma}_{\kappa, I}^{-1/2} & -\kappa^{-1/2} U \tilde{\Sigma}_{m, \kappa}^{1/2} \end{bmatrix} \\ \Lambda &= \begin{bmatrix} \tilde{\Sigma}_\kappa^{1/2} \\ 0 \end{bmatrix} \\ Z &= V, \end{aligned}$$

where $\tilde{\Sigma}_\kappa = \kappa (I_p - \kappa\Gamma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Gamma^T)$, $\tilde{\Sigma}_{\kappa,m} = \begin{bmatrix} \tilde{\Sigma}_\kappa & 0 \\ 0 & I_{m-p} \end{bmatrix}$ if $m \geq n \geq p$ and $\tilde{\Sigma}_{\kappa,m} = \tilde{\Sigma}_{\kappa,I}$ if $m < n \leq p$, and the square root denotes the entry wise square root. If $p > n$ the diagonal matrix $\tilde{\Sigma}_\kappa$ will have zeros on the diagonal. We denote $\tilde{\Sigma}_{\kappa,I}$ to be the matrix $\tilde{\Sigma}_\kappa$ where the zeros have been replaced by ones.

Proof. Using the GSVD of (A, L) we have $H_\kappa^{-1} = X^{-1}(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}X^{-T}$ and hence $LH_\kappa^{-1}L^T = V\Gamma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Gamma^TV^T$. Given the fact that V is orthonormal and $\Gamma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Gamma^T$ is a diagonal matrix the above expression is the SVD of $LH_\kappa^{-1}L^T$ and we obtain the expressions for Λ and Z . To obtain Y , we first partition $Y = \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{bmatrix}$. We have

$$\begin{aligned} F_\kappa F_\kappa^T &= Y\Lambda\Lambda^T Y^T \\ &\iff \begin{bmatrix} \kappa(I - \kappa LH_\kappa^{-1}L^T)^2 & \kappa\sqrt{\kappa}(I - \kappa LH_\kappa^{-1}L^T)LH_\kappa^{-1}A^T \\ \kappa\sqrt{\kappa}AH_\kappa^{-1}L^T(I - \kappa LH_\kappa^{-1}L^T) & \kappa^2 AH_\kappa^{-1}LL^T H_\kappa^{-1}A^T \end{bmatrix} \\ &= \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{bmatrix} \begin{bmatrix} \tilde{\Sigma}_\kappa & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} Y_{11}^T & Y_{21}^T \\ Y_{12}^T & Y_{22}^T \end{bmatrix} = \begin{bmatrix} Y_{11}\tilde{\Sigma}_\kappa Y_{11}^T & Y_{11}\tilde{\Sigma}_\kappa Y_{21}^T \\ Y_{21}\tilde{\Sigma}_\kappa Y_{11}^T & Y_{21}\tilde{\Sigma}_\kappa Y_{21}^T \end{bmatrix}. \end{aligned}$$

Plugging in the GSVD gives

$$F_\kappa F_\kappa^T = \begin{bmatrix} \kappa^{-1}V\tilde{\Sigma}_\kappa^2 V^T & \sqrt{\kappa}V\tilde{\Sigma}_\kappa\Gamma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Sigma^T U^T \\ \sqrt{\kappa}U\Sigma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Gamma^T\tilde{\Sigma}_\kappa V^T & \kappa^2 U\Sigma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Gamma^T\Gamma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Sigma^T U^T \end{bmatrix}.$$

Solving for Y_{11} gives:

$$Y_{11} = \kappa^{-1/2}V\tilde{\Sigma}_{\kappa,I}^{1/2}.$$

Using this in the upper right part gives:

$$Y_{21} = \kappa U\Sigma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Gamma^T\tilde{\Sigma}_{\kappa,I}^{-1/2}.$$

To solve for Y_{12} and Y_{22} , we use

$$YY^T = \begin{bmatrix} Y_{11}Y_{11}^T + Y_{12}Y_{12}^T & Y_{11}Y_{21}^T + Y_{12}Y_{22}^T \\ Y_{21}Y_{11}^T + Y_{22}Y_{12}^T & Y_{21}Y_{21}^T + Y_{22}Y_{22}^T \end{bmatrix} = \begin{bmatrix} I_p & 0 \\ 0 & I_m \end{bmatrix}.$$

The upper left part yields

$$Y_{12} = \kappa V\tilde{\Sigma}_{\kappa,I}^{-1/2}\Gamma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Sigma^T.$$

The upper right part yields

$$Y_{22} = -\kappa^{-1/2}U\tilde{\Sigma}_{\kappa,m}^{1/2}.$$

□

Note that the singular values are ordered in ascending order. We have the following corollary.

Corollary 1. *If $m \geq n$ and $p < n$ the singular values of F_κ are given by*

$$\psi_i(F_\kappa) = \sqrt{\frac{\sigma_{n-i+1}^2}{\sigma_{n-i+1}^2/\kappa + \gamma_{n-i+1}^2}}.$$

If $m < n$ and $p > n$ the singular values of F_κ are given by

$$\psi_i(F_\kappa) = \begin{cases} \sqrt{\kappa} & \text{if } i \leq p - r_L \\ \sqrt{\frac{\sigma_{m-i+1}^2}{\sigma_{m-i+1}^2/\kappa + \gamma_{m-i+1}^2}} & \text{if } p - r_L < i \leq p - r_L + r_A \\ 0 & \text{if } i > p - r_L + r_A \end{cases}$$

The question arises whether there is a direct relation between the singular values of A and the σ_i . The answer is no, but we do, however, have the following result from [52]:

Theorem 6 ([52, Thm. 2.4]). *Let $\psi_i(A)$ and $\psi_i(L)$ denote the singular values of A and L respectively and let σ_i and γ_i denote the nonzero entries of the matrices Σ and Γ respectively. Then for all $\sigma_i, \gamma_i \neq 0$*

$$\begin{aligned} \left\| \begin{bmatrix} A \\ L \end{bmatrix}^\dagger \right\|_2^{-1} &\leq \frac{\psi_{r-i+1}(A)}{\sigma_i} \leq \left\| \begin{bmatrix} A \\ L \end{bmatrix} \right\|_2, \\ \left\| \begin{bmatrix} A \\ L \end{bmatrix}^\dagger \right\|_2^{-1} &\leq \frac{\psi_i(L)}{\gamma_i} \leq \left\| \begin{bmatrix} A \\ L \end{bmatrix} \right\|_2. \end{aligned}$$

Remark 1. *This result shows that, if the operator A has quickly decaying singular values, the σ_i will have the same behavior, see also [55, p. 24]. This is an important result because it shows how the ill-conditioning of A transfers over to F_κ . Note that if*

$\sigma_i \approx 0$ we have $\gamma_i \approx 1$ and the singular values of $\psi_i(F_\kappa) = \sqrt{\frac{\sigma_{r-i+1}}{\sigma_{r-i+1}/\kappa + \gamma_{r-i+1}}} \approx \sqrt{\frac{\sigma_{r-i+1}}{\sigma_{r-i+1}/\kappa + 1}} \approx 0$. Hence, if the operator A is severely ill-posed, this ill-posedness is inherited by the operator F_κ .

6.2.4 Limiting cases

The limit $\kappa \rightarrow \infty$ if $p < n$

If $L = I$ the limit $\kappa \rightarrow \infty$ yields the original optimization problem. However, if $L \neq I$, it is not immediately clear what happens in the limit $\kappa \rightarrow \infty$ due to the presence of the operator L . In this section we derive this limit using the GSVD of (A, L) . We will show that in the limit $\kappa \rightarrow \infty$ SR3 applies a standard-form transformation. We will proceed as follows. Recall that the variable x in SR3 is given by

$$\bar{x}_\kappa = H_\kappa^{-1} (\kappa L^T \bar{y}_\kappa + A^T b) = \kappa H_\kappa^{-1} L^T \bar{y}_\kappa + H_\kappa^{-1} A^T b := x_1 + x_2, \quad (6.14)$$

consisting of the two parts x_1 and x_2 . We will now show that, in the limit $\kappa \rightarrow \infty$, SR3 applies a standard-form transformation, by showing that x_1 and x_2 defined in eq. (6.14) satisfy

$$x_1 = \bar{x}_{\mathcal{M}}, \quad x_2 = \bar{x}_{\mathcal{N}}, \quad (6.15)$$

where $x_{\mathcal{M}}$ and $x_{\mathcal{N}}$ are determined by the standard-form transformation, given by eq. (6.12) and eq. (6.13) respectively.

Given the GSVD of (A, L) , the matrix F_κ and the vector g_κ are given by

$$F_\kappa = \begin{bmatrix} \sqrt{\kappa}V \left(I_p - \kappa\Gamma (\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1} \Gamma^T \right) V^T \\ \kappa U \Sigma (\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1} \Gamma^T V^T \end{bmatrix}, \quad (6.16)$$

and

$$g_\kappa = \begin{bmatrix} \sqrt{\kappa}V\Gamma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1}\Sigma^T U^T b \\ U \left(I_m - \Sigma (\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1} \Sigma^T \right) U^T b \end{bmatrix}. \quad (6.17)$$

As $\kappa \rightarrow \infty$ we have

$$F_\kappa \rightarrow \begin{bmatrix} 0 \\ U\Sigma\Gamma^\dagger V^T \end{bmatrix} \quad \text{and} \quad g_\kappa \rightarrow \begin{bmatrix} 0 \\ b \end{bmatrix}.$$

Hence, as $\kappa \rightarrow \infty$, we obtain

$$\bar{y}_\kappa = \underset{y}{\operatorname{argmin}} \frac{1}{2} \|U\Sigma\Gamma^\dagger V^T y - b\|_2^2 + \mathcal{R}(y). \quad (6.18)$$

Using the GSVD, we have

$$H_\kappa^{-1} = X^{-1} (\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1} X^{-T},$$

and hence as $\kappa \rightarrow \infty$ we have

$$(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1} \rightarrow \begin{bmatrix} 0 & 0 \\ 0 & I_{p-r_L} \end{bmatrix}.$$

Hence,

$$H_\kappa^{-1} \rightarrow X^{-1} \begin{bmatrix} 0 & 0 \\ 0 & I_{p-r_L} \end{bmatrix} X^{-T}. \quad (6.19)$$

Recall that the last columns of X are a basis for the nullspace of L and hence H_κ projects onto the nullspace of L . Using the GSVD of (A, L) we see that

$$\lim_{\kappa \rightarrow \infty} x_1 := \lim_{\kappa \rightarrow \infty} H_\kappa^{-1} A^T b = X^{-1} \begin{bmatrix} 0 & 0 \\ 0 & I_{p-r_L} \end{bmatrix} U^T b,$$

which is equivalent to the nullspace component from (6.13).

We now show that x_1 corresponds to the part in the range of L . We have

$$x_1 := \kappa H_\kappa^{-1} L^T \bar{y} = \kappa X^{-1} (\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)^{-1} \Gamma^T V^T \bar{y}.$$

The elements of the diagonal matrix $\kappa (\Sigma^T \Sigma + \kappa \Gamma^T \Gamma)^{-1} \Gamma^T$ are

$$\begin{cases} \frac{\gamma_i}{\sigma_i^2/\kappa + \gamma_i^2} & \text{if } i \leq r_L, \\ 0 & \text{if } i > r_L, \end{cases}$$

and as $\kappa \rightarrow \infty$

$$\begin{cases} \frac{1}{\gamma_i} & \text{if } i \leq r_L, \\ 0 & \text{if } i > r_L. \end{cases}$$

Hence, as $\kappa \rightarrow \infty$

$$\kappa (\Sigma^T \Sigma + \kappa \Gamma^T \Gamma)^{-1} \Gamma^T \rightarrow \Gamma^\dagger,$$

and thus

$$\kappa H_\kappa^{-1} L^T \rightarrow X^{-1} \Gamma^\dagger V^T = L_A^\dagger.$$

The limit for the component x_1 is now given by

$$\lim_{\kappa \rightarrow \infty} x_1 = X^{-1} \Gamma^\dagger V^T \bar{y}_\kappa = L_A^\dagger \bar{y}_\kappa,$$

where \bar{y}_κ solves

$$\bar{y}_\kappa = \operatorname{argmin}_y \frac{1}{2} \|U \Sigma \Gamma^\dagger V^T y - b\|_2^2 + \mathcal{R}(y),$$

which is equivalent to (6.12).

The limit $\kappa \rightarrow \infty$ if $p > n$

If $p > n$, the limit $\kappa \rightarrow \infty$ is a bit more subtle. For large κ , we have

$$F_\kappa \sim \begin{bmatrix} V \begin{bmatrix} 0_{r_L \times r_L} & 0 \\ 0 & \sqrt{\kappa} I_{p-r_L} \end{bmatrix} V^T \\ U \Sigma \Gamma^\dagger V^T \end{bmatrix}, \quad \text{and } g_\kappa \sim \begin{bmatrix} 0_{p \times 1} \\ b \end{bmatrix} \quad (6.20)$$

Hence, for large κ , SR3 solves a system of the form

$$\begin{bmatrix} \sqrt{\kappa} V_{p-r_L} V_{p-r_L}^T \\ U \Sigma \Gamma^\dagger V^T \end{bmatrix} y = \begin{bmatrix} 0_{p \times 1} \\ b \end{bmatrix},$$

where V_{p-r_L} are the *last* $p - r_L$ columns of V , which means that $V_{p-r_L} V_{p-r_L}^T = \mathcal{P}_{\mathcal{N}(L^T)}$. Because $V_{p-r_L} V_{p-r_L}^T y = 0$, the solution has no parts in $\mathcal{N}(L^T)$, and is restricted to the subspace $\mathcal{R}(L)$. The bottom part of F_κ is equal to the case $p < n$, and hence corresponds to matrix AL_A^\dagger . Let \bar{y}_{std} be the solution to the standard-form transformed system. Then, as $\kappa \rightarrow \infty$, the minimizer \bar{y}_κ of SR3 satisfies

$$\bar{y}_{\text{std}} = \mathcal{P}_{\mathcal{R}(L)} \bar{y}_\kappa. \quad (6.21)$$

However, looking at the original formulation in eq. (6.2), we see that as $\kappa \rightarrow \infty$ we have

$$y = Lx,$$

which means that $y \in \mathcal{R}(L)$. Hence, condition eq. (6.21) is immediately satisfied and the solutions are the same.

6.2.5 The limit $\kappa \rightarrow 0$

The limit $\kappa \rightarrow 0$ is much easier to derive. Recall that

$$\bar{x}_\kappa = H_\kappa^{-1} (A^T b + \kappa L^T \bar{y}_\kappa).$$

As $\kappa \rightarrow 0$ we have $\kappa H_\kappa^{-1} L^T \bar{y}_\kappa \rightarrow 0$ and $H_\kappa \rightarrow (A^T A)^{-1}$. Hence $\lim_{\kappa \rightarrow 0} x_\kappa = (A^T A)^{-1} A^T b$ which is the unregularized minimum norm solution.

6.2.6 Relation to the standard-form transformation

The case $p \leq n$

We have shown that as $\kappa \rightarrow \infty$ SR3 implicitly applies a standard-form transformation and that as $\kappa \rightarrow 0$ the system is unregularized. The question arises what happens for finite $\kappa > 0$. To show what happens, we rewrite the singular values of F_κ as

$$\psi_i(F_\kappa) = \sqrt{\frac{\sigma_{r-i+1}^2}{\sigma_{r-i+1}^2/\kappa + \gamma_{r-i+1}^2}} = \sqrt{\frac{\sigma_{r-i+1}^2/\gamma_{r-i+1}^2}{\frac{\sigma_{r-i+1}^2/\gamma_{r-i+1}^2}{\kappa} + 1}} = \sqrt{\frac{\psi_i^2(AL_A^\dagger)}{\psi_i^2(AL_A^\dagger)/\kappa + 1}}.$$

This is equivalent to equation 9 in [130], where it was shown that if $L^T L = I$,

$$\psi_i(F_\kappa) = \frac{\psi_i^2(A)}{\psi_i^2(A)/\kappa + 1}.$$

This shows that SR3 is applied to the matrix AL_A^\dagger . This leads to the following theorem.

Theorem 7. *Let $p \leq n$. The following diagram commutes.*

$$\begin{array}{ccc} \min_x \frac{1}{2} \|Ax - b\|_2^2 + \lambda \mathcal{R}(Lx) & \xrightarrow{\text{SR3}} & \begin{array}{l} \bar{y}_\kappa = \operatorname{argmin}_y \frac{1}{2} \|F_\kappa y - g_\kappa\|_2^2 + \lambda \mathcal{R}(y) \\ \bar{x}_\kappa = H_\kappa^{-1} (\kappa L^T \bar{y}_\kappa + A^T b) \end{array} \\ \downarrow & & \downarrow \uparrow \\ \begin{array}{l} \bar{z}_\kappa = \operatorname{argmin}_z \frac{1}{2} \|AL_A^\dagger z - b\|_2^2 + \lambda \mathcal{R}(z) \\ \bar{x}_\kappa = L_A^\dagger \bar{z}_\kappa + x_N \end{array} & \xrightarrow{\text{SR3}} & \begin{array}{l} \bar{y}_\kappa = \operatorname{argmin}_y \frac{1}{2} \|F_\kappa y - g_\kappa\|_2^2 + \lambda \mathcal{R}(y) \\ \bar{z}_\kappa = H_\kappa^{-1} (\kappa \bar{y}_\kappa + (AL_A^\dagger)^T b) \\ \bar{x}_\kappa = L_A^\dagger \bar{z}_\kappa + x_N \end{array} \end{array}$$

The case $p > n$

If $p > n$ the situation is different. Recall that the singular values of F_κ are given by

$$\psi_i(F_\kappa) = \begin{cases} \sqrt{\kappa} & \text{if } i \leq p - r_L \\ \sqrt{\frac{\psi_i^2(AL_A^\dagger)}{\psi_i^2(AL_A^\dagger)/\kappa + 1}} & \text{if } p - r_L < i \leq p - r_L + r_A \\ 0 & \text{if } i > p - r_L + r_A \end{cases}$$

The singular values for F_κ when SR3 is applied to AL_A^\dagger are given by

$$\psi_i(F_\kappa) = \begin{cases} \sqrt{\frac{\psi_i^2(AL_A^\dagger)}{\psi_i^2(AL_A^\dagger)/\kappa + 1}} & \text{if } i \leq r_A \\ 0 & \text{if } i > r_A \end{cases}$$

Hence, there are extra singular values $\sqrt{\kappa}$ when SR3 is applied to the general-form system as opposed to the standard-form system. The difference may be seen from the expression eq. (6.16). We have

$$\kappa\Gamma(\Sigma^T\Sigma + \kappa\Gamma^T\Gamma)\Gamma^T = \begin{cases} \begin{bmatrix} I_{n-r_A} & 0 & 0 \\ 0 & \kappa\Gamma_m(\Sigma^T\Sigma + \kappa\Gamma_m^T\Gamma_m)\Gamma_m^T & 0 \\ 0 & 0 & 0 \end{bmatrix} & \text{if } p > n \\ \begin{bmatrix} I_{p-r_A} & 0 \\ 0 & \kappa\Gamma_m(\Sigma^T\Sigma + \kappa\Gamma_m^T\Gamma_m)\Gamma_m^T \end{bmatrix} & \text{if } p \leq n \end{cases}$$

Hence, the top part of F_κ is different. Before we state our theorem let us introduce some notation. For the general-form problem, let the function φ be defined as the spectral cut-off function that makes the first $p - r_L$ singular values of F_κ zero. Similarly, for the standard-form transformed problem, let ϱ be defined as the function that makes $p - r_L$ singular values that are 0 equal to $\sqrt{\kappa}$ and accordingly permutes the SVD. We then have $\varphi \circ \varrho = \text{Id}$. We have the following theorem.

Theorem 8. *Let $p > n$. The following diagram commutes.*

$$\begin{array}{ccc} \min_x \frac{1}{2} \|Ax - b\|_2^2 + \lambda\mathcal{R}(Lx) & \xrightarrow{\text{SR3}} & \begin{array}{l} \bar{y}_\kappa = \operatorname{argmin}_y \frac{1}{2} \|F_\kappa y - g_\kappa\|_2^2 + \lambda\mathcal{R}(y) \\ \bar{x}_\kappa = H_\kappa^{-1}(\kappa L^T \bar{y}_\kappa + A^T b) \end{array} \\ \downarrow & & \varphi \downarrow \uparrow \varrho \\ \begin{array}{l} \bar{z}_\kappa = \operatorname{argmin}_z \frac{1}{2} \|AL_A^\dagger z - b\|_2^2 + \lambda\mathcal{R}(y) \\ \bar{x}_\kappa = L_A^\dagger \bar{z}_\kappa + x_N \end{array} & \xrightarrow{\text{SR3}} & \begin{array}{l} \tilde{y}_\kappa = \operatorname{argmin}_y \frac{1}{2} \|\tilde{F}_\kappa y - g_\kappa\|_2^2 + \lambda\mathcal{R}(y) \\ \tilde{z}_\kappa = H_\kappa^{-1}(\kappa \tilde{y}_\kappa + (AL_A^\dagger)^T b) \\ \tilde{x}_\kappa = L_A^\dagger \tilde{z}_\kappa + x_N \end{array} \end{array}$$

6.3 Approximating the value function

In this section we quantify the distance between the Pareto curve of the original problem and the Pareto curve of the relaxed problem in terms of κ . We first describe the value function of the problem and then present our theorem.

The value function of an optimization problem expresses the value of the objective at the solution as a function of the other parameters. Using the standard-form transformation, we can, without loss of generality, consider the standard-form value function:

$$\phi_\kappa(\tau) = \min_y \|F_\kappa y - g_\kappa\|_2 \quad \text{s.t.} \quad \|y\|_p \leq \tau.$$

6.3.1 Value function for $\kappa \rightarrow \infty$

We have seen that for $\kappa \rightarrow \infty$, we retrieve the unrelaxed problem with value function

$$\phi_\infty(\tau) = \min_y \|Ay - b\|_2 \quad \text{s.t.} \quad \|y\|_p \leq \tau.$$

Following [111] we obtain the following (computable) upper and lower bounds for the value function

$$b^T \tilde{r} - \tau \|A^T \tilde{r}\|_q \leq \phi_\infty(\tau) \leq \|\tilde{r}\|_2,$$

where \tilde{y} is any feasible point (i.e., $\|\tilde{y}\|_p \leq \tau$), and $\tilde{r} = b - A\tilde{y}$ is the corresponding residual and $p^{-1} + q^{-1} = 1$. Moreover, by [111, Col. 2.2] the derivative of the value function is given by

$$\phi'_\infty(\tau) = -\|A^T \bar{r}\|_q / \|\bar{r}\|_2,$$

with $\bar{r} = b - A\bar{y}$ and $\bar{y} = \operatorname{argmin}_{\|y\|_p \leq \tau} \|Ay - b\|_2$.

To gain some insight in the behaviour of the value function, we consider ϕ_∞ and ϕ'_∞ at $\tau = 0$ and $\tau = \tau_* = \|A^\dagger b\|_p$:

$$\phi_\infty(0) = \|b\|_2, \quad \phi'_\infty(0) = -\|A^T b\|_q / \|b\|_2,$$

$$\phi_\infty(\tau_*) = \|(I - AA^\dagger)b\|_2, \quad \phi'_\infty(\tau_*) = 0.$$

This immediately suggests that ϕ_∞ decreases linearly near $\tau = 0$ (the zero solution) and flattens near $\tau = \tau_*$ (the unconstrained minimizer). Since ϕ_∞ is known to be convex, its second derivative is always positive and will gradually bend the curve from decreasing to flat. How fast this happens and whether one can expect the typical L-shape, depends on how fast the curve decreases initially. We can bound $\phi'_\infty(0)$ as follows. We let $b = Ay$ and find

$$\|A^T b\|_q = \|A^T Ay\|_q \geq C_q \|A^T Ay\|_2 \geq C_q \|A^\dagger\|_2^2 \|y\|_2,$$

where C_q is a constant that exists due to the equivalence of norms. Furthermore,

$$\|b\|_2 = \|Ay\|_2 \leq \|A\|_2 \|y\|_2.$$

From this we get

$$\phi'_\infty(0) \leq -C_q \kappa_2(A) \|A^\dagger\|_2,$$

with $\kappa_2(A) = \|A\|_2 \|A^\dagger\|_2$ the condition number of A . We thus expect a steep slope for ill-conditioned problems, giving rise for the characteristic L-shape of the curve. While this behavior is well-established for $p = 2$ where it can be analysed using the SVD of A [53], this analysis gives us new insight in the behavior of the Pareto curve for ill-posed problems for general p . An example for $p = 1$, $L = I$ is shown in figure 6.4.

6.3.2 Relaxed value function

We now present our theorem on the distance between the Pareto curve of the original problem and the Pareto curve of the relaxed problem.

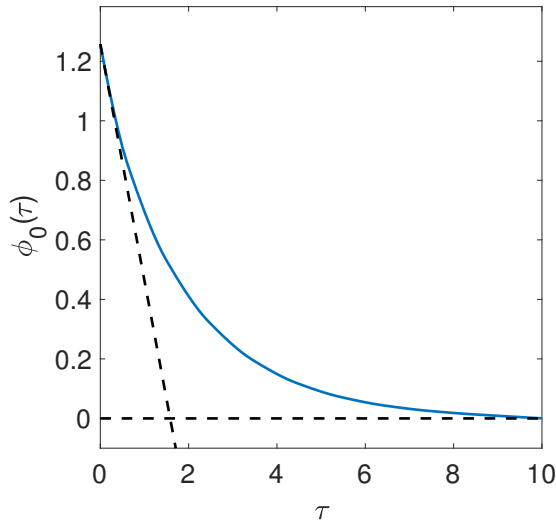


Figure 6.4: Pareto curve for an ill-posed problem; the matrix A is diagonal with elements $e^{-(i-1)/2}$ for $i = 1, 2, \dots, 10$; $b = Ax$ with $x = (1, 1, \dots, 1)$. The tangent lines at $\tau = 0$ and $\tau = \tau_*$ are shown in black.

Theorem 9. *The distance between the Pareto curve of the original problem and the Pareto curve of the relaxed problem is given by*

$$(\phi_\kappa(\tau))^2 - (\phi_\infty(\tau))^2 = -\kappa^{-1} \|A^T(b - A\bar{y}_\kappa)\|_2^2 + \mathcal{O}(\kappa^{-2}),$$

where \bar{y}_κ is the solution of the relaxed problem. In particular, we have

$$\phi_\kappa(\tau) \leq \phi_\infty(\tau).$$

Proof. Let $\epsilon = \kappa^{-1}$. The relaxed value function can be expressed as

$$\phi_\epsilon(\tau) = \min_y \|F_\epsilon y - g_\epsilon\|_2 \quad \text{s.t.} \quad \|y\|_p \leq \tau.$$

For $\epsilon < \|A\|_2^2$ we can expand $H_\epsilon^{-1} = \epsilon I - \epsilon^2 A^T A + \mathcal{O}(\epsilon^3)$ and get

$$F_\epsilon = \begin{pmatrix} A - \epsilon A A^T A + \mathcal{O}(\epsilon^2) \\ \epsilon^{1/2} A^T A + \mathcal{O}(\epsilon^{3/2}) \end{pmatrix}, \quad g_\epsilon = \begin{pmatrix} b - \epsilon A^T b + \mathcal{O}(\epsilon^2) \\ \epsilon^{-1/2} A^T b + \mathcal{O}(\epsilon^{3/2}) \end{pmatrix}.$$

Introduce

$$f(\epsilon) = (\phi_\epsilon(\tau))^2 = \min_{x,y} \|Ax - b\|_2^2 + \epsilon^{-1} \|x - y\|_2^2 \quad \text{s.t.} \quad \|y\|_p \leq \tau.$$

We have $f(0) = \min_{\|y\|_p \leq \tau} \|Ay - b\|_2^2 = (\phi_0(\tau))^2$. Furthermore

$$f'(\epsilon) = -\epsilon^{-2} \|\bar{x}_\epsilon - \bar{y}_\epsilon\|_2^2,$$

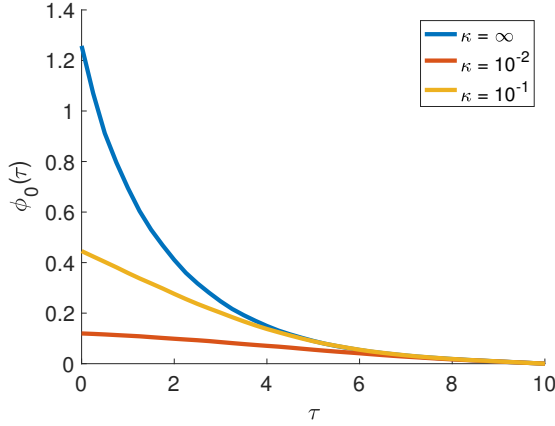


Figure 6.5: Pareto curve for an ill-posed problem; the matrix A is diagonal with elements $e^{-(i-1)/2}$ for $i = 1, 2, \dots, 10$; $b = Ax$ with $x = (1, 1, \dots, 1)$. The approximations for various values of κ are shown as well.

where $\bar{x}_\epsilon = H_\epsilon^{-1}(A^T b + \epsilon^{-1} \bar{y}_\epsilon)$ and \bar{y}_ϵ is the optimal y . With this we find

$$(\phi_\epsilon(\tau))^2 - (\phi_0(\tau))^2 = \epsilon f'(\eta) = -\epsilon \eta^{-2} \|\bar{x}_\eta - \bar{y}_\eta\|_2^2. \quad (6.22)$$

We conclude that $\phi_\epsilon(\tau) \leq \phi_0(\tau)$. Alternatively, we can express

$$(\phi_\epsilon(\tau))^2 - (\phi_0(\tau))^2 = -\epsilon^{-1} \|\bar{x}_\epsilon - \bar{y}_\epsilon\|_2^2 + \mathcal{O}(\epsilon^2). \quad (6.23)$$

For small ϵ we get

$$f'(\epsilon) = -\|A^T(b - A\bar{y}_\epsilon)\|_2^2 + \mathcal{O}(\epsilon).$$

Plugging this expression into eq. (6.23) gives the desired result. \square

Remark 2. theorem 9 can be used to explain the behaviour of the Pareto curves observed in the examples in section 6.1.1:

- The error gets smaller for large τ . For an unconstrained problem we have $\|A^T(b - A\bar{y}_\kappa)\|_2 = 0$ as $\kappa \rightarrow \infty$. An example is shown in fig. 6.5.
- The elbow of the Pareto curves coincide; ϕ_∞ decreases fast initially for ill-posed problems (cf. fig. 6.4) while ϕ_κ decreases less fast due to the implicit regularizing effect of the relaxation. Since $0 \leq \phi_\kappa \leq \phi_\infty$, the relaxed Pareto curve is pushed down and is therefore likely to have the elbow at the same location as ϕ_∞ .

6.4 Implementation

Recall from the introduction that we implement SR3 as follows:

$$x_{k+1} \leftarrow (A^T A + \kappa L^T L)^{-1} (A^T b + \kappa L^T y_k) \quad (6.24)$$

$$y_{k+1} \leftarrow \text{prox}_{1/\kappa \mathcal{R}}(Lx_k). \quad (6.25)$$

The last equation shows that for the choice $\mathcal{R}(\cdot) = \lambda \|\cdot\|_p^p$ there is a relation between the parameters κ and λ . More specifically, λ depends on κ and hence we write $\lambda(\kappa)$. Given the optimal λ_* , we have $\lambda(\kappa) = \lambda_* \cdot \kappa$. Note that if we use the constrained formulation eq. (6.9), the dependence on the stepsize is lost because the proximal operator is the indicator function, and there is no relation between τ and κ .

The computational bottleneck is in the first step, which is the solution to the large-scale linear system

$$(A^T A + \kappa L^T L) x_k = A^T b + \kappa L^T y_{k-1}. \quad (6.26)$$

To avoid explicitly forming $A^T A$ and $L^T L$, we instead solve the following minimization problem

$$\min_x \left\| \begin{bmatrix} A \\ \sqrt{\kappa} L \end{bmatrix} x - \begin{bmatrix} b \\ \sqrt{\kappa} y_{k-1} \end{bmatrix} \right\|_2^2, \quad (6.27)$$

with LSQR.

We will numerically investigate how only partially solving eq. (6.27) affects the convergence of SR3. This has been investigated for ADMM in [32, 33, 2]. The convergence of FISTA with an inexact gradient has been analyzed in [103]. The key message is that the error has to go down as the iterations increase.

In our implementation, we propose two extra ingredients to make SR3 suitable for large-scale problems: warm starts and inexact solves of (6.27). Both ingredients are also used in the implementation of ADMM [18]. However, we propose a new stopping criterion for the inexact solves of (6.27).

A *warm start* is a technique used in inner-outer schemes, where the solution of the previous inner iteration serves as an initial guess to the new inner iteration. That is, we solve

$$\min_x \left\| \begin{bmatrix} A \\ \sqrt{\kappa} L \end{bmatrix} x - \left(\begin{bmatrix} b \\ \sqrt{\kappa} y_{k-1} \end{bmatrix} - \begin{bmatrix} A \\ \sqrt{\kappa} L \end{bmatrix} x_{k-1} \right) \right\|_2^2. \quad (6.28)$$

By *inexact solves* we mean finding an approximate solution to (6.28). The level of inexactness is determined by the difference between the true solution and the inexact solution. There are various ways in which one can solve the optimization problem inexactly. One way is to simply determine a maximum number of iterations. However, the number of iterations to solve (6.27) can vary strongly per outer iteration. Moreover, we may not want to solve the inner system with high precision in the first few outer iterations, because this does not result in significant improvement in the next outer iteration. Recently, the authors in [117] proposed a criterion to determine the amount of inexactness for inner-outer schemes. The idea is to stop the inner iteration once the difference in the resulting outer iterate becomes stagnant. Let x_k denote the current inner iterate and $y_k = \text{prox}_{1/\kappa \mathcal{R}}(Lx_k)$ the resulting outer iterate by applying the proximal operator. Then the authors in [117] propose to stop the inner iterations if

$$\|x_{k+1} - x_k\| < \rho \|y_{k+1} - y_k\|, \quad (6.29)$$

for some user defined constant ρ . We propose a similar criterion, namely to stop if

$$\frac{\|y_{k+1} - y_k\|}{\|y_k\|} < \epsilon, \quad (6.30)$$

for some user defined threshold ϵ . The index k refers to the iteration of the iterative method applied to the inner iteration. This yields the proposed implementation of SR3, shown in algorithm 8. Note that in line 4 of the algorithm we use the LSQR algorithm, and we build on the Krylov subspace from the previous step.

Algorithm 8 Implementation of SR3

Require: Operators A and L , the data b and the parameters κ , λ and ϵ .

Ensure: Approximate solution x_k .

```

1: while  $\|x_{k+1} - x_k\| > \delta$  do
2:    $l = 0$ .
3:   while  $\frac{\|\tilde{y}_{l+1} - \tilde{y}_l\|}{\|\tilde{y}_l\|} > \epsilon$  do            $\triangleright$  Run LSQR. We do not restart LSQR every
      iteration!
4:      $x_l = \operatorname{argmin}_{x \in \mathcal{K}_l} \left( \left[ \begin{array}{c} A \\ \sqrt{\kappa}L \end{array} \right], \left[ \begin{array}{c} b - Ax_k \\ \sqrt{\kappa}(y_0 - Lx_k) \end{array} \right] \right) \left\| \left[ \begin{array}{c} A \\ \sqrt{\kappa}L \end{array} \right] x - \left( \left[ \begin{array}{c} b \\ \sqrt{\kappa}y_k \end{array} \right] - \left[ \begin{array}{c} A \\ \sqrt{\kappa}L \end{array} \right] x_k \right) \right\|_2^2$ .
5:      $\tilde{y}_{l+1} = \operatorname{prox}_{1/\kappa\mathcal{R}}(Lx_l)$ .            $\triangleright$  Prospective update
6:      $l = l + 1$ .
7:   end while
8:    $y_k = \tilde{y}_l$ .
9:    $k = k + 1$ .
10: end while

```

It is important to note that the influence of κ on the outer iteration is different from the influence of κ on the inner iteration. The improved conditioning of the matrix F_κ pertains to the convergence of the outer iteration. The convergence of the inner iteration is completely determined by the properties of the matrix H_κ^{-1} . It is important to note that using the GSVD of (A, L) we get

$$H_\kappa = A^T A + \kappa L^T L = X^T (\Sigma^T \Sigma + \kappa \Gamma^T \Gamma) X,$$

but this is not the SVD of H_κ , because X is not orthonormal. Therefore, the matrix $\Sigma^T \Sigma + \kappa \Gamma^T \Gamma$ does not tell us anything about the convergence rate when solving linear systems involving H_κ .

6.5 Numerical experiments

In this section we verify the results from section 6.2 numerically. Furthermore, we implement algorithm 8 and test it on two examples. We use two examples that are regularized by TV regularization, which we solve in its constrained form, i.e.

$$\min_x \|Ax - b\|_2^2 \quad \text{s.t.} \quad \|Lx\|_1 \leq \tau.$$

6.5.1 Examples

We will use two examples that are very different in nature in terms of their singular values. For both examples, we will show how their spectra are changed as a

function of κ by applying SR3, and how this relates to the inner and outer iterations. After that, we will show how our inexact SR3 greatly reduces the total number of iterations. We do not add noise to the data.

Gravity surveying

The first example is the gravity example from the regu toolbox, [56, 54]. This example models gravity surveying. An unknown mass distribution that generates a gravity field is located in the subsurface, and the measured data is related to the gravity field via a Fredholm integral of the first kind, i.e.

$$b(s) = \int_{\Omega} k(s, t)x(t)dt.$$

The variable $x(t)$ is the mass density at the location t in the subsurface and $b(s)$ is the gravity field at location s at the surface. The kernel is given by:

$$k(s, t) = d(d^2 + (s - t)^2)^{-3/2},$$

where d is the depth. The integral is discretized using the midpoint quadrature rule and yields a symmetric Toeplitz matrix A that is square and severely ill-posed. We have chosen an $x(t)$ that is piecewise constant and hence we regularize the problem with TV regularization. The operator $L = D$, where D is the first-order finite difference discretization, i.e.

$$D = \begin{bmatrix} -1 & 1 & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{(n-1) \times n}.$$

The operator is underdetermined and its nullspace has dimension 1. We choose $n = 512$. The true gravity profile is shown in fig. 6.6.

Tomography

Our second example is the tomography example PRtomo from the IR Tools toolbox [37], see also [59], which models parallel tomography. It models X-ray attenuation tomography, often referred to as computerized tomography (CT). Parallel rays at different angles penetrate an object. The rays are attenuated at a rate proportional to the length of the ray and the density of the object. The i -th ray can be modeled as

$$b_i = \sum_{j \in \mathcal{S}_i} a_{ij}x_j.$$

The set \mathcal{S} denotes the set of pixels that are penetrated, a_{ij} denotes the length of the i -th ray through the j -th pixel and x_j is the attenuation coefficient. This is a 2D example where the matrix A is underdetermined and the singular values decay mildly. Again, we use TV regularization for the reconstruction. For 2D regularization, the operator $L = \begin{bmatrix} I \otimes D \\ D \otimes I \end{bmatrix}$. Hence, the operator L is overdetermined and has a nullspace of dimension 1. We choose 18 angles between 0 and 180 degrees and discretize the image on a 128×128 -pixel grid. This means that $A \in \mathbb{R}^{3258 \times 16384}$. Our experiments are on the Shepp-Logan phantom, shown in fig. 6.6.

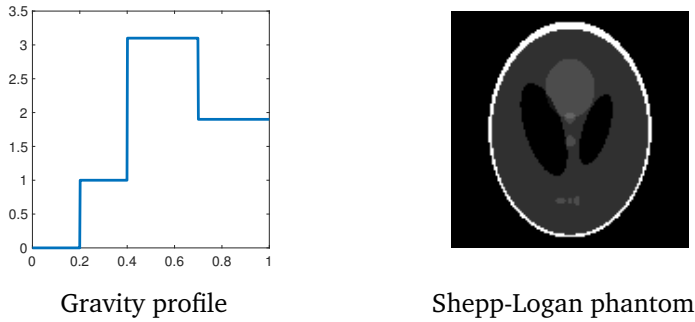


Figure 6.6

Parameters

For our experiments, we have adapted the implementation of the accelerated proximal gradient algorithm from [94] for SR3 and use the same stopping criterion for the proximal gradient algorithm. For the inexact stopping criterion for the inner iteration we choose $\epsilon = 10^{-6}$. For the exact SR3 method, we let LSQR run to convergence with the standard tolerance of 10^{-6} . For τ , we choose the optimal value $\tau = \|Lx_{\text{true}}\|_1$.

6.5.2 Singular values of F_κ

In this section we show the singular values of F_κ for the gravity and the tomography example. For the tomography example, the generalized singular values are calculated on a 64×64 grid to reduce computational time, instead of the 128×128 grid for our experiments. We show the generalized singular values, i.e. the singular values of AL_A^\dagger , and the singular values of F_κ for different values of κ for the gravity example in fig. 6.7. Note that irrespective of the value of κ , the

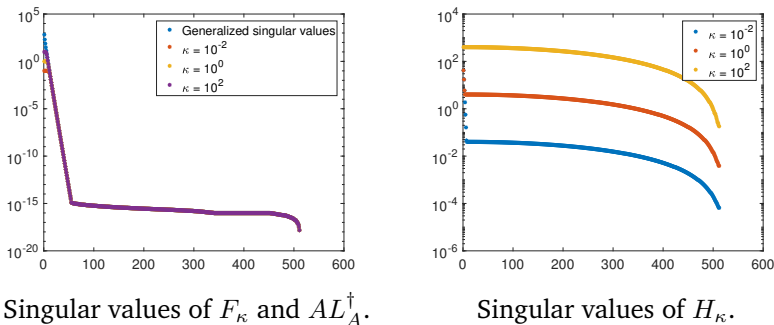
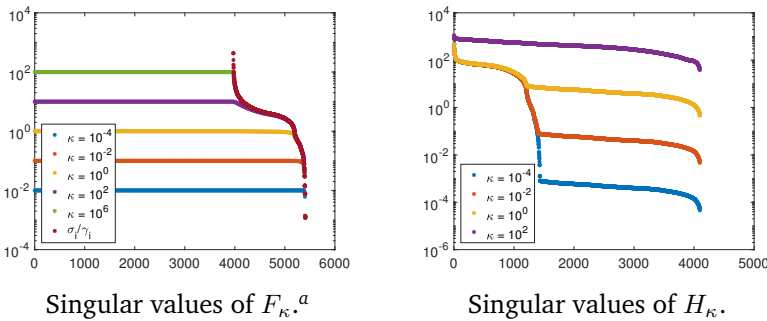


Figure 6.7: Spectral properties of F_κ and H_κ for the gravity example. Left figure: We show the singular values of AL_A^\dagger and the singular values of F_κ for different values of κ . Note that the singular values of F_κ have a very similar structure to the singular values of AL_A^\dagger . Right figure: The singular values of the matrix H_κ .

matrix F_κ remains severely ill-posed. For the tomography example, A is not severely ill-posed. The singular values decay only mildly and the situation is different. In this case, for small κ ,

$$\psi_i(F_\kappa) = \sqrt{\frac{\sigma_{r-i+1}^2}{\sigma_{r-i+1}^2/\kappa + \gamma_{r-i+1}^2}} \approx \sqrt{\frac{\sigma_{r-i+1}^2}{\sigma_{r-i+1}^2/\kappa}} = \sqrt{\kappa}.$$

Hence, for small κ the singular values of $F_\kappa \approx \sqrt{\kappa}$ and the condition number is 1. As $\kappa \rightarrow \infty$ we have seen that $\psi_i(F_\kappa) \rightarrow \frac{\sigma_{r-i+1}}{\gamma_{r-i+1}}$. We show the singular values, the generalized singular values, and the singular values of F_κ in fig. 6.8. Note that for this example, the conditioning of the matrix F_κ is improved.



^aThe matrix is numerically rank deficient and we have truncated the SVD.

Figure 6.8: Spectral properties of F_κ and H_κ for the tomography example. The left figure shows the singular values of F_κ . Recall that the first $p - r_L$ singular values of F_κ are $\sqrt{\kappa}$. The right figure shows the singular values of H_κ . There is an inverse relation between the condition number of H_κ and F_κ as a function of κ .

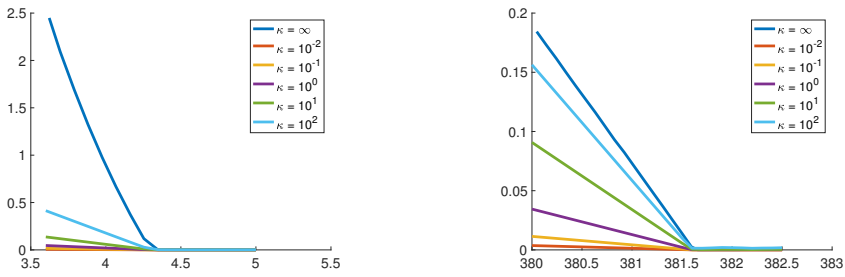
6.5.3 The Pareto curves

In fig. 6.9 we show the Pareto curves for the original problem and SR3 for both our examples. As we explained in section 6.3, the corner of the Pareto of the original problem and SR3 is likely to be in the same place. This is confirmed by fig. 6.9.

6.5.4 The influence of κ on the number of iterations

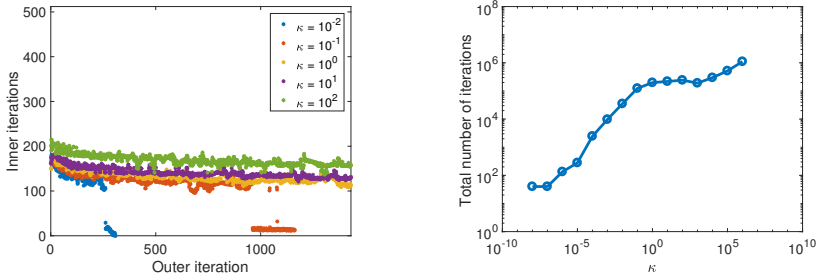
To investigate the influence of κ , we show the number of inner and outer iterations for varying values of κ and the total number of iterations. The results are shown in fig. 6.10 and fig. 6.11. As we have stated before, the improved convergence rate due to an improved conditioning of κ pertains to the outer iterations. The effect of κ on the convergence of the inner iteration may be completely opposite.

For the gravity example, we see that the number of inner iterations varies very little as κ increases, and even goes up a little bit. This is not unexpected, because the decay of the singular values changes very little as κ increases, see fig. 6.7. The



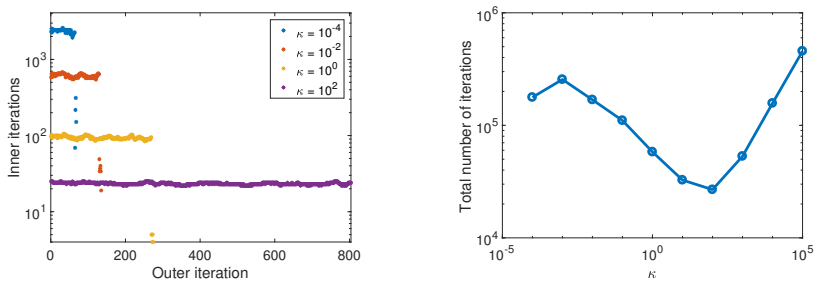
Pareto curves for the gravity example. Pareto curves for the tomography example.

Figure 6.9: The left figure shows the Pareto curves for the gravity example. The right figure shows the Pareto curves for the tomography example. The x-axis is τ and the y-axis is $\|A\bar{x}_\kappa - b\|_2$.



Inner iterations versus outer iterations. Total number of iterations.

Figure 6.10: The left panel shows the inner and outer iterations for varying κ for the gravity example. The right panel shows the total number of iterations.



Inner iterations versus outer iterations. Total number of iterations.

Figure 6.11: The left panel shows the inner and outer iterations for varying κ for the tomography example. The right panel shows the total number of iterations.

number of outer iterations goes down rapidly as κ decreases, something that is not expected from the distribution of the singular values. This shows that the

distribution of the singular values is not the sole property explaining the convergence behavior.

For the tomography example we see a clear trade-off between inner and outer iterations. From fig. 6.8 we clearly see that as the condition number of F_κ decreases, the condition number of H_κ increases. This explains that, as the number of inner iterations goes down with increasing κ , the number of outer iterations goes down.

6.5.5 Inexact SR3

In this section we compare the error and the total number of iterations for SR3 and inexact SR3 as a function of κ . The results are shown in fig. 6.12. We see

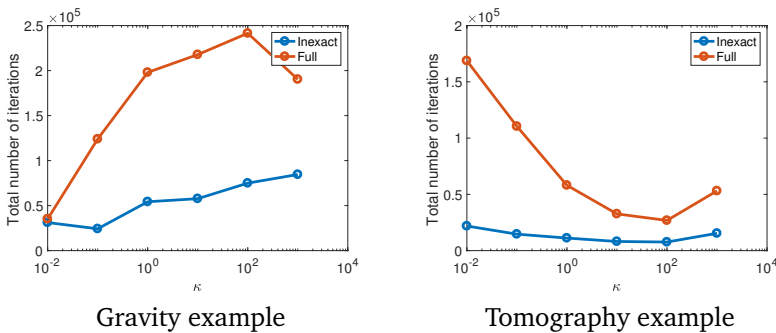


Figure 6.12: Comparison of the total number of iterations for SR3 and inexact SR3 as a function of κ . Note that the axes are on a log-log scale.

that the total number of iterations needed is greatly reduced by implementing the automated stopping criterion. Another important contribution is that the stopping criterion seems to mitigate the influence of κ on the total number of iterations. fig. 6.13 and fig. 6.14 show some reconstructions for different values of κ .

6.6 Conclusion and outlook

In this chapter we have analyzed the method SR3 which was introduced in [130]. We have extended theorem 1 from [130] about the singular values of F_κ to the general form case. We have shown that SR3, as $\kappa \rightarrow \infty$, implicitly applies a standard-form transformation, and that for finite $\kappa > 0$, the singular values of F_κ are related to the standard-form transformed operator.

In section 6.3 we have shown that the distance between the Pareto curve of the original problem and the Pareto curve of the relaxed problem is of $\mathcal{O}(1/\kappa^2)$ plus the norm of the gradient, which depends on κ .

In section 6.4 we have presented our implementation of the inexact SR3 algorithm, where we have proposed an automated stopping criterion for the inner iterations.

In our numerical experiments in section 6.5 we have compared the SR3 algorithm for two example problems with very different spectra. The gravity example is a

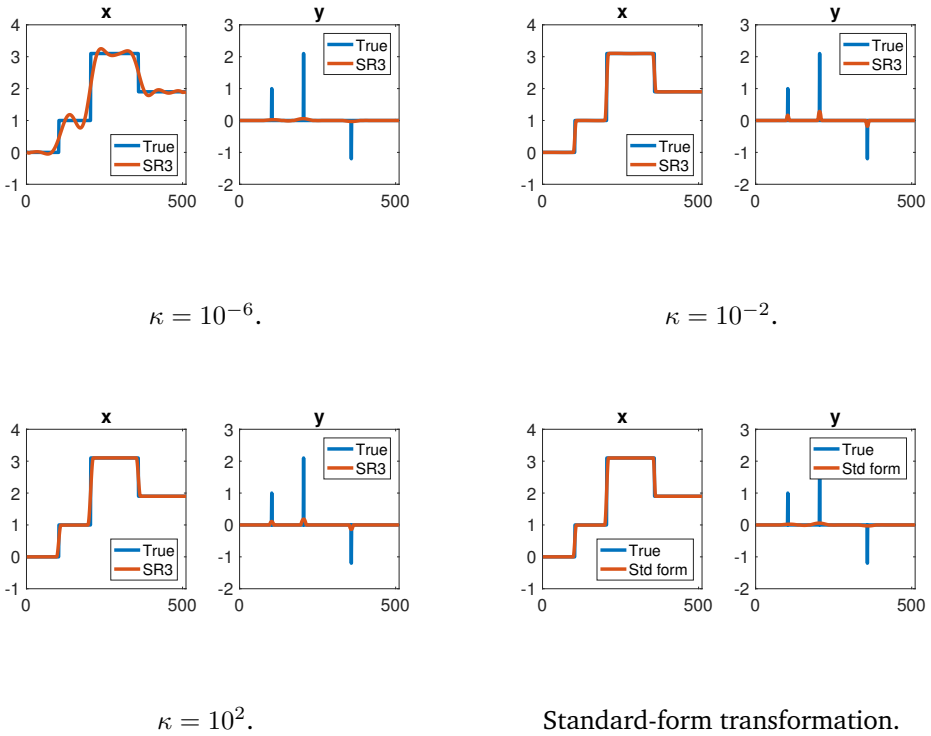
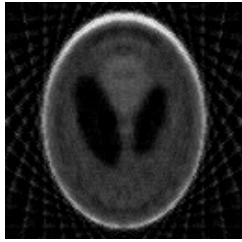


Figure 6.13: Solution to the gravity example for different κ and the optimal τ . We show both \bar{x}_κ and \bar{y}_κ .

severely ill-posed problem and we have shown, numerically, that the convergence of inner iterations is not affected much by κ , but the convergence of the outer iteration is. For the tomography example we saw a trade-off: as κ decreases the outer iterations converge rapidly, but the number of inner iterations is large. We have shown that our automated stopping criterion greatly reduces the number of iterations needed.

For future research it would be interesting to further investigate the relation between the Pareto curve of the original problem and of the relaxed problem. Specifically, it would be great if we could prove that the corner of the curves are in the same place, something that we have only been able to show qualitatively through theorem 9. This would lead to automatic selection of the regularization parameter λ .

Another interesting topic of research is the selection of κ . As we have seen in our experiments, the choice of κ strongly influences the number of iterations needed for SR3, although this is largely mitigated by the inexact stopping criterion. The relation between the tolerance for the stopping criterion and κ should also be



$\kappa = 10^{-8}$.



$\kappa = 10^{-2}$.



$\kappa = 10^0$.



Standard-form transformation.

Figure 6.14: Solution to the tomography example for different κ and the optimal τ .

further investigated.



Conclusion and outlook



7.1 Conclusion

In this thesis we have developed algorithms for linear inverse problems. Our main contributions are:

- **Compared Lanczos- and RSVD-based algorithm for regularization parameter estimation λ**

We have compared the use of Lanczos- and RSVD-based algorithms for estimating the regularization parameter. The goal is to obtain a low-dimensional model that allows for rapid evaluation of linear systems for multiple values of λ . We have implemented a new adaptive RSVD-based algorithm that automatically determines the dimensions of the low-dimensional model, such that it is both of low-dimension and accurately represents the full model. Using the Lanczos process one can derive exact lower and upper bounds for the parameter selection methods. In some cases, this yields a favorable reconstruction as compared to the RSVD method. Furthermore, we have compared the use of a randomized trace estimator versus estimating the trace using estimates of the singular values obtained via the Lanczos process or the RSVD. It turns out that the performance of each method depends strongly on the spectrum of the matrix whose trace is estimated.

- **Defined a mathematical framework for regularizing Multi-Dimensional Deconvolution**

We have posed Multi-Dimensional Deconvolution (MDD) as a constrained optimization problem. We have shown the ill-posedness of the MDD problem and have discussed how to incorporate source-receiver reciprocity and causality as constraints. We have shown that even with these constraints the problem remains ill-posed, as it still exhibits semi-convergence. Additional regularization has to take care of this. We have applied Tikhonov regularization and have shown that this mitigates the effects of semi-convergence. However, standard parameter selection methods fail to predict a good estimate of regularization parameter λ .

- **Extended the analysis on SR3**

We have extended the analysis on the SR3 method from [130]. We have used the Generalized Singular Value Decomposition (GSVD) to derive the spectrum of the matrix formed by applying SR3. We have related this to the spectrum of the matrix of the original problem and show that the condition number decreases. Moreover, we have shown that in the limit $\kappa \rightarrow \infty$ SR3 applies a standard-form transformation.

- **Proposed an automated stopping criterion for SR3**

We have proposed an automated stopping criterion for the inner iterations of SR3. The stopping criterion is based on the progress made in the outer iteration with one inner iteration. When this stagnates, we stop the inner iterations. We have shown that this stopping criterion reduces the total number of iterations and that the total number of iterations varies far less with κ .

7.2 Discussion and outlook

The question whether one can obtain a good estimate for the regularization parameter based on the data remains an open question. For Tikhonov regularization, we have reviewed some parameter selection methods and studied how to efficiently evaluate them. However, we have seen that for Multi-Dimensional Deconvolution these methods all fail to provide a correct estimate of the regularization parameter. This MDD problem is, however, an extremely difficult problem to solve, and it stands to reason that, given that all attempts so far have failed, a parameter selection method may not be an achievable goal.

For SR3, there are a number of future directions based on our research. The first one would be to obtain an estimate for κ . However, we have shown that the influence of κ can only be determined a-priori by looking at either the singular values or the generalized singular values. Perhaps this problem is as hard as obtaining an estimate for the regularization parameter.

We have shown how the Pareto curve of the relaxed problem obtained by SR3 relates to the Pareto curve of the original problem. We have argued that the corner, or elbow, of the Pareto curve for the relaxed problem and the original problem may coincide. This is confirmed by numerical experiments. For future research it would be interesting to see whether the exact location of the corner can be quantified for both the original and the relaxed problem.

Finally, experiments with real data have to show whether the corner of the Pareto curve gives a good estimate for the regularization parameter, and whether SR3 is truly competitive with FISTA and ADMM.

This thesis has only covered linear inverse problems, and the logical extension would be to consider also non-linear inverse problems. It is unlikely that the contents of this thesis extend to this case. Ideally, we would like to obtain a low-dimensional model that can be evaluated for multiple regularization parameters, and accurately reflects the full model.

As a first attempt, we have tried to view solving the inverse problem via gradient descent as a dynamical system. Consider the problem

$$\min_x \frac{1}{2} \|Ax - b\|_2^2 + \lambda \|Lx\|_2^2. \quad (7.1)$$

We can consider the solution to this problem as a gradient flow, namely

$$\dot{x}_\lambda(t) = F(x_\lambda(t)) := (A^T A + \lambda L^T L)x_\lambda(t), \quad (7.2)$$

which is a parametric dynamical system. By taking snapshots of (7.2) one obtains the iterates of gradient descent. Our goal is to use model order reduction

approaches from the dynamical systems literature to obtain a low dimensional model in terms of λ for (7.2). There exist various approaches in the dynamical systems literature for model order reduction, for an overview see [11]. One approach that seems appropriate here, is to evaluate this dynamical system for various $\lambda_1, \dots, \lambda_p$, and interpolate between these values to obtain a low-dimensional surrogate model. The question is how to interpolate. We have attempted to use manifold interpolation, where the λ_i are interpolated on the underlying manifold, for details we refer to [11, 3]. Although it is an interesting approach, the first results were not satisfactory. We have tested this on a tomography example where we try to image a smooth object, and hence L is the discretization of the first derivative operator.

We describe the approach for interpolating between two values of λ .

1. Pick two values of λ and run gradient descent. These yield snapshots of the gradient flow.
2. Use Proper Orthogonal Decomposition (POD) (in this case the SVD) to obtain orthonormal bases U_1 and U_2 for the respective gradient flows. Truncate the POD if necessary.
3. Interpolate the orthonormal bases using manifold interpolation, yielding a basis U_3 .
4. Project the gradient flow onto U_3 to obtain the reduced order model and run gradient descent.

We determined the optimal value for λ by hand, which turned out to be $\lambda = 1000$. We then chose $\lambda_1 = 500$ and $\lambda_2 = 1500$ and interpolate the solution to compare the two. We add 10% noise to the data. As a benchmark, we also compare this approach to direct linear interpolation, which it should outperform. The true background is shown in figure (7.1), and the solutions for λ , λ_1 and λ_2 are shown in figure (7.2). We show the solution and the interpolated solutions in figure (7.3).

Unfortunately, the interpolated solution differs quite a lot from the true solution,

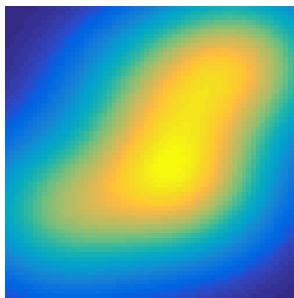


Figure 7.1: True background.

and does not significantly outperform linear interpolation.

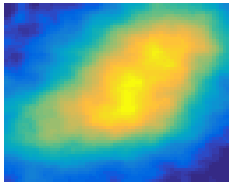
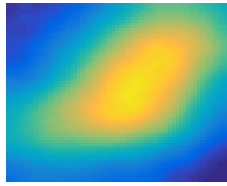
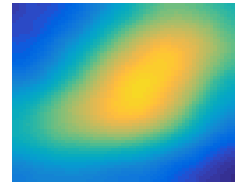
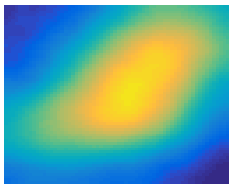
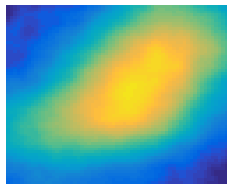
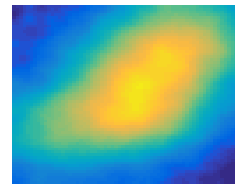
Solution for $\lambda_1 = 500$ Solution for $\lambda = 1000$ Solution for $\lambda_2 = 1500$

Figure 7.2: Solution for different λ . We interpolate between λ_1 and λ_2 to obtain the solution for $\lambda = 1000$.

Solution for $\lambda = 1000$ 

Manifold interpolation



Linear interpolation

Figure 7.3: Interpolated solutions.

The goal of the example is to obtain a low-dimensional model that we can evaluate for multiple λ . If $L = I$ and the matrix A is linear then Krylov methods are excellent for this purpose, but for other regularization methods no such technique exists. Unfortunately, manifold interpolation on the gradient flow does not work and we need other approaches.



Bibliography



- [1] ARSAC notes for guidance: good clinical practice in nuclear medicine. <https://www.gov.uk/government/publications/arsac-notes-for-guidance>.
- [2] M.M. Alves, J. Eckstein, M. Geremia, and G. M. Jefferson. Relative-error inertial-relaxed inexact versions of Douglas-Rachford and ADMM splitting algorithms. *Comput. Optim. Appl.*, 75:389–422, 2020.
- [3] D. Amsallem and C. Farhat. Interpolation method for the adaptation of reduced-order models to parameter changes and its application to aeroelasticity. *AIAA J.*, 46(7):1803–1813, 2008.
- [4] A. Aravkin, R. Kumar, H. Mansour, B. Recht, and F.J. Herrmann. Fast methods for denoising matrix completion formulations, with applications to robust seismic data interpolation. *SIAM J. Sci. Comput.*, 36:237–266, 2014.
- [5] A.Y. Aravkin, J.V. Burke, and M.P. Friedlander. Variational properties of value functions. *SIAM J. Optim.*, 23:1689–1717, 2013.
- [6] S.R. Arridge and J.C. Schotland. Optical tomography: forward and inverse problems. *Inverse Problems*, 25(12):123010, 2009.
- [7] Z. Bai, M. Fahey, and G.H. Golub. Some large-scale matrix computation problems. *J. Comput. Appl. Math.*, 74:71–89, 1996.
- [8] A.B. Bakushinskii. Remarks on choosing a regularization parameter using the quasi-optimality and ratio criterion. *USSR Comput. Math. Math. Phys.*, 24:181–182, 1981.
- [9] F. Bauer and M.A. Lukas. Comparing parameter choice methods for regularization of ill-posed problems. *Math. Comput. Simulation*, 81(9):1795–1841, 2011.
- [10] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2:183–202, 2009.
- [11] P. Benner, S. Gugercin, and K. Wilcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Rev.*, 57:483–531.
- [12] A.J. Berkhout. *Seismic Migration. Imaging of Acoustic Energy by Wave Field Extrapolation*. Elsevier: Elsevier, 1982.
- [13] A.J. Berkhout. Review paper: An outlook on the future of seismic imaging,

- part I: forward and reverse modelling. *Geophysical Prospecting*, 62(5):911–930, 2014.
- [14] A. Berrington de Gonzalez, M. Mahesh, K.P. Kim, M. Bhargavan, R. Lewis, F. Mettler, and C. Land. Projected cancer risks from computed tomographic scans performed in the united states in 2007. *Archive of Internal Medicine*, 169:2071–2077, 2009.
- [15] M. Bertero and P. Boccacci. *Introduction to Inverse Problems in Imaging*. Institute of Physics Publishing, Bristol, 1998.
- [16] A.G.J. Besson, J.P. Thiran, and Y. Wiaux. *Imaging from Echoes: On Inverse Problems in Ultrasound*. Ecole Polytechnique Fédérale de Lausanne, 2019.
- [17] B. Borden. Mathematical problems in radar inverse scattering. *Inverse Problems*, 18:R1, 2001.
- [18] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the Alternating Direction Method of Multipliers. *Foundations and Trends® in Machine Learning*, 3:1–122, 2011.
- [19] D.J. Brenner and E.J. Hall. Computed tomography - an increasing source of radiation exposure. *The New England Journal of Medicine*, 357:2277–2284, 2007.
- [20] C. C. Stolk. A pseudodifferential equation with damping for one-way wave propagation in inhomogeneous acoustic media. *Wave Motion*, 40(2):111–121, 2004.
- [21] D. Calvetti, G. H. Golub, and L. Reichel. Estimation of the L-curve via Lanczos bidiagonalization. *BIT*, 39(4):603–619, 1999.
- [22] D. Calvetti and L. Reichel. Tikhonov regularization of large linear problems. *BIT*, 43(2):261–281, 2003.
- [23] D. Calvetti, G. Spaletta, L. Reichel, and F. Sgallari. An L-ribbon for large underdetermined linear discrete ill-posed problems. *Numer. Algorithms*, 25:89–107, 2000.
- [24] E.J. Candès and L. Demanet. The curvelet representation of wave propagators is optimally sparse. *Comm. Pure Appl. Math.*, 58:1472–1528, 2005.
- [25] E.J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52:489–509, 2006.
- [26] J. Chung, J.G.. Nagy, and D.P. O’Leary. A weighted GCV method for Lanczos hybrid regularization. *Electron. Trans. Numer. Anal.*, 28:149 – 167, 2008.
- [27] J.F. Claerbout. Toward a unified theory of reflector mapping. *Geophysics*, 36(3):467–481, 1971.
- [28] I. Daubechies, M. Defrise, and C. de Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Comm. Pure Appl. Math.*, 57:1413–1457, 2004.
- [29] F.J.H. Don. On the symmetric solutions of a linear matrix equation. *Linear Algebra App*, 93:1–7, 1987.
- [30] M.W. Downey. Oil and natural gas exploration. In *Encyclopedia of Energy*, pages 549–558. Elsevier, 2004.
- [31] J. Eckstein and W. Yao. Understanding the convergence of the Alternating Direction Method of Multipliers: Theoretical and Computational

- Perspectives. *Pac. J. Optim.*, 11:619–644, 2015.
- [32] J. Eckstein and W. Yao. Approximate ADMM algorithms derived from Lagrangian splitting. *Comput. Optim. Appl.*, 68:363–405, 2017.
- [33] J. Eckstein and W. Yao. Relative-error approximate versions of Douglas–Rachford splitting and special cases of the ADMM. *Math. Program.*, 170:417–444, 2018.
- [34] L. Eldén. A weighted pseudoinverse, generalized singular values, and constrained least squares problems. *BIT*, 22:487–502, 1982.
- [35] H.W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Kluwer Academic Publishers, Dordrecht, 1996.
- [36] A. Garg, S. Sharma, and D.J. Verschuur. Reservoir elastic parameters estimation from surface seismic data using JMI-res: A full-wavefield approach. *80th EAGE Conference and Exhibition 2018*, 2018.
- [37] S. Gazzola, P.C. Hansen, and J.G. Nagy. IR tools: a MATLAB package of iterative regularization methods and large-scale test problems. *Numer. Algorithms*, 81:773–811, 2019.
- [38] S. Gazzola, P. Novati, and M.R. Russo. On Krylov projection methods and Tikhonov regularization. *Electron. Trans. Numer. Anal.*, 44:83–123, 2015.
- [39] G.H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.
- [40] G.H. Golub, F.T. Luk, and M.L. Overton. A block Lanczos method for computing the singular values and corresponding singular vectors of a matrix. *ACM Trans. Math. Softw.*, 7(2):149–169, 1981.
- [41] G.H. Golub and G. Meurant. *Matrices, Moments and Quadrature with Applications*. Princeton University Press, Princeton, NJ, USA, 2009.
- [42] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 3 edition, 2013.
- [43] G.H. Golub and U. von Matt. Generalized cross-validation for large scale problems. *J. Comput. Graph. Stat.*, 6:1–34, 1995.
- [44] G.H. Golub and U. von Matt. Tikhonov regularization for large scale problems. in workshop on scientific computing, 1997.
- [45] I. Goodfellow, Y. Bengio, A. Courville, and F. Bach. *Deep Learning*. MIT Press, Cambridge, MA, 2016.
- [46] M. Gutknecht. Block Krylov space methods for linear systems with multiple right-hand sides: An introduction, 2006.
- [47] J. Hadamard. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton University Bulletin*, pages 49–52, 1902.
- [48] N. Halko, P. Martinsson, and J. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, 53(2):217–288, 2011.
- [49] M. Hanke. On Lanczos based methods for the regularization of discrete ill-posed problems. *BIT*, 41:pages1008–1018, 2001.
- [50] M. Hanke. A note on Tikhonov regularization of large linear problems. *BIT*, 43:449–451, 2003.

- [51] P.C. Hansen. The truncated SVD as a method for regularization. *BIT*, 27:534–553, 1987.
- [52] P.C. Hansen. Regularization, GSVD and truncated GSVD. *BIT*, 29:491–504, 1989.
- [53] P.C. Hansen. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Rev.*, 34:561–580, 1992.
- [54] P.C. Hansen. Regularization tools: A MATLAB package for analysis and solution of discrete ill-posed problems. *Numer. Algorithms*, 6:1–35, 1994.
- [55] P.C. Hansen. *Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion*. SIAM, 1998.
- [56] P.C. Hansen. Deconvolution and regularization with Toeplitz matrices. *Numerical Algorithms*, pages 323–378, 2002.
- [57] P.C. Hansen. *Discrete Inverse Problems: Insight and Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2010.
- [58] P.C. Hansen. Oblique projections and standard-form transformations for discrete inverse problems. *Numer. Linear Algebra Appl.*, 20:250–258, 2013.
- [59] P.C. Hansen and J.S. Jørgensen. Air tools II: algebraic iterative reconstruction methods, improved implementation. *Numer. Algorithms*, 79:107–137, 2018.
- [60] P.C. Hansen, J.G. Nagy, and D.P. O’Leary. *Deblurring images: Matrices, Spectra and Filtering*. SIAM, 2006.
- [61] P.C. Hansen and D.P. O’Leary. The use of the L-curve in the regularization of discrete ill-posed problems. *SIAM J. Sci. Comput.*, 14(6):1487–1503, 1993.
- [62] G. Hennenfent and F.J. Herrmann. Simply denoise: Wavefield reconstruction via jittered undersampling. *Geophysics*, 73:19–28, 2008.
- [63] G.T. Herman. *Fundamentals of Computerized Tomography: Image Reconstruction from Projections*. Springer, New York, 2009.
- [64] F.J. Herrmann, D. Wang, and D.J. Verschuur. Adaptive curvelet-domain primary-multiple separation. *Geophysics*, 73:1MJ–Z46.
- [65] M.R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49:409–436, 1952.
- [66] M.E. Hochstenbach and L. Reichel. An iterative method for Tikhonov regularization with a general linear regularization operator. *Journal of Integral Equations*, 22:463–480, 2010.
- [67] M.E. Hochstenbach, L. Reichel, and X. Yu. A Golub-Kahan-type reduction method for matrix pairs. *Journal of Scientific Computing*, 65:767–789, 2015.
- [68] M.F. Hutchinson. A stochastic estimator for the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics, Simulation and Computation*, 19:433–450, 1990.
- [69] J. Johnson. Seismic wavefield reconstruction using reciprocity. masters, The University of British Columbia, Vancouver, 03 2013. (MSc).
- [70] J. Johnson, T.T.Y. Lin, and F.J. Herrmann. Estimation of primaries via sparse inversion with reciprocity. *EAGE Annual Conference Proceedings*, 2010.
- [71] B. Jumah and F.J. Herrmann. Dimensionality-reduced estimation of primaries by sparse inversion. *Geophysical Prospecting*, 62:972–993, 2014.
- [72] M.E. Kilmer, P.C. Hansen, and M.I. Espanol. A projection-based approach

- to general-form Tikhonov regularization. *SIAM J. Sci. Comp.*, 29:315–330, 2007.
- [73] M.E. Kilmer and D.P. O’Leary. Choosing regularization parameters in iterative methods for ill-posed problems. *SIAM J. Matrix Anal. Appl.*, 22(4):1204–1221, 2001.
- [74] Eric Chu King-wah. Symmetric solutions of linear matrix equations by matrix decompositions. *Linear Algebra Appl.*, 119:35–50, 1989.
- [75] R. Kumar, C. Da Silva, O. Akalin, A.Y. Aravkin, H. Mansour, B. Recht, and F.J. Herrmann. Efficient matrix completion for seismic data reconstruction. *Geophysics*, 80:97–114, 2015.
- [76] J. Lampe, L. Reichel, and H. Voss. Large-scale Tikhonov regularization via reduction by orthogonal projection. *Linear Algebra Appl.*, 436:2845–2865, 2012.
- [77] A.S. Leonov. On the choice of regularization parameters by means of the quasi-optimality and ratio criteria. *Soviet Math. Dokl.*, 19:537–540, 1978.
- [78] A.S. Leonov. On the accuracy of Tikhonov regularizing algorithms and quasioptimal selection of a regularization parameter. *Soviet Math. Dokl.*, 44:711–716, 1991.
- [79] T.T.Y Lin and F.J. Herrmann. Estimation of primaries by sparse inversion with scattering-based multiple predictions for data with large gaps. *Geophysics*, 81:183–197.
- [80] T.T.Y Lin and F.J. Herrmann. Robust estimation of primaries by sparse inversion via one-norm minimization. *Geophysics*, 78:133–150.
- [81] T.T.Y Lin and F.J. Herrmann. Estimating primaries by sparse inversion in a curvelet-like representation domain. *73rd EAGE Conference and Exhibition 2011*, 2011.
- [82] N.A. Luiken and A. Garg. Block-Krylov methods for multi-dimensional deconvolution. *Society of Exploration Geophysicists*, pages 5070–5074, 2019.
- [83] N.A. Luiken and T. van Leeuwen. Comparing RSVD and Krylov methods for linear inverse problems. *Computers & Geosciences*, 137:104427, 2020.
- [84] N.A. Luiken and T. van Leeuwen. Seismic wavefield redatuming with regularized multi-dimensional deconvolution. *Inverse Problems*, 36:095010, 2020.
- [85] N.A. Luiken and T. van Leeuwen. Relaxed regularization for linear inverse problems. *SIAM J. Sci. Comp.*, Accepted for publication.
- [86] M.A. Lukas. Robust generalized cross-validation for choosing the regularization parameter. *Inverse Problems*, 22(5):1883, 2006.
- [87] M.A. Lukas. Strong robust generalized cross-validation for choosing the regularization parameter. *Inverse Problems*, 24(3):034006, 2008.
- [88] C. L. Mallows. Some comments on Cp. *Technometrics*, 15:661–675, 1973.
- [89] T. Mejer Hansen and K. Mosegaard. Visim: Sequential simulation for linear inverse problems. *Computers & Geosciences*, 34:53–76, 2008.
- [90] M.A. Morozov. *Methods for Solving Incorrectly Posed Problems*. Springer-Verlag, New York, 1984.
- [91] F. Natterer. *The Mathematics of Computerized Tomography*. SIAM, Philadelphia, PA, 2001.

- [92] R. Neupauer and B. Borchers. A MATLAB implementation of the minimum relative entropy method for linear inverse problems. *Computers & Geosciences*, 27:757–762, 2001.
- [93] C.J. Nolan and M. Cheney. Synthetic aperture inversion. *Inverse Problems*, 18(1):221–235, 2002.
- [94] B. O’Donoghue. `apg`. <https://github.com/bodono/apg>, 2016.
- [95] C.C. Paige and M.A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 12:617–629, 1975.
- [96] C.C. Paige and M.A. Saunders. Towards a generalized singular value decomposition. *SIAM J. Numer. Anal.*, 18(3):398–405, 1981.
- [97] C.C. Paige and M.A. Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Software*, 8(1):43–71, 1982.
- [98] T. Planès, R. Snieder, and S. Singh. Model-based redatuming of seismic data: An inverse-filter approach. *Geophysics*, 83(2):1–13, 2018.
- [99] T. Reginska. A regularization parameter in discrete ill-posed problems. *SIAM J. Sci. Comput.*, 17, 1996.
- [100] L. I. Rudin and S. Osher. Total variation based image restoration with free local constraints. In *Proceedings of 1st International Conference on Image Processing*, volume 1, pages 31–35 vol.1, 1994.
- [101] Y. Saad and M.H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci.Stat. Comp.*, 7:856–869, 1986.
- [102] F. Santosa and W.W. Symes. Linear inversion of band-limited reflection seismograms. *SIAM J. Sci.Stat. Comp.*, 7:1307–1330, 1986.
- [103] M. Schmidt, Nicolas L.R., and Francis R.B. Convergence rates of inexact proximal-gradient methods for convex optimization. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 1458–1466. Curran Associates, Inc., 2011.
- [104] G.T. Schuster and M. Zhou. A theoretical overview of model-based and correlation-based redatuming methods. *Geophysics*, 71(4):130–110, 2006.
- [105] L.A. Shepp and B.F. Logan. The Fourier reconstruction of a head section. *IEEE Transactions on Nuclear Science*, 21:21–43, 1974.
- [106] J.L. Starck. Sparsity and inverse problems in astrophysics. *Journal of Physics: Conference Series*, 699:012010, 2016.
- [107] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer Verlag, 1983.
- [108] W.W. Symes. The seismic reflection inverse problem. *Inverse Problems*, 25(12):123008, 2009.
- [109] J. Tamminen. Inverse problems and uncertainty quantification in remote sensing. ESA Earth Observation Summer School on Earth System Monitoring and Modeling, 2012.
- [110] S. Ubaru, J. Chen, and Y. Saad. Fast estimation of $\text{tr}(f(A))$ via stochastic Lanczos quadrature. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1075–1099, 2017.

- [111] E. van den Berg and M.P. Friedlander. Probing the Pareto frontier for basis pursuit solutions. *SIAM J. Sci. Comput.*, 31(2):890–912, 2008.
- [112] J. van der Neut and F.J. Herrmann. Interferometric redatuming by sparse inversion. *Geophysical Journal International*, 192:666–670, 2013.
- [113] J. van der Neut, M. Tatanova, J. Thorbecke, E. Slob, and K. Wapenaar. Controlled source interferometric redatuming by crosscorrelation and multidimensional deconvolution in elastic media. *Geophysics*, 76(4):63–76, 2011.
- [114] J. van der Neut, M. Tatanova, J. Thorbecke, E. Slob, and K. Wapenaar. Deghosting, demultiple, and deblurring in controlled-source seismic interferometry. *International Journal of Geophysics*, 2011:1–28, 2011.
- [115] S. van Huffel and J. Vandewalle. *The total least squares problem: computational aspects and analysis*, volume 9. SIAM, 1991.
- [116] S. Van Huffel and H. Zha. The restricted total least squares problem: Formulation, algorithm, and properties. *SIAM J. Matrix Anal. Appl.*, 12(2):292–309, 2012.
- [117] T. van Leeuwen and A.Y. Aravkin. Non-smooth variable projection. <https://arxiv.org/abs/1601.05011>, 2020.
- [118] J.M. Varah. Pitfalls in the numerical solution of linear ill-posed problems. *SIAM J. Sci.Stat. Comp.*, 4(2):164–176, 1983.
- [119] S. Vatankehah, R.A. Renaut, and V.E. Ardestani. A fast algorithm for regularized focused 3D inversion of gravity data using randomized singular-value decomposition. *Geophysics*, 83(4):25–34, 2018.
- [120] J. Virieux and S. Operto. An overview of Full-Waveform Inversion in exploration geophysics. *Geophysics*, 74:1–26, 2009.
- [121] C.R. Vogel. *Computational Methods for Inverse Problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2002.
- [122] G. Wahba. *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, Philadelphia, 1990.
- [123] C.P.A. Wapenaar. Reciprocity properties of one-way propagators. *Geophysics*, 63(5):1795–1798, 1998.
- [124] C.P.A. Wapenaar, E. Slob, and R. Snieder. Seismic and electromagnetic controlled-source interferometry in dissipative media. *Geophysical Prospecting*, 56:419–434, 2008.
- [125] C.P.A. Wapenaar, J. Thorbecke, J. van der Neut, F. Brogini, E. Slob, and R. Snieder. Marchenko imaging. *Geophysics*, 79(3):WA39–WA57, 2014.
- [126] C.P.A. Wapenaar and J. van der Neut. A representation for Green’s function retrieval by multi-dimensional deconvolution. *The Journal of the Acoustical Society of America*, 128:366–371, 2010.
- [127] Y. Wei, P. Xie, and LP. Zhang. Tikhonov regularization and randomized GSVD. *SIAM J. Matrix Anal. Appl.*, 37:649–675, 2016.
- [128] H. Xiang and J. Zou. Regularization with randomized SVD for large-scale discrete inverse problems. *Inverse Problems*, 29(8):085008, 2013.
- [129] H. Xiang and J. Zou. Randomized algorithms for large-scale inverse problems with general form Tikhonov regularization. *Inverse Problems*, 29:085088, 2015.

- [130] P. Zheng, T. Askham, S.L. Brunton, J.N. Kutz, and A.Y. Aravkin. A Unified Framework for Sparse Relaxed Regularized Regression: SR3. *IEEE Access*, 7:1404–1423, 2019.
- [131] L. Zhi-Pei and P.C. Lauterbur. *Principles of magnetic resonance imaging: a signal processing perspective*. SPIE Optical Engineering Press, 2000.
- [132] A. Zunino and K. Mosegaard. An efficient method to solve large linearizable inverse problems under Gaussian and separability assumptions. *Computers & Geosciences*, 122:77–86, 2019.
- [133] I.N. Zwaan and M.E. Hochstenbach. Multidirectional subspace expansion for one-parameter and multiparameter Tikhonov regularization. *SIAM J. Sci. Comp.*, 70:990–1009, 2017.



Summary



Inverse problems arise in many applications in science and engineering. They are characterized by the fact that directly computing a solution to an inverse problem via a well-defined operator is generally not possible. We have measured data that are generated by an (approximately) known model, the forward model, that depends on some input. This forward model generates simulated data, and the solution to the inverse problem is the input that matches the simulated data and the measured data.

Limitations in the measurement setup, noise in the data, et cetera, add extra difficulty to obtaining a solution. Multiple solutions may lead to roughly the same data, and one has to choose which solution is best. Moreover, we may not want the input to match the data exactly, since we know the data are corrupted.

Selecting the best among many possible solutions is done through a technique called regularization. Regularization is prior information about the solution that we want to incorporate when solving the inverse problem. We now have two elements that we have to rely on when solving the inverse problem: how well the simulated data fits the measured data and how well the solution is in accordance with our prior knowledge. The balance between these two terms is determined by the regularization parameter, which has to be specified by the user. For certain types of regularization there exist parameter selection rules that can be evaluated to get an estimate of the regularization parameter, but evaluating them is as expensive as solving the inverse problem. We generally have to solve the inverse problem for multiple regularization parameters and select the best solution.

On top of that, the type of regularization has a large influence on how the inverse problem is solved. This is due to the fact that we have to use different mathematical tools for different regularization methods.

The goal of this thesis has been to develop fast algorithms for inverse problems and to investigate the estimation of the regularization parameter. We have worked on a

number of different regularization methods for linear inverse problems. For the simplest one, Tikhonov regularization, we have compared two algorithms that can be used to estimate the regularization parameter efficiently. Moreover, we have developed a new algorithm where the dimension of the low-dimensional surrogate model is automatically determined.

We have developed a mathematical framework for a problem arising in geophysics, called Multi-Dimensional Deconvolution. We have discussed the ill-posedness of the problem and show how to incorporate constraints induced by the laws of physics. Moreover, we have discussed additional regularization that is needed to obtain a stable solution.

Finally, we have extended the analysis on SR3, which is a fast algorithm for solving inverse problems with a certain type of regularization. Additionally, we have shown how it may be used to estimate the regularization parameter and proposed a novel implementation to make it suitable for large-scale problems.



Samenvatting



Inverse problemen zijn veelvoorkomend in de wetenschap en techniek. Ze worden gekenmerkt door het feit dat een directe oplossing voor een inverse probleem over het algemeen niet uitgerekend kan worden. Bij een inverse probleem meten we data die gegenereerd zijn door een (deels) bekend model, het voorwaartse model, dat afhangt van een bepaalde variabele. Het voorwaartse model genereert gesimuleerde data, en de oplossing voor het inverse probleem is de variabele die de kleinste fout tussen de gesimuleerde en de gemeten data genereert.

Door, onder andere, beperkingen van de meetinstrumenten en ruis op de data, is het extra moeilijk om een oplossing voor een inverse probleem te vinden. Verschillende oplossingen kunnen nagenoeg dezelfde gesimuleerde data genereren, waardoor we een keuze moeten maken voor een oplossing. Daar komt bij dat we de fout tussen de gesimuleerde data en gemeten data niet willekeurig klein willen hebben, omdat de gemeten data ruis bevat.

Het selecteren van de beste oplossing wordt gedaan door zogeheten regularizatie. Regularizatie is veronderstelde kennis over de oplossing die we willen meewegen in het bepalen van de oplossing. Dit betekent dat we met twee dingen rekening moeten houden in het oplossen van een inverse probleem. Enerzijds moet de gesimuleerde data overeenkomen met de gemeten data, maar anderzijds moet de oplossing voldoen aan onze veronderstelde kennis over de oplossing. De balans tussen deze twee elementen wordt gewogen door de regularizatieparameter, die bepaald moet worden door de gebruiker. Voor bepaalde regularizatie bestaan er methoden om de regularizatieparameter te schatten, maar deze methoden kosten vaak net zo veel rekentijd, zo niet nog meer, als het oplossen van het inverse probleem. Daarom moeten we het inverse probleem vaak meerdere keren oplossen voor verschillende regularizatieparameters, en een keuze maken voor de beste oplossing.

Daar komt bij dat het type regularizatie bepaalt hoe we het inverse probleem kunnen oplossen. Dit komt omdat we voor verschillende regularizaties

verschillende algoritmen gebruiken voor het oplossen van het inverse probleem.

Het doel van dit proefschrift is geweest om snelle algoritmen te ontwikkelen voor inverse problemen en om te onderzoeken hoe de regularisatieparameter geschat kan worden. We hebben gewerkt aan verschillende regularizaties voor lineaire inverse problemen. Voor het meest simpele geval, Tikhonov regularizatie, hebben we twee algoritmen vergeleken die gebruikt kunnen worden om de regularisatieparameter op een efficiënte manier te schatten. Daarnaast hebben we een nieuw algoritme ontwikkeld waar de dimensie van een laag-dimensionaal surrogaat model automatisch wordt bepaald.

We hebben een wiskundig raamwerk opgezet voor een inverse probleem in geofysica, multi-dimensionale deconvolutie. We hebben laten zien dat het probleem slechtgesteld is en hebben laten zien hoe constraints door de wetten van de natuurkunde meegenomen kunnen worden in de optimalisatie. Daarbovenop hebben we laten zien dat er nog extra regularizatie nodig is om het inverse probleem op te lossen.

Ten slotte hebben we de analyse van het SR3 algoritme aanzienlijk uitgebreid. SR3 is een snel algoritme dat wordt gebruikt voor het oplossen van lineaire inverse problemen met bepaalde typen regularizatie. We hebben daarbij laten zien hoe het algoritme gebruikt kan worden om de regularisatieparameter te schatten en we hebben een nieuwe implementatie voorgesteld, die het algoritme geschikt maakt voor grootschalige inverse problemen.



Dankwoord



Ik wil graag deze gelegenheid aangrijpen om een aantal mensen te bedanken voor hun hulp, direct of indirect, bij de totstandkoming van dit proefschrift.

Allereerst wil ik Tristan bedanken voor zijn begeleiding. Tristan, je creativiteit en je vermogen om zaken vanuit een breder perspectief te bekijken zijn erg behulpzaam geweest. Ik kon altijd bij je terecht met vragen en je hebt een grote bijdrage geleverd aan de totstandkoming van dit proefschrift. Bedankt daarvoor! Ik wens je veel succes de komende jaren met je nieuwe baan bij het CWI.

Eric, ik wil je bedanken voor het financieren van mijn PhD positie en je begeleiding de afgelopen 4 jaar. Ondanks dat we elkaar niet wekelijks spraken kon ik altijd bij je terecht voor commentaar, en heb je de aanleiding gegeven voor hoofdstuk 4 en hoofdstuk 5 van dit proefschrift. Ik ben erg onder de indruk van de manier waarop je het consortium leidt en ik bewonder je harde werk. Ik wens je veel succes hierbij in de toekomst.

Rob, bedankt dat je mijn promotor wilde zijn en voor het grondig lezen en corrigeren van een eerste versie van dit proefschrift.

Ik bedank de beoordelingscommissie, Joost Batenburg, Felix Herrmann, Michiel Hochstenbach, Kees Oosterlee, en Kees Vuik voor het lezen en beoordelen van mijn proefschrift.

Matteo, thank you for being part of the defense committee and hiring me as a Postdoc. I look forward to working with you the coming two years.

Ajinkya, you have been an invaluable part of my time as a PhD student. I enjoyed our lunch breaks, coffee breaks, and many discussions about work and other topics. It has been inspiring to always see you strive for the best and setting high standards. I will remember our many, many Indian dinners and hope that many more will follow.

Michiel, bedankt voor onze samenwerking en de plezierige werkbezoeken in Eindhoven. Ik kon goed met je over mijn onderzoek praten en het was prettig om met je van gedachten wisselen. Ik hoop dat we komende jaren nog contact kunnen houden en kunnen samenwerken.

I want to thank my colleagues from the Delphi consortium, Abdulrahman, Ali, Aparajita, Bouchaib, Billy, Dong, Jan-Willem, Gerrie, Gerrit, Hussain, Matteo, Mikhail, Runhai, Shan, Shogo, Shotaro, Siddharth, Sixue for the fun times we had together during the sponsor meetings and conferences. I also want to thank Gerrie for organizing our meetings. I especially want to thank Aayush, who was a co-author on chapter 4 and assisted a lot on chapter 5, and with whom I have spent many dinners and drinks at conferences and when I visited to Delft.

Andre, na mijn verdediging moet jij mij met doctor aanspreken, ongeacht de, ongetwijfeld, hatelijke aanspreekvorm. Ik weet dat ik je de afgelopen vier jaar heb verblijd met mijn gezelschap, en dit was, af en toe, en tot op zekere hoogte, wederzijds. Een mooie herinnering aan de afgelopen tijd is onze rondreis door Amerika, die dankzij mij door kon gaan, omdat ik wél een rijbewijs heb en bereid was overal heen te rijden. Ten slotte, omdat ik van Marion met iets aardigs moet eindigen, zal ik maar toegeven dat ik je erg waardeer als vriend.

Manon, uiteraard kun jij in dit dankwoord niet ontbreken. Ik kan bij jou altijd terecht voor de nodige ontspanning en gezelligheid. Daarnaast help je altijd met uitstekend kledingadvies, met name ook als ik naar een ander werelddeel afreis, en ben je een uitstekende crisismanager gebleken rond het einde van oktober (mede namens Marion, bedankt daarvoor). Je hebt zelf in de afgelopen 4 jaar mooie stappen gemaakt in je carrière en ik ben ervan overtuigd dat er nog veel succes zal volgen. Hier spreekt met recht een trotse broer.

Pap en mam, bedankt voor jullie opvoeding en steun de afgelopen jaren. Zonder jullie was er weinig van dit proefschrift terechtgekomen.

Lieve Marion, het hoogtepunt de afgelopen 4 jaar was jou leren kennen. We hebben elkaar in het begin van mijn periode als PhD-student leren kennen, en mijn mooiste herinneringen aan deze tijd zijn voornamelijk mijn herinneringen aan ons. Je bent mijn steun en toeverlaat geweest en ben je dankbaar voor je vele ongevraagde tips. Ik zie uit naar een mooie toekomst samen.



Curriculum Vitae



Nick Luiken is geboren op 17 oktober 1991 te Hengelo. Na afronding van het Gymnasium aan Lyceum de Grundel in Hengelo, begint hij in 2010 zijn studie Toegepaste Wiskunde aan de Universiteit Twente. In 2014 rondde hij zijn bachelor af, en in 2016 studeerde hij cum laude af voor zijn master.

Van 2017 tot 2021 volgde hij een promotietraject aan de Universiteit Utrecht, onder begeleiding van Tristan van Leeuwen en Eric Verschuur. Het promotietraject werd gefinancierd door het Delphi consortium. In deze periode presenteerde hij zijn werk op conferenties en sponsorbijeenkomsten en heeft hij zijn werk in gerenommeerde vakbladen gepubliceerd.

Vanaf 2021 zal hij aan het werk gaan als Postdoc aan de King Abdullah University of Science and Technology in Thuwal, Saudi-Arabië, onder begeleiding van Matteo Ravasi.

