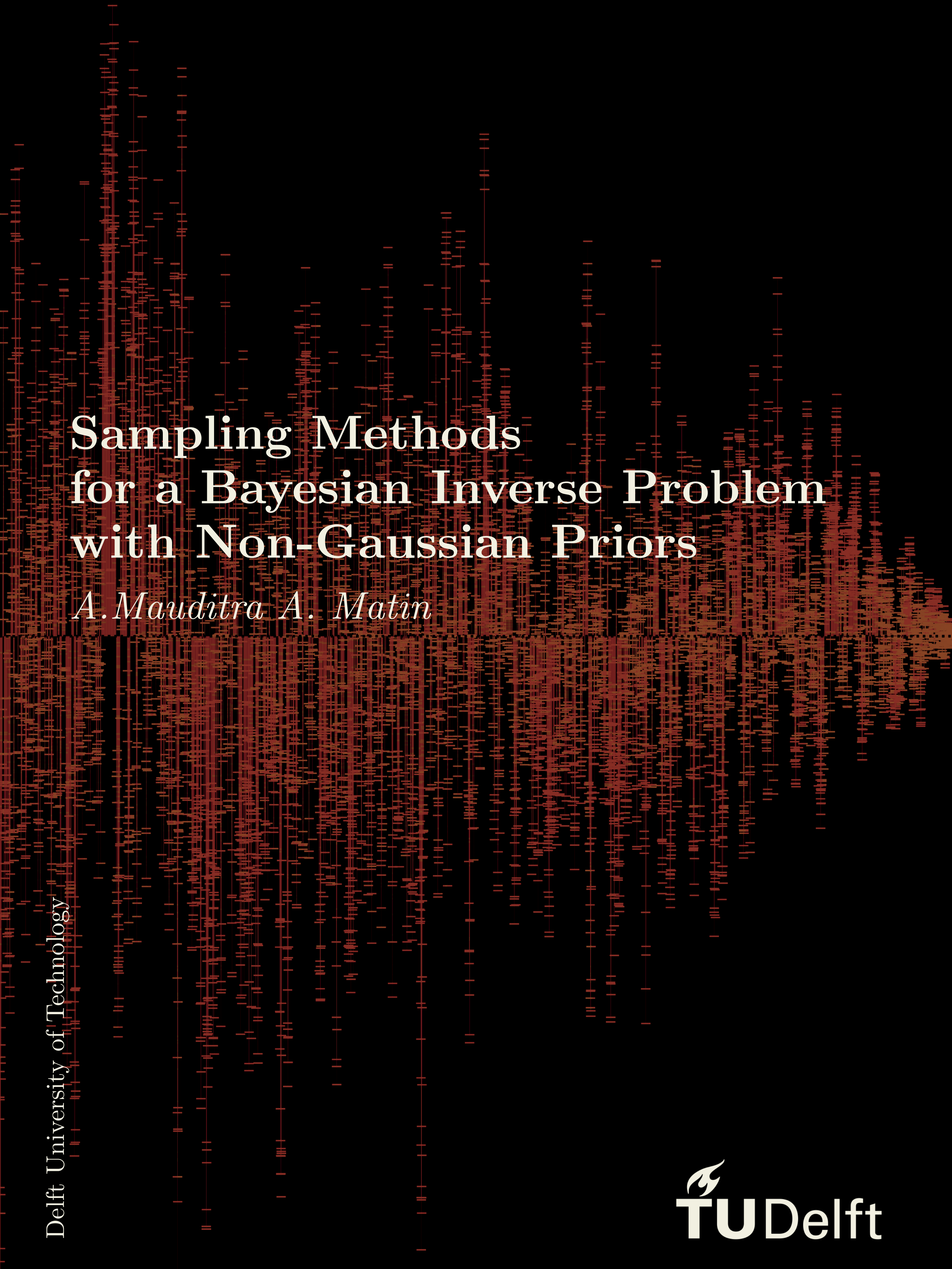


Sampling Methods for a Bayesian Inverse Problem with Non-Gaussian Priors

A.Mauditra A. Matin

Delft University of Technology

Sampling Methods for a Bayesian Inverse Problem with Non-Gaussian Priors

by Alvedian Mauditra Aulia Matin
(Student number: 5689252)

to obtain the degree of Master of Science
at the Delft Institute of Technology,
to be defended publicly on 29 August 2025 at 13:00.

Thesis committee:
Dr. Hanne Kekkonen *(TU Delft, daily supervisor)*
Prof. dr. Martin Verlaan *(TU Delft, responsible supervisor)*
dr. ir. Joris Bierkens

Academic Year: 2024 - 2025

Cover: Plot of ACF between samples obtained using RTO, taken from Figure 5.10e.
An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

Abstract

This thesis explores Markov chain Monte Carlo (MCMC) methods for a discrete linear Bayesian inverse problem with non-Gaussian priors. The non-Gaussian priors are total variation and Besov space priors, which are called edge-preserving due to their ability to model sparse features and discontinuities. Three sampling algorithms are compared: random walk Metropolis-Hastings, preconditioned Crank–Nicholson (pCN), and randomise-then-optimise (RTO). Prior transformations are developed to adapt RW, pCN and RTO for edge-preserving priors. Results show a trade-off between computational efficiency and accuracy. RTO with prior transformations yields more accurate reconstructions and credible intervals, but at significant computational cost and with sensitivity to prior choice. pCN is faster, more robust to discretisation, and provides more control over the sampling process but produces highly correlated samples and less accurate estimates.

Acknowledgements

The process of writing this thesis taught me many valuable lessons, not all of them in mathematics. Now that this process is at its end, I would like to express my gratitude to the people who have guided me through it.

I would like to thank my daily supervisor, Hanne Kekkonen, for her thoughtful supervision throughout this project. During our meetings, you have shared your knowledge with the kind of generosity and enthusiasm any student would be lucky to find in a mentor. Your thorough feedback on the drafts of this report has also taught me valuable lessons about clarity in writing. I would also like to thank my responsible supervisor, Martin Verlaan, as key aspects of this thesis were inspired by the internship project I worked on under his supervision. I chose to work with the library `JAX` in this thesis, which shares many features with `Julia`. You introduced me to `Julia` and opened my eyes to many interesting developments in scientific computing.

Thank you to the fellow students whose support I've relied on throughout this thesis project. I am grateful for your warmth, encouragement, and friendship. I will always cherish the memories we've made together. To Lia Kondova, Rodrigo Carvalho, and Mirte Brouwer from the TU Delft Writing Centre, thank you for all the illuminating discussions and productivity boosts. In particular, I would like to thank my thesis brother Taras Budko, whose whimsy and wisdom gave me strength at many crucial moments. And my most heartfelt gratitude goes to Niels Goedegebure, with whom I have shared this entire Master's journey. Near or far, I always know I can turn to you for much-needed perspective and fuel for the spirit.

To my husband and partner, Daffa Sudrajat, I could not possibly thank you enough for bringing serenity to my many chaotic days, for your loving encouragement, and for always sharing my excitement. Lastly, thank you to my parents, Retno Audite Dwiastini and Isandra Matin Ahmad, for having faith in me.

Alvedian Mauditra Aulia Matin
Delft, August 2025

Frequently-used notation

Random variables are denoted by uppercase letters. Random values taking values in function spaces are denoted by italicised uppercase letters (e.g. U, Y). Random variables taking values in $\mathbb{R}^n$, with $n \in \mathbb{N}$, are denoted by boldface uppercase letters (e.g. $\mathbf{U}, \mathbf{Y}$). Realisations of function-valued random variables are denoted by italicised lowercase letters (e.g. u, y). Realisations of $\mathbb{R}^n$-valued random variables, and $\mathbb{R}^n$-valued vectors in general, are denoted by boldface lowercase letters (e.g. $\mathbf{u}, \mathbf{y}$). To refer to the j-th element of a vector $\mathbf{v}$, we write $\mathbf{v}_j$.

Matrices in $\mathbb{R}^{m \times n}$ with $m, n \in \mathbb{N}$ are denoted by boldface uppercase letters (e.g. $\mathbf{L}, \mathbf{D}$). To avoid confusion with $\mathbb{R}^n$-valued random variables, we will always state that matrices are in $\mathbb{R}^{m \times n}$ explicitly. Operators on Banach spaces are denoted by calligraphic uppercase letters (e.g. $\mathcal{A}, \mathcal{F}$).

Unless otherwise specified, μ denotes a measure and π a density function. Subscripts are used to distinguish between different measures and different density functions. A sigma algebra over a set B is written as $\sigma(B)$.

Contents

Chapter 1

Introduction

Image deconvolution, the process of denoising and sharpening images captured by imaging devices, such as light microscopes or telescopes, is an important problem in imaging. In many application areas, such as medicine [1], biology [2], astronomy [3], deconvolution is a necessary post-processing step to reduce the distorting effects of noise and blur, which are unavoidable impacts of optics and electronics. Mitigating the impact of noise and blur yields clearer images. Aside from images, deconvolution is also used to process one-dimensional signals. In seismology, it is used to process seismic trace signals and other seismic data [4], [5]. In astronomy, is it used to analyse the characteristics of stars from stellar intensity and polarisation spectra [6]. In chemistry, one-dimensional chromatographic analysis makes use of deconvolution to detect chemical components [7], [8]. Deconvolution is an example of an inverse problem, which is a problem of finding an unknown quantity when we have a noisy measurement and knowledge of process through which the measurements were obtained.

Inverse problems are difficult to solve because they are ill-posed [9], [10]. These problems fail to fulfill at least one of Hadamard's well-posedness conditions [11], [12]. These conditions check whether the formulation of a problem leads to a solution that exists, is unique, and is continuously dependent on the data. The ill-posedness of inverse problems is typically addressed through regularisation, which imposes penalties on specific function characteristics, such as smoothness or sparsity with respect to a function space basis. The standard approach to regularising inverse problems is Tikhonov regularisation [9], [13]–[15], which stabilizes the solution by penalizing extreme or implausible parameter values. It balances fidelity to the observed data with smoothness or simplicity in the solution.

Alternatively, the ill-posedness of an inverse problem can be addressed by viewing the problem as one of Bayesian statistical inference. This follows the work of [10], where the Bayesian view was taken for inverse problems with finite-dimensional components. The work of [16] takes the Bayesian view of inverse problems in general Banach spaces, including function spaces. This approach leads to systematic quantification of uncertainty about the unknown. Ill-posedness is addressed by modeling all the elements of the inverse problem as quantities with statistical properties. The unknown is modeled as a random variable with a prior probability density function. The prior is constructed based on our *a priori* knowledge about the unknown. The data misfit is quantified following the distribution of the noise. The solution of a Bayesian inverse problem is the posterior distribution which is the conditional probability measure of the unknown given the observed data, obtained by combining the prior and the likelihood via Bayes' rule. Tikhonov regularisation can be interpreted as a Bayesian approach where the unknown parameters have a Gaussian prior and the measurement noise is Gaussian. The quadratic regularization term corresponds to the negative log of the likelihood (also known as the potential), and the solution is the maximum a posteriori estimate of the posterior distribution.

Generally, the posterior distribution cannot be sampled directly. To approximate posterior statistics, indirect sampling techniques are often used. These techniques include Markov chain Monte Carlo (MCMC) methods, which obtain samples by constructing a Markov chain whose stationary distribution is equal to the target posterior. MCMC algorithms are commonly built in Metropolis-Hastings [17], [18] or Gibbs [19] frameworks. Gibbs samplers have been applied to large-scale nonlinear inverse problems [20]. Adaptive Metropolis methods have been applied to low- to moderate-dimensional settings [21], [22]. Gradient-informed Metropolis-adjusted Langevin algorithms (MALA) [23] improve efficiency for large-scale problems, and Hamiltonian Monte Carlo (HMC) [24] incorporates higher-order geometry into sampling. These samplers were originally designed to draw samples from posterior distributions in which the prior over the unknown is assumed to follow a Gaussian distribution.

While Gaussian priors are convenient and lead to tractable posterior structures, they may fail to capture important characteristics of many real-world problems, such as sharp discontinuities or edges. These characteristics are particularly significant in deconvolution, as sharp discontinuities usually mark the boundaries between distinct objects in images. Capturing these discontinuities is key to obtaining clear, deconvolved images. This thesis thus concentrates on non-Gaussian prior models, with a particular emphasis on edge-preserving priors [25]–[27], which are specifically designed to maintain sharp transitions in the reconstructed quantities while reducing the effects of noise.

In this thesis, we compare the results of modifications to Markov Chain Monte Carlo (MCMC) methods that enable their use in solving Bayesian inverse problems with edge-preserving priors. Previous work has introduced sampling strategies tailored for inverse problems [28], [29], which improve sampling efficiency by exploiting the structure of the posterior distribution arising from Bayesian inversion with Gaussian prior and Gaussian noise. The first of these, the preconditioned Crank–Nicholson (pCN) method, extends the classical random walk Metropolis–Hastings algorithm to function spaces. In related studies [30], [31], non-Gaussian priors were transformed into Gaussian random variables to facilitate this approach. The second method, randomise-then-optimise (RTO), leverages optimisation techniques to efficiently generate samples from high-dimensional probability distributions. For example, in [31], the total variation prior was transformed into a Gaussian random variable to solve a one-dimensional deconvolution problem.

We test the transformed MCMC methods on one-dimensional deconvolution. In one-dimensional deconvolution, we aim to retrieve a piecewise continuous function (or true signal) defined over an interval in $\mathbb{R}$ from a convolved signal. The image deconvolution problem is a two-dimensional version of this problem. As one-dimensional deconvolution is simpler, it is relatively straightforward to compare the true signal to estimated point estimators and credible intervals by visual inspection. It is also a problem that allows us to study the effect of dimension size, as we can refine the discretisation of the piecewise continuous function to examine the effects on the sampling methods and the solutions obtained.

In this thesis, we aim to address two main questions. The questions are as follows.

1. What are the relative strengths and limitations of the preconditioned Crank–Nicholson method and the randomize-then-optimize approach when applied to sampling from posterior distributions arising in Bayesian inversion with edge-preserving priors?
2. How does the dimension of the discrete problem affect the performance of the sampling algorithm?

This thesis is structured as follows. In Chapter 1, we provide a brief introduction to the problem of sampling with non-Gaussian priors. In Chapter 2, preliminaries on inverse problems and the Bayesian approach, including prior modeling, are introduced. One-dimensional deconvolution is given as a running example in Chapter 2 to illustrate the concepts to the reader. A brief overview of basic concepts in Markov chain Monte Carlo (MCMC) methods is provided in Chapter 3, followed by a presentation of the MCMC methods used in this thesis. In Chapter 4, the Bayesian solution for the one-dimensional deconvolution problem is derived. Prior transformations and algorithms for sampling from the posterior distribution are also presented in Chapter 4. Sampling results are presented in Chapter 5. In Chapter 6, conclusions are presented along with directions for further work and avenues of improvement.

Chapter 2

Preliminaries on Bayesian inverse problems

This thesis explores sampling methods for a discrete linear Bayesian inverse problem. In this chapter, we present a short introduction to inverse problems in Section 2.1 and highlight the classical notion of a solution in Subsection 2.1.2. The Bayesian approach and solution is introduced in Section 2.2. A key element in the Bayesian approach, which also affects the sampling methods studied in this thesis, is how prior information about the problem is incorporated in the problem-solving process. In Section 2.3, three ways to do this for the one-dimensional deconvolution problem are presented.

2.1 Inverse problems

A problem arises in situations where we would like to study an object or quantity that we cannot measure directly. For example, we wish to recover a de-blurred, de-noised picture from the blurry, noisy picture in Fig. 2.1.

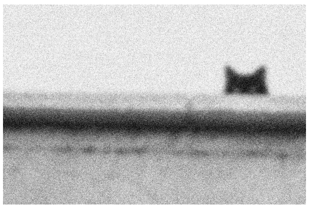


Figure 2.1: A blurred and noisy picture.

To approach this problem mathematically, we first introduce notation for the linear measurement model

$$\mathbf{y} = \mathcal{A}u + \mathbf{e}, \tag{2.1}$$

where $\mathbf{y} \in \mathbb{R}^m$ is our measurement or observation, the linear operator $\mathcal{A} : B_u \to \mathbb{R}^m$ is our forward operator, $\mathbf{e} \in \mathbb{R}^m$ and u is a member of the Banach space B_u. The picture in Fig. 2.1 is our *measurement* or *observation*. The de-blurred picture is our *unknown*. Suppose we have knowledge of the process that generated this blurred image. The mathematical model of this process is the *forward operator*. Aside from the blurriness of Fig. 2.1, we also have to consider the *noise* in the picture.

Measurement noise cannot be avoided in practical situations, and errors due to measurement noise are modeled by the vector $\mathbf{e}$. In the linear measurement model (2.1), this is a deterministic but unknown quantity. This approach to noise does not rule out the possibility that $\mathbf{e}$ is a realisation of a random process, which is an accurate model of noise in many types of measurements.

The reader may notice that u may be in any Banach space B_u, including spaces of functions, but the observation $\mathbf{y}$ is discrete. This is because the quantities we cannot directly measure can often be functions—for example, u may represent the initial temperature distribution of a metal rod or a piecewise continuous function. Meanwhile, observations of physical quantities are taken using measurement devices, which store vectors of numbers. This is why the observation $\mathbf{y}$ is a member of a finite-dimensional vector space. Additionally, quantities in general Banach spaces must be mapped to quantities in finite-dimensional spaces (or discretised) for computation purposes.

In terms of the linear measurement model (2.1), the *direct problem* is,

$$\text{"Given } u\text{, find } \mathbf{y}\text{."} \tag{2.2}$$

Noise has to be considered when dealing with measurements, but are not necessarily present in direct problems. Direct problems are usually well-posed, meaning that they are formulated in a way that leads to a meaningful solution. In his study of partial differential equations, Hadamard [11], [12] proposed that a problem is considered well-posed if it fulfills three conditions, which are stated below.

Definition 2.1.1: Well-posedness conditions

A problem is considered well-posed if it satisfies the following conditions.

(H_1) There is at least one solution. *(Existence)* (2.3)

(H_2) There is at most one solution. *(Uniqueness)* (2.4)

(H_3) The solution depends continuously on the data. *(Continuous dependence)* (2.5)

The *linear inverse problem* that corresponds to the direct problem (2.2) is,

$$\text{"Given a noisy measurement } \mathbf{y} = \mathcal{A}u + \mathbf{e}\text{, extract information about } u\text{."} \tag{2.6}$$

In the case where the space B_u is $\mathbb{R}^n$, we have a *discrete linear inverse problem*,

$$\text{"Given a noisy measurement } \mathbf{y} = \mathbf{A}\mathbf{u} + \mathbf{e}\text{, extract information about } \mathbf{u}\text{."} \tag{2.7}$$

Here, the forward operator is multiplication of the matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ with $\mathbf{u} \in \mathbb{R}^n$. Strictly speaking, the observation $\mathbf{y}$ in (2.6) is not the same as the observation $\mathbf{y}$ in (2.7), as the two problems involve different models. We denote both observations by $\mathbf{y}$ to streamline notation. The problems (2.6) and (2.7) are ill-posed, meaning that they do not satisfy the well-posedness conditions.

For the discussions of ill-posedness, it is useful to introduce notation for the ideal (noiseless) measurement corresponding to the problem (2.7), given by

$$\mathbf{y}_0 = \mathbf{A}\mathbf{u}. \tag{2.8}$$

The different formulations presented in (2.6) and (2.7) can represent different approaches to solving an inverse problem where u is a function. The linear inverse problem formulation (2.6) is used when discretisation of u is left for last and the discrete linear inverse problem (2.7) is used when the discretisation of u occurs at the beginning of the problem-solving process. Choosing one formulation impacts how we approach the inverse problem. In this thesis, we focus on the formulation (2.7) for a discrete one-dimensional deconvolution problem.

2.1.1 Ill-posedness of discrete linear inverse problems

A problem that fails to fulfill one or more of the conditions in Definition 2.1.1 is ill-posed. Inverse problems, which are characterised by their sensitivity to errors due to measurement noise and model errors, are ill-posed.

The conditions in Definition 2.1.1 can be broken in several ways by a problem of the form (2.7). Consider a situation where $m > n$; in other words, the dimensions of the measurement exceed those of the unknown. Even if there is a $\mathbf{u} \in \mathbb{R}^n$ such that $\mathbf{A}\mathbf{u} = \mathbf{y}_0$, the addition of noise leads to a system with no solutions, as the system (2.8) is overdetermined. This breaks condition (2.3). If $n > m$, the linear system in (2.6) is underdetermined, and several elements of $\mathbb{R}^n$ may be solutions of the system. Therefore, condition (2.4) is broken.

To see how condition (2.5)can be broken, we consider the case where $m = n$. To measure ill-posedness, the condition number of $\mathbf{A}$ can be computed. A definition of the condition number is given below [32].

Definition 2.1.2: Condition number

Let $\mathbf{A}$ be a matrix in $\mathbb{R}^{n\times n}$. The condition number of $\mathbf{A}$ is

$$\text{cond}(\mathbf{A}) = ||\mathbf{A}||||\mathbf{A}^{-1}||$$

with $||\cdot||$ denoting any matrix norm. In this thesis, we compute the condition number with the 2-norm, $||\cdot||_2$.

A value of $\text{cond}(\mathbf{A})$ that is much larger than 1 indicates that the problem of solving for $\mathbf{u}$ in $\mathbf{A}\mathbf{u} = \mathbf{y}$ is highly sensitive to small measurement errors [32]. Attempting to solve the problem (2.7) by multiplying both sides of the linear equation with $\mathbf{A}^{-1}$ would result in $\mathbf{A}^{-1}\mathbf{e}$ dominating the solution. The continuous dependence condition (2.5) would then be broken. This is illustrated in Example 2.1.3 using naive deconvolution.

Before discussing naive deconvolution, we first introduce continuous and discrete convolution. Discrete convolution is the direct problem that corresponds to discrete deconvolution, and discrete convolution is an approximation of continuous convolution.

Example 2.1.1 (Continuous convolution)**.** In seismology, sensors often capture convolved signals due to the Earth's impulse response. The effect introduced by the sensors and the Earth's impulse response is modeled mathematically by *convolution.*

In this project, we discuss the convolution of piecewise continuous functions u which are compactly supported on $[0, 1)$. By restricting our scope to functions compactly supported on $[0, 1)$, we can focus on the ill-posedness of the inverse problem over any difficulties we may encounter by having to extend u at the boundary of its domain.

Blurring of a function u (sometimes also referred to as a signal) is modeled by taking the convolution of the function with a point spread function. The point spread function (PSF) ρ is sometimes called the convolution kernel, device function, impulse response, blurring kernel, or transfer function in other fields.

Definition 2.1.3: Point spread function

A point spread function ρ is a non-negative function that satisfies the following conditions:

$$\int_{\mathbb{R}} x \cdot \rho(x)\, dx = 0 \text{ and} \tag{2.9}$$

$$\int_{\mathbb{R}} \rho(x)\, dx = 1. \tag{2.10}$$

Condition (2.9) ensures the function is centered around zero. Condition (2.10) means that constant functions are unchanged when they are convolved with ρ. Thus, the scale of any function u is preserved when it is convolved with ρ. Additionally, we often want to work with even functions, or $\rho(-x) = \rho(x)$. We define three point spread functions that fulfill conditions (2.9) and (2.10): the triangle PSF ρ_T, the quartic PSF ρ_Q, and the Gaussian PSF ρ_G. Each function can be parameterised by $a > 0$. They are defined as

$$\rho_T(x) = \frac{1}{a^2}(a - |x|) \quad \text{for } x \in [-a, a], \tag{2.11}$$

$$\rho_Q(x) = \frac{15}{16a^5}(x - a^2)^2 \quad \text{for } x \in [-a, a], \text{ and} \tag{2.12}$$

$$\rho_G(x) = \frac{1}{\sqrt{2\pi a^2}} \exp\left(-\frac{x^2}{2a^2}\right) \quad \text{for } x \in \mathbb{R}. \tag{2.13}$$

We now define what it means to convolve a function u with a PSF.

Definition 2.1.4: Continuum model of 1D convolution

Let C_{pc} be the space of piecewise continuous functions compactly supported on $[0, 1)$. We fix a point spread function ρ as in Definition 2.1.3. For any $u \in C_{pc}$, the convolution of u and ρ is

$$(\rho * u)(x) = \int_{\mathbb{R}} \rho(\tau) u(x - \tau)\, d\tau = \int_{\mathbb{R}} \rho(x - \tau) u(\tau)\, d\tau. \tag{2.14}$$

The equality in (2.14) is due to the translation invariance of the integral over $\mathbb{R}$.

Example 2.1.2 (Discrete convolution)**.** We now want to define a discrete counterpart to the continuous convolution defined in (2.14). To do so, we need to discretise the domain, the integral operator, and the PSF.

To discretise the domain $[0, 1)$, we define

$$\Delta x = \frac{1}{n} \tag{2.15}$$

$$\mathbf{x}_j = j\Delta x \quad \forall j \in 0, 1, \dots, n-1 \tag{2.16}$$

and call $\mathbf{x}_j$ our *grid points*. The discretisation of a piecewise continuous function u on our grid is

$$\mathbf{u} = \begin{pmatrix} u(\mathbf{x}_0) & u(\mathbf{x}_1) & \cdots & u(\mathbf{x}_n) \end{pmatrix}^\intercal .$$

We can approximate the integral of a piecewise continuous function u over an interval $[b_0, b_1]$ by numerical quadrature. Our discrete approximation of this integral is

$$\int_{b_0}^{b_1} u(x)dx \approx \Delta x \sum_{j=0}^{n-1} u(\mathbf{x}_j) = \Delta x \sum_{j=0}^{n-1} \mathbf{u}_j . \tag{2.17}$$

Next, we define the support of our discrete PSF in the form of an interval $[-a_\rho, a_\rho]$ with $a_\rho > 0$. The triangle PSF (2.11) and quartic PSF (2.12) are zero outside $[-a, a]$, so for these PSFs we can choose $a_\rho = a$. For the Gaussian PSF, we can choose $a_\rho = 6a$, as the Gaussian PSF (2.13) is close to zero for $|x| > [-6a, 6a]$. The support of the discrete PSF is an interval centered around zero, as we work with PSFs that are symmetric and centered around zero.

Let ν be the largest integer such that $\nu \leq \frac{a_\rho}{\Delta x} < \nu + 1$ and define our discrete PSF $\hat{\mathbf{p}}$ as

$$\hat{\mathbf{p}} = \frac{1}{\left(\Delta x \sum_{j=-\nu}^{\nu} \rho(j\Delta x)\right)} \begin{pmatrix} \rho(-\nu\Delta x) & \rho((-\nu+1)\Delta x) & \cdots & \rho((\nu-1)\Delta x) & \rho(\nu\Delta x) \end{pmatrix}^\intercal . \tag{2.18}$$

This ensures

$$\Delta x \sum_{j=-\nu}^{\nu} \hat{\mathbf{p}}_j = 1$$

which corresponds to the normalisation condition (2.10). The discrete PSF $\hat{\mathbf{p}}$ is thus also scale-preserving.

We first define our approximation of $(\rho * u)(\mathbf{x}_j)$. Substituting the convolution (2.14) into our quadrature rule (2.17) yields the approximation

$$\int_{-a_\rho}^{a_\rho} \rho(y) u(\mathbf{x}_j - y)\, dy \approx \Delta x \sum_{l=-\nu}^{\nu} \hat{\mathbf{p}}_l \cdot u(\mathbf{x}_j - \mathbf{x}_l) = \Delta x \sum_{l=-\nu}^{\nu} \hat{\mathbf{p}}_l \cdot \mathbf{u}_{j-l} . \tag{2.19}$$

As u is compactly supported on $[0, 1)$, we can extend the function by considering it to be periodic. Hence $\mathbf{u}_{j-l} = u(\mathbf{x}_{j-l}) = 0$ whenever $j - l < 0$ and $j - l > n - 1$ for $j \in \{0, 1, \ldots, n-1\}$ and $l \in \{-\nu, \nu + 1, \ldots, \nu - 1, \nu\}$. By substituting values of extended u into (2.19), we can define a discrete convolution operation.

Definition 2.1.5: Discrete convolution

Let $\mathbf{p} = \Delta x \cdot \hat{\mathbf{p}}$ with $\hat{\mathbf{p}}$ as defined in (2.18). The discrete convolution of $\mathbf{u}$ and $\mathbf{p}$ at $\mathbf{x}_j$ is

$$\sum_{l=-\nu}^{\nu} \mathbf{p}_l \cdot \mathbf{u}_{j-l} \tag{2.20}$$

for $j \in \{0, 1, \ldots, n-1\}$.

Let ρ be the quartic PSF (2.12) with $a = 0.04$ and let our grid have $n = 64$ points. Then $a_\rho = 0.04$ and

$\nu = \lfloor 0.04 \cdot 64 \rfloor = 2$. Let $\mathbf{p}$ be as in Definition 2.1.5 and define $\mathbf{A}_{64}\mathbb{R}^{64\times 64}$ as

$$\mathbf{A}_{64} = \begin{pmatrix} \mathbf{p}_0 & \mathbf{p}_{-1} & \mathbf{p}_{-2} & 0 & \cdots & \cdots & \cdots & \mathbf{p}_2 & \mathbf{p}_1 \\ \mathbf{p}_1 & \mathbf{p}_0 & \mathbf{p}_{-1} & \mathbf{p}_{-2} & 0 & & & & \mathbf{p}_2 \\ \mathbf{p}_2 & \mathbf{p}_1 & \mathbf{p}_0 & \mathbf{p}_{-1} & \mathbf{p}_{-2} & 0 & & & \vdots \\ 0 & & & & \ddots & \ddots & & & \\ \vdots & & & & \ddots & & & & \\ \mathbf{p}_{-2} & & & & 0 & \mathbf{p}_2 & \mathbf{p}_1 & \mathbf{p}_0 & \mathbf{p}_{-1} \\ \mathbf{p}_{-1} & \mathbf{p}_{-2} & \cdots & \cdots & \cdots & 0 & \mathbf{p}_2 & \mathbf{p}_1 & \mathbf{p}_0 \end{pmatrix}. \tag{2.21}$$

If we compute $\mathbf{g} = \mathbf{A}_{64}\mathbf{u}$, each j-th element of $\mathbf{g}_j$ will be the sum (2.20).

We perform discrete convolution on a piecewise continuous function and plot the results in Figure 2.2.

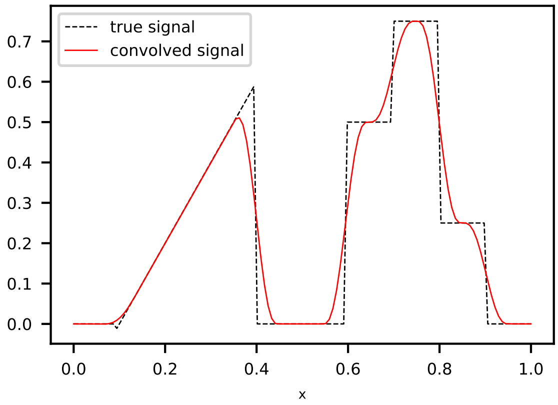
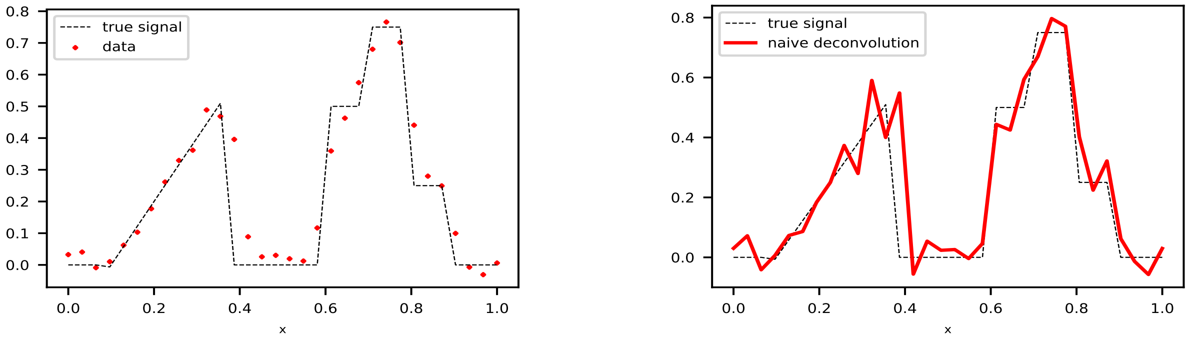


Figure 2.2: A piecewise continuous function on the domain $[0, 1]$ (in black) and the convolved function (in red).

We now illustrate how ill-posedness manifests numerically in 1D deconvolution.

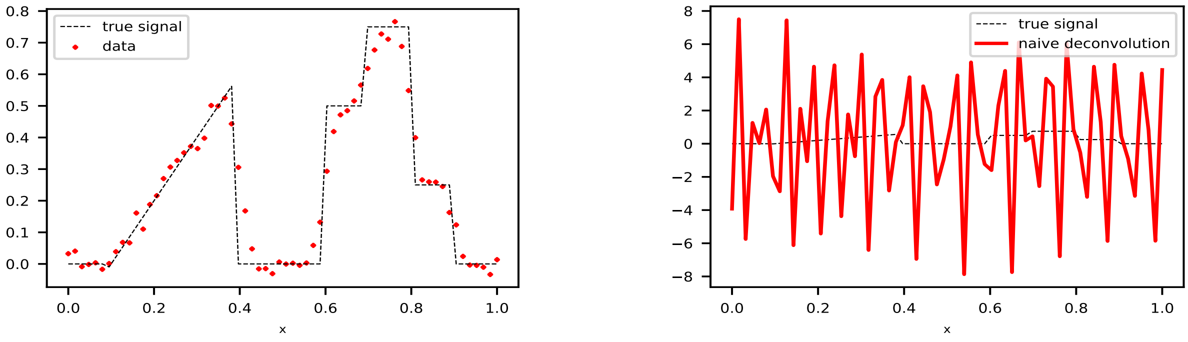
Example 2.1.3 (Naive deconvolution)**.** Naive deconvolution entails multiplying the inverse of the discrete convolution matrix (2.21) with the observation $\mathbf{y}$. We perform naive deconvolution to try to recover the true signal from a synthetic observation $\mathbf{y}$ of a noisy, convolved piecewise continuous function.



(a) Observations and true signal for $n = 32$.

(b) Naive deconvolution for $n = 32$

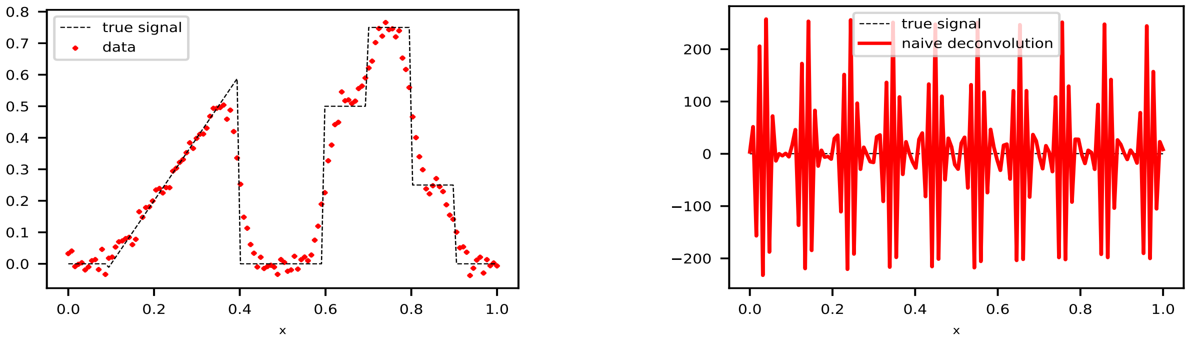
Figure 2.3: Naive deconvolution for $n = 32$, $\mathrm{cond}(\mathbf{A}_{32}) = 5.221$



(a) Observations and true signal for $n = 64$.

(b) Naive deconvolution result for $n = 64$

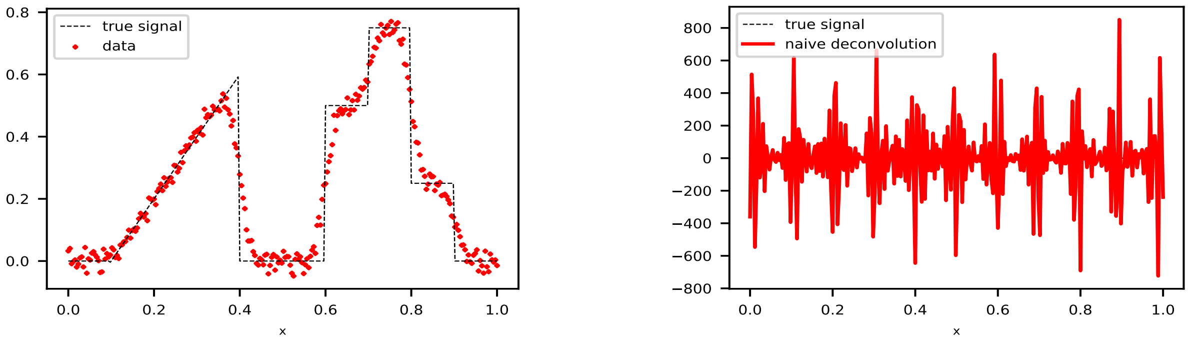
Figure 2.4: Naive deconvolution for $n = 64$, $\mathrm{cond}(\mathbf{A}_{64}) = 2242.303$



(a) Observations and true signal for $n = 128$.

(b) Naive deconvolution result for $n = 128$

Figure 2.5: Naive deconvolution for $n = 128$, $\mathrm{cond}(\mathbf{A}_{128}) = 34179.688$



(a) Observations and true signal for $n = 256$.

(b) Naive deconvolution result for $n = 256$

Figure 2.6: Naive deconvolution for $n = 256$, $\mathrm{cond}(\mathbf{A}_{256}) = 1281190.0$

As n increases, the condition number of $\mathbf{A}_n$ increases rapidly. The results of naive deconvolution bear little resemblance to the true signal, marked in black lines.

2.1.2 Tikhonov regularisation

One way to approach the ill-posedness discussed in Subsection 2.1.1 is through Tikhonov regularisation [14], [15]. Tikhonov regularisation is discussed in detail in [9]. In Tikhonov regularisation, the ill-posedness in the inverse problem is mitigated by finding an approximate solution subject to some requirements. To illustrate the concepts involved, we discuss Tikhonov regularisation for inverse problems of the form (2.7). The Tikhonov functional [9] is given by

$$T_\alpha(\mathbf{u}) = ||\mathbf{A}\mathbf{u} - \mathbf{y}||_2^2 + \alpha||\mathbf{L}\mathbf{u}||_2^2. \tag{2.22}$$

The regularisation parameter α is a value that we choose and $\mathbf{L}$ is in $\mathbb{R}^{n\times n}$. The matrix $\mathbf{L}$ can be constructed using prior knowledge of our solution. The Tikhonov regularised solution $\mathbf{u}_\alpha$ is

$$\mathbf{u}_\alpha = \operatorname*{argmin}_{\mathbf{u}\in\mathbb{R}^n} T_\alpha(\mathbf{u}). \tag{2.23}$$

The parameter α is the called the regularisation parameter. The minimisation problem (2.23) can be interpreted as a balance between two requirements. First, we would like our solution to fit our forward model and observations, meaning that the residual $||\mathbf{A}\mathbf{u} - \mathbf{y}||_2^2$ is small. Second, we would like our solution to be stable, meaning we want $\alpha||\mathbf{L}\mathbf{u}||_2^2$ to be small in norm.

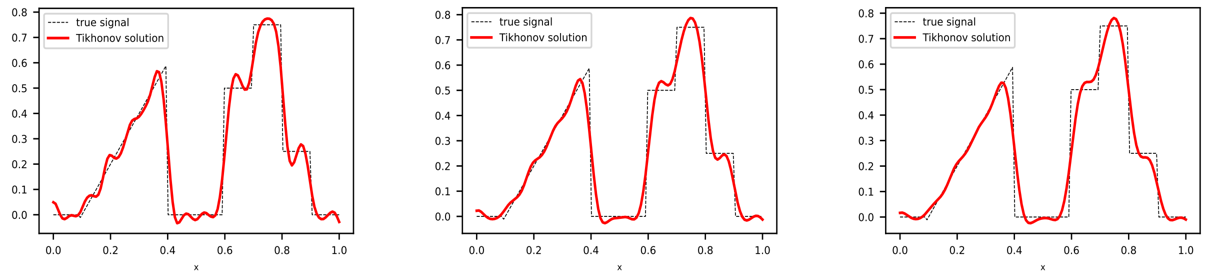
Example 2.1.4 (Tikhonov solution of 1D deconvolution)**.** We set

$$\mathbf{D} = \frac{1}{\Delta x}\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & 1 \\ -1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{pmatrix}. \tag{2.24}$$

This matrix will henceforth be referred to as the *difference matrix.* The vector $\mathbf{u}$ is a discrete approximation of a piecewise continuous function u, and the product $\mathbf{D}\mathbf{u}$ is a discrete approximation of the first derivative of u. For $j = 1, \dots, n-1$ the j-th element of $\mathbf{D}\mathbf{u}$ is

$$\frac{1}{\Delta x}(\mathbf{u}_j - \mathbf{u}_{j-1}),$$

which is a finite difference approximation of $u'(\mathbf{x}_j)$. We set the regularisation matrix in the functional (2.22) as $\mathbf{D}$, meaning $\mathbf{L} = \mathbf{D}$. Keeping the term $\alpha||\mathbf{D}\mathbf{u}||_2^2$ small in norm reflects an *a priori* belief that the function u is continuous, and hence its first derivative will be small in norm. We present Tikhonov solutions for the deconvolution problem with a few different values of α. A larger α value leads to a smoother solution.



(a) Tikhonov regularised solution with matrix **D** and $\alpha = 0.1$.

(b) Tikhonov regularised solution with matrix **D** and $\alpha = 0.5$.

(c) Tikhonov regularised solution with matrix **D** and $\alpha = 1.0$.

Figure 2.7: Tikhonov solutions for the 1D deconvolution problem with varying α values.

As seen in Figure 2.7, the Tikhonov regularisation approach can give us relatively accurate approximations of the true signal. However, uncertainty about the true signal is not easy to quantify using this approach. In linear inverse problems found in practice, such as image deconvolution, we will not be able to compare our solution to the ground truth, which makes uncertainty quantification valuable. This leads us to the approach taken in this thesis: the Bayesian approach to inverse problems.

2.2 The Bayesian approach to inverse problems

In the previous section, Tikhonov regularisation was applied to find a numerical solution for an ill-posed inverse problem. The linear measurement model (2.1) is deterministic, meaning that statistical properties are not modeled in the problem formulation. In this section, the inverse problem is restated as one of statistical inference. When this approach is taken, the elements of our inverse problem—the observation, noise, and unknown—are modeled as quantities with statistical properties. Uncertainty quantification of the solution can be performed [10] once it is obtained. Important definitions in probability theory are given in Appendix A and the reader may find these useful in the discussions that follow.

Consider a linear measurement model where the quantities are modeled as random variables. Let Ω_1 and Ω_2 be sample spaces. Let $\Omega = \Omega_1 \times \Omega_2$. We consider the linear measurement model

$$\mathbf{Y} = \mathbf{A}\mathbf{U} + \mathbf{E} \tag{2.25}$$

where $\mathbf{U} : \Omega_1 \to \mathbb{R}^n$, $\mathbf{E} : \Omega_2 \to \mathbb{R}^m$, and $\mathbf{Y} : \Omega \to \mathbb{R}^m$.

The linear inverse problem corresponding to (2.25) is,

$$\text{"Approximate } \mathbf{U} \text{ when an observation } \mathbf{y} \text{ is given."} \tag{2.26}$$

In other words, we would like to condition $\mathbf{U}$ on a single realisation of $\mathbf{Y}$. For brevity, we denote conditioning a random variable on a realisation of another random variable using the $|$ symbol (e.g. $\mathbf{U} \,|\, \mathbf{y}$ denotes $\mathbf{U}$ conditioned on a single realisation of $\mathbf{Y}$.)

In this measurement model, $\mathbf{U}$ is a random variable, and we can choose its distribution to reflect what we know about $\mathbf{U}$ before obtaining an observation. Larger probabilities are assigned to values in $\mathbb{R}^n$ that we consider more likely based on our *a priori* information. The distribution constructed following this principle is the *prior distribution* of $\mathbf{U}$. Using Bayes' Theorem, the probability distribution of $\mathbf{U}$ can be conditioned on a realisation of $\mathbf{Y}$. The conditioned distribution is the *posterior distribution* of $\mathbf{U}$ given an observation $\mathbf{y}$. Bayes' formula for inverse problems is stated below.

2.2.1 Bayes' formula

Assume that $\mathbf{U}$ follows a prior $\mu_{\mathbf{U}}$ with Lebesgue density $\pi_{\mathbf{U}}(\mathbf{u})$. We assume that the noise $\mathbf{E}$ is independent of $\mathbf{U}$ and is distributed according to the measure $\mu_{\mathbf{E}}$ with Lebesgue density $\pi_{\mathbf{E}}(\mathbf{e})$. Then the *likelihood* of $\mathbf{Y}\,|\,\mathbf{u}$ is found by shifting $\mu_{\mathbf{E}}$ by $\mathbf{Au}$. This shifted measure is denoted by $\mu_{\mathbf{Y}}^{\mathbf{u}}$ with the Lebesgue density $\pi_{\mathbf{Y}}^{\mathbf{u}}(\mathbf{y}) = \pi(\mathbf{y}|\mathbf{u}) = \pi_{\mathbf{E}}(\mathbf{y} - \mathbf{Au})$.

Theorem 2.2.1: Bayes' Theorem

Suppose

$$Z(\mathbf{y}) = \int_{\mathbb{R}^n} \pi_{\mathbf{E}}(\mathbf{y} - \mathbf{Au})\pi_{\mathbf{U}}(\mathbf{u})\, d\mathbf{u} > 0.$$

Then $\mathbf{U}\,|\,\mathbf{y}$ is a random variable following the measure $\mu_{\mathbf{U}}^{\mathbf{y}}$ with Lebesgue density $\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})$ given by

$$\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u}) = \pi(\mathbf{u}\,|\,\mathbf{y}) = \frac{\pi_{\mathbf{E}}(\mathbf{y} - \mathbf{Au})\pi_{\mathbf{U}}(\mathbf{u})}{Z(\mathbf{y})}. \tag{2.27}$$

The term *posterior distribution* refers to the measure $\mu_{\mathbf{U}}^{\mathbf{y}}$.

The likelihood $\pi_{\mathbf{E}}(\mathbf{y} - \mathbf{Au})$ summarises our information about the observation, noise, and forward operator. It quantifies the data misfit. The prior density $\pi_{\mathbf{U}}(\mathbf{u})$ is independent of the measurement and assigns higher probabilities to values of $\mathbf{U}$ that we expect to see based on our *a priori* information. The probability of our measurement is denoted by $Z(\mathbf{y})$, which plays the role of a normalising constant. The posterior density $\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})$ is our solution, obtained by updating the prior density.

We define

$$\phi(\mathbf{u}) = -\log(\pi_{\mathbf{E}}(\mathbf{y} - \mathbf{Au})) \tag{2.28}$$

and call this the *potential.*

Let $\mu_{\mathbf{U}}^{\mathbf{y}}$ and $\mu_{\mathbf{U}}$ be measures on $\mathbb{R}^n$ with densities $\pi_{\mathbf{U}}^{\mathbf{y}}$ and $\pi_{\mathbf{U}}$, respectively. Then we can rewrite Theorem 2.2.1 as

$$\frac{d\mu_{\mathbf{U}}^{\mathbf{y}}}{d\mu_{\mathbf{U}}}(\mathbf{u}) = \frac{1}{Z(\mathbf{y})}\exp\left(-\phi(\mathbf{u})\right) \tag{2.29}$$

with

$$Z(\mathbf{y}) = \int_{\mathbb{R}^n} \exp\left(-\phi(\mathbf{u})\right)\mu_{\mathbf{U}}(d\mathbf{u}).$$

The posterior (2.27) is absolutely continuous with respect to the prior and the Radon-Nikodym derivative is proportional to the likelihood. To find a single value that represents the posterior distribution, a point estimator can be taken.

2.2.2 Estimators

The solution of a Bayesian inverse problem (2.26) is the posterior probability distribution with the density (2.27). For discrete inverse problems where $n = 1$, 2, or 3, it can be straightforward to represent the solution visually using plots. Discrete inverse problems in these spaces form a very limited subset of inverse problems, and most inverse problems are in higher (or even infinite) dimensional spaces. Estimators are needed to investigate the properties of the posterior distribution.

In many applications, it is useful to compute one value that can represent the distribution, or a *point estimator*. A point estimator can be used to answer the questions we are interested in, depending on the application. For example, in deblurring, we may want to recover one image out of our posterior distribution; in 1D deconvolution, we want to recover one signal u over the domain.

The maximum a posteriori (MAP) estimator is a popular choice of estimator. For discrete inverse problems, a definition is given below.

Definition 2.2.2: Maximum a posteriori estimator

Given the posterior probability density $\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})$, the MAP estimator $\mathbf{u}_{MAP}$ satisfies

$$\mathbf{u}_{MAP} = \underset{\mathbf{u}\in\mathbb{R}^n}{\operatorname{argmax}}\, \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u}) \tag{2.30}$$

provided it exists.

The MAP estimator, if it exists, may not be unique. The MAP estimator may also be referred to as the posterior mode.

Another possible estimator is the conditional mean.

Definition 2.2.3: Conditional mean

Given the posterior probability density $\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})$, the conditional mean $\mathbf{u}_{CM}$ satisfies

$$\mathbf{u}_{CM} = \int_{\mathbb{R}^n} \mathbf{u}\, \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})\, d\mathbf{u}. \tag{2.31}$$

Finding the conditional mean requires the computation of high-dimensional integrals. This is rarely feasible in practice. Instead of computing the conditional mean directly, the integral (2.31) can be approximated with the empirical averange [33],

$$\bar{\mathbf{u}}_{est} = \frac{1}{M}\sum_{k=1}^{M} \mathbf{u}_{\mathbf{k}}, \tag{2.32}$$

where $\mathbf{u}_{\mathbf{k}}$ with $k = 1, 2, \ldots, M$ are samples following a distribution with the density $\pi_{\mathbf{U}}^{\mathbf{y}}$. Computational methods for this purpose are the subject of Chapter 3.

2.3 Priors

2.3.1 Gaussian priors

Gaussian priors offer a number of practical advantages. They are easy to construct and often lead to exact estimators. Additionally, computational methods for sampling from discrete Gaussian distributions are available and implemented in many software libraries. They are also useful for suppressing the effects of measurement noise, as Gaussian priors tend to encourage smoothness.

For the $\mathbb{R}^n$-valued random variable $\mathbf{U}$, a Gaussian prior can be constructed by setting the *mean* and *covariance*, which charaterise the Gaussian prior. The definition of a multivariate Gaussian random variable below.

Definition 2.3.1: Multivariate Gaussian random variable

Let $\mathbf{m} \in \mathbb{R}^n$ and $\mathbf{\Gamma} \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. A Gaussian random variable $\mathbf{X}$ with the mean $\mathbf{m}$ and covariance $\mathbf{\Gamma}$ has the probability density

$$\pi(\mathbf{x}) \propto \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{m})^\intercal \mathbf{\Gamma}^{-1}(\mathbf{x}-\mathbf{m})\right). \tag{2.33}$$

We say that $\mathbf{X} \sim \mathcal{N}(\mathbf{m}, \mathbf{\Gamma})$.

A Gaussian prior models the a priori belief that realisations of $\mathbf{X}$ are likely to be close to the mean $\mathbf{m}$, with some deviations described my the covariance matrix $\mathbf{\Gamma}$. The values on the diagonal of $\mathbf{\Gamma}$ describe the uncertainty of individual elements of $\mathbf{X}$. The off-diagonals model *a priori* belief about how the different elements of $\mathbf{X}$ are correlated to each other. A diagonal covariance matrix $\mathbf{\Gamma}$ models uncorrelated variables. If the values of the diagonal elements of $\mathbf{\Gamma}$ are small, $\mathbf{X}$ is believed to take values close to the mean.

A formula for the posterior density when the prior is Gaussian, the noise is Gaussian, and the forward operator is linear is derived in Theorem 3.10 in [10]. The formula is given below in Theorem 2.3.2.

Theorem 2.3.2

Let $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{A} \in \mathbb{R}^{m\times n}$. Suppose $\mathbf{E} \sim \mathcal{N}(\mathbf{0}, \Gamma_\mathbf{E})$. Let $\mathbf{D}$ be a matrix such that $\mathrm{Ker}(\mathbf{A}) \cap \mathrm{Ker}(\mathbf{D}) = \{0\}$. The function

$$\pi_\mathbf{U}^\mathbf{y}(\mathbf{u}) \propto \exp\left(-\frac{1}{2}(||\mathbf{Du}||^2 + (\mathbf{y}-\mathbf{Au})^\intercal {\Gamma_\mathbf{E}}^{-1}(\mathbf{y}-\mathbf{Au})\right) \tag{2.34}$$

defines a Gaussian density function over $\mathbb{R}^n$ with the mean

$$(\mathbf{D}^\intercal\mathbf{D} + \mathbf{A}^\intercal {\Gamma_\mathbf{E}}^{-1}\mathbf{A})^{-1}\mathbf{A}^\intercal {\Gamma_\mathbf{E}}^{-1}\mathbf{y} \tag{2.35}$$

and covariance matrix

$$(\mathbf{D}^\intercal\mathbf{D} + \mathbf{A}^\intercal {\Gamma_\mathbf{E}}^{-1}\mathbf{A})^{-1}.$$

Proof. Let $\mathbf{M} = \mathbf{D}^\intercal\mathbf{D} + \mathbf{A}^\intercal {\Gamma_\mathbf{E}}^{-1}\mathbf{A}$ and $\mathbf{b} = \mathbf{A}^\intercal {\Gamma_\mathbf{E}}^{-1}\mathbf{y}$.

Let $\mathbf{x} \in \operatorname{Ker}(\mathbf{M})$. Then we have $\mathbf{x}^\intercal \mathbf{M}\mathbf{x} = ||\mathbf{D}\mathbf{x}||_2^2 + ||{\Gamma_\mathbf{E}}^{-1/2}\mathbf{A}\mathbf{x}||_2^2 = 0$ and so $\operatorname{Ker}(\mathbf{M}) \subseteq \operatorname{Ker}(\mathbf{A}) \cap \operatorname{Ker}(\mathbf{D})$. Since $\operatorname{Ker}(\mathbf{A}) \cap \operatorname{Ker}(\mathbf{D}) = \{0\}$, it follows that $\operatorname{Ker}(\mathbf{M}) = \{0\}$ and $\mathbf{M}$ is invertible.

Next, we can expand $||\mathbf{D}\mathbf{u}||^2 + (\mathbf{y} - \mathbf{A}\mathbf{u})^\intercal {\Gamma_\mathbf{E}}^{-1}(\mathbf{y} - \mathbf{A}\mathbf{u})$ to obtain

$$||\mathbf{D}\mathbf{u}||^2 + (\mathbf{y} - \mathbf{A}\mathbf{u})^\intercal {\Gamma_\mathbf{E}}^{-1}(\mathbf{y} - \mathbf{A}\mathbf{u}) = \mathbf{u}^\intercal(\mathbf{D}^\intercal\mathbf{D} + \mathbf{A}^\intercal{\Gamma_\mathbf{E}}^{-1}\mathbf{A})\mathbf{u} - 2\mathbf{u}^\intercal\mathbf{A}^\intercal{\Gamma_\mathbf{E}}^{-1}\mathbf{y} + \mathbf{y}^\intercal{\Gamma_\mathbf{E}}^{-1}\mathbf{y}.$$

Then

$$\begin{aligned}\mathbf{u}^\intercal(\mathbf{D}^\intercal\mathbf{D} + \mathbf{A}^\intercal{\Gamma_\mathbf{E}}^{-1}\mathbf{A})\mathbf{u} - 2\mathbf{u}^\intercal\mathbf{A}^\intercal{\Gamma_\mathbf{E}}^{-1}\mathbf{y} + \mathbf{y}^\intercal{\Gamma_\mathbf{E}}^{-1}\mathbf{y} &= \mathbf{u}^\intercal\mathbf{M}\mathbf{u} - 2\mathbf{b}^\intercal\mathbf{u} + \mathbf{y}^\intercal{\Gamma_\mathbf{E}}^{-1}\mathbf{y} \\ &= (\mathbf{y} - \mathbf{M}^{-1}\mathbf{b})^\intercal\mathbf{M}(\mathbf{y} - \mathbf{M}^{-1}\mathbf{b}) - \mathbf{b}^\intercal\mathbf{M}^{-1}\mathbf{b} + \mathbf{y}^\intercal{\Gamma_\mathbf{E}}^{-1}\mathbf{y}.\end{aligned}$$

The last two terms are independent of $\mathbf{u}$.

$$\exp\left(-\frac{1}{2}(||\mathbf{D}\mathbf{u}||^2 + (\mathbf{y} - \mathbf{A}\mathbf{u})^\intercal{\Gamma_\mathbf{E}}^{-1}(\mathbf{y} - \mathbf{A}\mathbf{u})\right) \propto \exp\left(-\frac{1}{2}\left((\mathbf{u} - \mathbf{M}^{-1}\mathbf{b})^\intercal\mathbf{M}(\mathbf{u} - \mathbf{M}^{-1}\mathbf{b})\right)\right)$$

which defines a Gaussian density with the mean $\mathbf{M}^{-1}\mathbf{b}$ and covariance $\mathbf{M}^{-1}$, as $\mathbf{M}$ is invertible. □

For discrete linear inverse problems with Gaussian noise and a Gaussian prior, the mean (2.35) is also the solution for the optimisation problem defined in (2.22) for $\Gamma_\mathbf{E} = \sqrt{\alpha}\mathbf{I}$ with α being the Tikhonov parameter. It is also the maximiser of the density function (2.34). In other words, the vector computed using the formula (2.35) is both the conditional mean and the *maximum a posteriori* estimator.

Gaussian processes can be used to construct function-valued priors. A definition of the Gaussian process from is given in [34] and presented below.

Definition 2.3.3: Gaussian process

A Gaussian process is a family $\{U_t\}_{t\in T}$ indexed by a parameter set $T \subset \mathbb{R}$ such that all its finite-dimensional distributions are Gaussian. In other words, all random vectors of the form $\begin{pmatrix} U_{t_0} & U_{t_1} & \dots & U_{t_n}\end{pmatrix}^\intercal$ are $\mathbb{R}^n$-valued Gaussian random vectors.
It is characterised by its positive definite covariance function

$$\operatorname{Cov}(U_t, U_s) = \mathbb{E}[(U_t - \mathbb{E}(U_s))(U_s - \mathbb{E}(U_t))], \quad t, s \in T.$$

To construct Gaussian process priors that lend themselves well to discretisation, we want to be able to represent them as infinite series which can be truncated to n terms. One such representation is the Karhunen-Loève expansion. It is briefly discussed in [35] for Gaussian processes taking values in general Hilbert spaces. Here, we present the definition from [35] for Gaussian processes taking values in $L^2(\mathbb{R})$.

Definition 2.3.4: Karhunen-Loève expansion of a Gaussian process

Let $\{b_j\}_{j\geq 0}$ be an orthonormal basis of $L^2(\mathbb{R})$, $z_j \sim \mathcal{N}(0,1)$ be a sequence of independent $\mathbb{R}$-valued random variables, and a_j be a sequence of real numbers such that $\sum_{j=0}^{\infty} a_j^2 < \infty$. Then

$$U = \sum_{j=0}^{\infty} a_j z_j b_j \tag{2.36}$$

is the Karhunen-Loève expansion of a Gaussian random process with mean 0 and the covariance

$$\mathrm{Cov}(U_t, U_s) = \sum_{j=0}^{\infty} a_j^2 \cdot \langle U_t, b_j \rangle_{L^2(\mathbb{R})} \cdot \langle U_s, b_j \rangle_{L^2(\mathbb{R})} . \tag{2.37}$$

The series representation is discussed in more depth in [36]. In particular, the covariance operator is discussed in Remark 2, Section 2.4 in [36]. In the study of Bayesian inverse problems in general Banach spaces, the Karhunen-Loève expansion [16] is used to construct Gaussian priors for function-valued unknowns.

2.3.2 Total variation prior

The Gaussian priors described in the previous section lack one important quality: they are not edge-preserving. An *edge-preserving* prior [25], [26], [37] is a prior that smooths out noise while maintaining sharp transitions in the signal. Sharp transitions include steps, discontinuities, and steep gradients. In the true function plotted in Figure 2.2, we have a steep drop after a linear increase and several steps. A Gaussian prior may not recover these features effectively.

Images can be modeled as functions that are piecewise smooth with jump discontinuities which represent edges [26]. These functions are in the set of functions of bounded variation. The definition of the *total variation* [26] of a function is given below, together with the criterion a function must fulfill to be of bounded variation.

Definition 2.3.5: Total variation

Let $f : [0,1) \to \mathbb{R}$ be a function in $L^1([0,1))$. We define the total variation of f, denoted by $TV(f)$, as

$$TV(f) = \sup \left\{ \int_{[0,1)]} f \nabla \cdot g \, dx | g \in C_0^1([0,1), \mathbb{R}^n), \|g\| \leq 1 \right\} . \tag{2.38}$$

Here, $\nabla\cdot$ denotes the divergence operator and the space of test functions $C_0^1([0,1), \mathbb{R}^n)$ is the space of compactly-supported differentiable functions. A function is said to have bounded variation if $TV(f) < \infty$.

To make sense of total variation in a discrete setting, the total variation of a vector $\mathbf{f}$ is given by a function of $\mathbf{f} \in \mathbb{R}^n$.

Definition 2.3.6: Discrete total variation

Let $\mathbf{f} \in \mathbb{R}^n$ be a discretisation of a function in $L^1([0,1))$. We define the total variation of $\mathbf{f}$, denoted by $\mathbf{TV}(\mathbf{f})$, as

$$\mathbf{TV}(\mathbf{f}) = \sum_{j=1}^{N} |(\mathbf{Df})_j| \tag{2.39}$$

where $\mathbf{D}$ is the difference matrix (2.24).

The total variation prior [10] is defined below.

Definition 2.3.7: Total variation prior

The total variation prior density of $\mathbf{U} : \Omega_1 \to \mathbb{R}^n$ is

$$\pi_{\mathbf{U}}(\mathbf{u}) \propto \exp\left(-\lambda \sum_{j=1}^{N} |(\mathbf{Du})_j|\right) \tag{2.40}$$

where $\lambda \in \mathbb{R}$ is a hyperparameter.

This TV prior is for a one-dimensional function u, and it is *edge-preserving* for discrete inverse problems with a fixed number of dimensions. There is no known continuous random variable with the density (2.40) [25]. The next subsection covers priors that are defined for continuous random variables and have the desired edge-preserving properties.

2.3.3 Besov priors

Besov priors were proposed as priors for linear inverse problems in [27] and studied for non-linear inverse problems in [38]. They are capable of modeling large jumps between adjacent values that represent sharp features in images while suppressing noise. They were also shown to be discretisation-invariant for n-dimensional discretisations of u. This means that they model the same prior information about u, independent of n.

They are named after Besov spaces, which generalise the Sobolev spaces. These spaces contain functions with particular smoothness, regularity, and sparsity characteristics. When prior information of these characteristics is available, this information can be encoded into a prior distribution by constructing the appropriate Besov priors. A definition for Besov spaces is given below [39].

Definition 2.3.8: Besov spaces

A Besov space $B^s_{pq}(\mathbb{R}^d)$, with $1 \leq p, q < \infty$ and $s > 0$, consists of functions $f \in L^p(\mathbb{R}^d)$ with s_0 weak partial derivatives in $L^p(\mathbb{R}^d)$, where s_0 is the smallest integer such that $s_0 \leq s < s_0 + 1$, and where the modulus of continuity $\omega_{k,p}$ satisfies

$$\{2^{sj}\omega_{k,p}(f, 2^{-j})\}_{j\geq 0} \in \ell^q(\mathbb{N})$$

for any $k > s$. Here, $\omega_{k,p}$ is defined as

$$\omega_{k,p}(f, u) = \sup_{||h||_2 \leq u} ||\Delta^k_h f||_{L^p(\mathbb{R}^d)}$$

with

$$\Delta^k_h f(x) = \Delta^{k-1}_h (f(x) - f(x-h))$$

is the k-th order finite difference operator with the step size h.

Besov random variables can be constructed using functions with specific sparsity characteristics [38]. These functions form orthogonal bases for $L^2(\mathbb{R}^d)$ with $d \leq 3$. Before discussing the functions in question and Besov random variables, preliminary definitions are presented below. Definitions 2.3.9-2.3.11 are from [40].

Definition 2.3.9: Multiresolution approximation

A multiresolution approximation of $L^2(\mathbb{R}^d)$ is an increasing sequence $V_j, j \in \mathbb{Z}$, with the following properties:

(1) $\bigcap_{-\infty}^{\infty} V_j = \{0\}$, $\bigcup_{-\infty}^{\infty} V_j$ is dense in $L^2(\mathbb{R}^d)$,

(2) for all $f \in L^2(\mathbb{R}^d)$ and $j \in \mathbb{Z}$, $f(x) \in V_j \iff f(2x) \in V_{j+1}$,

(3) for all $f \in L^2(\mathbb{R}^d)$ and all $k \in \mathbb{Z}^d$, $f(x) \in V_0 \iff f(x-k) \in V_0$, and

(4) there exists a function, $g(x) \in V_0$ such that the sequence $g(x-k), k \in \mathbb{Z}^d$ is a Riesz basis of the space V_0.

A sequence of elements $\{f_j\}_{j\geq 0} \in L^2(\mathbb{R}^d)$ is a Riesz basis of $L^2(\mathbb{R}^d)$ if there exist constants $C' > C > 0$ such that, for every sequence of scalars $\alpha_0, \alpha_1, \alpha_2, \ldots$ we have

$$C\left(\sum_{k=0}^{\infty} |\alpha_k|^2\right)^{1/2} \leq ||\sum_{k=0}^{\infty} \alpha_k f_k||_{L^2(\mathbb{R}^d)} \leq C'\left(\sum_{k=0}^{\infty} |\alpha_k|^2\right)^{1/2}$$

and the vector space of finite sums $\sum_{k=0}^{K} \alpha_k f_k$, $K \geq 0$ is dense in $L^2(\mathbb{R}^d)$.

Definition 2.3.10: r-regular

A function f is r-regular if $f \in C^r$ and

$$|\partial^a f(x)| \leq C_l (1 + ||x||_1)^{-l} \tag{2.41}$$

for any $l \in \mathbb{N}$ and any multi-index $|a| = a_1 + a_2 + \ldots + a_d \leq r$, where C_l is a constant depending on l.

The wavelet expansion of a function $f \in L^2(\mathbb{R}^d)$ [41] is given below.

Definition 2.3.11: Wavelet expansion

Suppose the function $\varphi \in L^2(\mathbb{R}^d)$ generates a multiresolution approximation (V_j) with dilation matrix $2\mathbf{I}^d$. Let $\psi^l \in L^2(\mathbb{R}^d), l \in \{1, 2, \ldots, 2^d - 1\}$ denote the associated family of wavelets [40]. Suppose φ and ψ are r-regular. The dilation and translation of the wavelets and the scaling function are given by

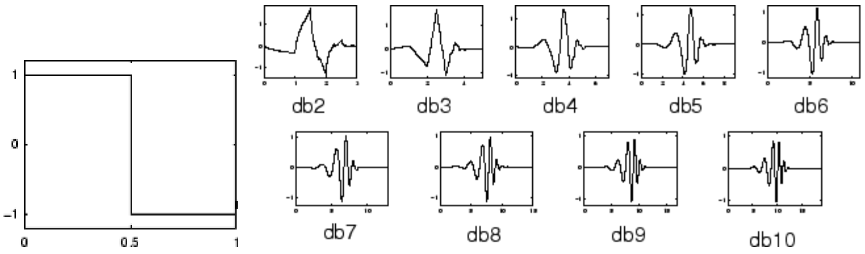
$$\psi^l_{j,k}(x) = 2^{jd/2}\psi^l(2^j x - k),$$
$$\varphi_k(x) = \varphi(x - k)$$

with $l \in \{1, 2, \ldots, 2^d - 1\}, j \geq 0, k \in \mathbb{Z}^d$. Then any function $f \in L^2(\mathbb{R}^d)$ has a wavelet expansion given by

$$f = \sum_{k \in \mathcal{Z}^d} v_k \varphi_k + \sum_{l=1}^{2^d-1} \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}^d} w^l_{j,k} \psi^l_{j,k} \tag{2.42}$$

where $v_k = \langle f, \varphi_k \rangle_{L^2(\mathbb{R}^d)}$ and $w^l_{j,k} = \left\langle f, \psi^l_{j,k} \right\rangle_{L^2(\mathbb{R}^d)}$ with unconditional convergence in the norm.

Background on wavelet representations can be found in [40], [42], [43]. Two examples of wavelet families ψ, the Haar and Daubechies wavelets, are shown in Figure 2.8.



(a) The Haar wavelet function ψ.

(b) Wavelet functions ψ in the Daubechies family of wavelets.

Figure 2.8: Plots of wavelet functions [44].

Wavelet basis functions are used in multiscale analysis, as they can model behavior on coarse and fine grids. Wavelet basis functions have localised supports [42], meaning that they can model sparse behavior, such as sudden jumps or steep slopes.

The wavelet characterisation of Besov spaces from [45] is presented as Definition 2.3.12 below.

Definition 2.3.12: Wavelet characterisation of Besov spaces

Let $\psi^l, l \in \{1, 2, \ldots, 2^d - 1\}$ be the family of wavelets and φ be the scaling function of a multiresolution analysis of $L^2(\mathbb{R}^d)$ with a regularity $r \geq 1$. Let $1 \leq p, q < \infty$ and $s > 0$ such that $s < r$. The function $f \in L^p(\mathbb{R}^d)$ has the wavelet expansion (2.42), which satisfies $\{v_k\}_{k \in \mathcal{Z}^d} \in \ell^p(\mathbb{Z}^d)$ and

$$\left\{ 2^{j(s+d/2-d/p)} \left(\sum_{l=1}^{2^d-1} \sum_{k \in \mathbb{Z}^d} \left| w^l_{j,k} \right|^p \right)^{\frac{1}{q}} \right\}_{j \geq 0} \in \ell^q(\mathbb{N}), \tag{2.43}$$

if and only if $f \in B^s_{pq}(\mathbb{R}^d)$. The norm induced on the Besov space is

$$||f||_{B^s_{pq}(\mathbb{R}^d)} = \left(\sum_{k \in \mathbb{Z}^d} |v_k|^p \right)^{\frac{1}{p}} + \left(\sum_{j=0}^{\infty} 2^{j(s+d/2-d/p)} \left(\sum_{l=1}^{2^d-1} \sum_{k \in \mathbb{Z}^d} \left| w^l_{j,k} \right|^p \right)^{\frac{q}{p}} \right)^{\frac{1}{q}}. \tag{2.44}$$

Besov random variables [38], which are constructed using wavelet expansions restricted to functions on the torus $(\mathbb{T}^d)$ with $p = q$, are defined below. The scaling coefficients γ_l in Definition 2.3.13 are adapted from [41] and correspond to the scaling of the wavelets in the `PyWavelets` package used in this thesis.

Definition 2.3.13: Besov random variables

Let $\psi^l, l \in \{1, 2, \ldots, 2^d - 1\}$ be the family of wavelets of a multiresolution analysis in $L^2(\mathbb{T}^d)$ with regularity $r \geq 1$. Let $1 \leq q < \infty$ and $s > 0$ such that $s < r$. Let $\mathcal{K}_j = \{0, \ldots, 2^j - 1\}^d$ be an index set and $\{\xi^l_{j,k}\}_{j \geq 0, k \in \mathcal{K}_j}, \xi_0$ be i.i.d. real-valued random variables with density $\pi_\Xi(\xi) \propto \exp\left(-\frac{1}{2}|\xi|^q\right)$. Let $\delta > 0$. Let U be defined as

$$U(x) = \xi_0 + \sum_{l=1}^{2^d-1} \sum_{j=0}^{\infty} \sum_{k \in \mathcal{K}_j} \delta^{-\frac{1}{q}} \gamma_j \xi^l_{j,k} \psi^l_j(x - k2^{-j}), \tag{2.45}$$

for almost everywhere $x \in \mathbb{T}^d$, where

$$\gamma_j = 2^{-j\left(s+\frac{d}{2}-\frac{d}{p}\right)}, \quad \text{and} \quad \psi^l_j(x) = 2^{jd/2} \sum_{m \in \mathbb{Z}^d} \psi^l(2^j(x-m)) \tag{2.46}$$

is the 1-periodisation of $\psi^l_{j,k}$. We say that U is a Besov $B^s_{qq}(\mathbb{T}^d)$ random variable if the series (2.45) converges. The Besov norm of U is

$$||U||_{B^s_{qq}(\mathbb{T}^d)} = \left(|\xi_0|^q + \sum_{l=1}^{2^d-1} \sum_{j=0}^{\infty} \sum_{k \in \mathcal{K}_j} 2^{j\left(s+\frac{d}{2}-\frac{d}{q}\right)} \left| \delta^{-\frac{1}{q}} \gamma_j \xi^l_{j,k} \right|^q \right)^{\frac{1}{q}}. \tag{2.47}$$

The Lebesgue density does not exist for infinite-dimensional spaces. Informally, the 'Lebesgue density' of the random variable U in (2.45) is $\pi(u) \propto \exp\left(-\frac{\delta}{2}||u||^q_{B^s_{qq}(\mathbb{T}^d)}.\right)$ We say that U in (2.45) is distributed according to the $(\kappa; B^s_{qq})$ measure with

$$\kappa = \frac{\delta}{2}. \tag{2.48}$$

In the rest of this thesis, $d = 1$ as this thesis focuses on one-dimensional deconvolution. The parameter s in (2.45) can be chosen to determine the decay of the wavelet coefficients and therefore control the regularity of the random functions constructed in (2.45).

The random variables presented so far are candidates for priors in the formula (2.26). The solution discussed in Subsection 2.2.1 is a probability density function. While it is possible to explicitly compute point estimators for posterior densities obtained using Gaussian priors (2.3.2), it is not possible to do so for the total variation and Besov priors. In the next chapter, we discuss theoretical underpinnings of the computational methods used to sample from posterior distributions obtained using the non-Gaussian priors.

Chapter 3

Markov chain Monte Carlo methods

The solution of a Bayesian inverse problem is the posterior probability density (2.27). In order to find an approximate value of the conditional mean (2.31) and perform uncertainty quantitification, we want to generate a large amount of values in $\mathbb{R}^n$ following $\pi_{\mathbf{U}}^{\mathbf{y}}$. This collection process is referred to as sampling.

Unfortunately, it is rarely possible to sample directly from our posterior distribution. To work around this limitation, we can determine the transition properties of a discrete time process such that its individual components are distributed according to the posterior distribution once enough samples have been gathered. This is the main idea behind Markov chain Monte Carlo methods. We present background on general (discrete time) Markov chains, then discuss Metropolis-Hastings algorithms for Bayesian inverse problems.

3.1 Markov chains and their invariant measures

This section presents background on Markov chains. The definitions are adapted from [33], which the reader may consult for more details. A Markov chain is a stochastic process characterised by a *transition kernel*, defined below.

Definition 3.1.1: Transition kernel

Let S_X be a space with a σ-algebra $\sigma(S_X)$. A transition kernel is a function $K(\cdot,\cdot)$ defined on $S_X \times \sigma(S_X)$ such that:

(i) $\forall x \in S_X, K(x,\cdot)$ is a probability measure;

(ii) $\forall B_X \in \sigma(S_X), K(\cdot, B_X)$ is measurable.

The transition kernel specifies the transition properties of the Markov chain.

Definition 3.1.2: Markov chain

Given a transition kernel K and a measurable space $(S_X, \sigma(S_X))$, a sequence of S_X-valued random variables $X_0, X_1, \dots, X_n, \dots$ is a (discrete-time) Markov chain $\{X_n\}_{n\geq 0}$ if, for any t, $\mathbb{P}(X_{t+1} \in B_X | x_0, \dots x_t) = \mathbb{P}(X_{t+1} \in B_X | x_t) = \int_{B_X} K(x_t, dx_{t+1}) \forall B_X \in \sigma(S_X)$.

Given the k-th element of the chain, the probability of the next element being in any set B_X is the same as the probability of the next element being in B_X conditioned on all k elements of the chain. In other words, we only need to know X_k when generating X_{k+1}. The computational cost of generating one sample is thus independent of the number of samples that have already been generated.

Now that we have defined a discrete time stochastic process that can be used to systematically produce k samples, we want to define a relationship between a Markov chain and the posterior distribution such that the distribution of elements in the chain eventually approaches the posterior distribution. This relationship is called invariance.

Definition 3.1.3: Invariant measure

A σ-finite measure μ is invariant for the transition kernel $K(\cdot, \cdot)$ and the associated chain if $\mu(B_{X_{k+1}}) = \int_{S_X} K(x_k, B_{X_{k+1}})\, \mu(dx_k) \quad \forall B_{X_{k+1}} \in \sigma(S_X)$.

If X_k is distributed according to μ, X_{k+1} is also distributed according to μ. Furthermore, for all k, if $X_0 \sim \mu$, X_k also has the distribution μ. Ideally, we would like this invariant measure to be the target distribution. For a Markov chain $\{X_k\}_{k\geq 0}$ with an invariant measure μ, the average

$$\frac{1}{M} \sum_{k=1}^{M} X_k$$

converges to $\mathbb{E}_\mu(X)$ almost surely [33] as $M \to \infty$. This average is thus an approximation of the conditional mean (2.31).

It can be difficult to use Definition 3.1.3 to construct an appropriate Markov chain. Another property, called reversibility, is a sufficient condition for a probability measure to be the invariant measure of a Markov chain.

Definition 3.1.4: Reversibility

A Markov chain with transition kernel K is reversible if and only if there is a measure μ satisfying the detailed balance relation

$$\mu(A)K(x, B) = \mu(B)K(y, A). \tag{3.1}$$

$\forall (x, y) \in S_X \times S_X$ and $A, B \in \sigma(S_X)$. The measure μ is the invariant measure of the Markov chain.

If we have a Markov chain $\{X_k\}_{k\geq 0}$ satisfying the balance equation and we build a reversed Markov Chain $Y_k = X_{k-l}$ for $l = 0, 1, \dots, k$, the one-step transition probability from X_k to X_{l+1} is the same as the one-step transition probability from Y_k to Y_{l+1} for all $l = 0, 1, \dots, k-1$. The Markov chain is then in a steady state, as the probability of returning to X_p from X_q for $q > p$ is the same as the probability of transitioning from X_p to X_q. We now prove that the balance condition (3.1) is a sufficient condition for determining that a distribution is invariant for a Markov chain.

Theorem 3.1.5

Suppose that a Markov chain with transition kernel K satisfies the balance condition (3.1) together with μ, a probability measure. Then the measure μ is the invariant measure of the chain.

Proof. Let K and μ be a transition kernel and probability measure such that the balance condition (3.1) is fulfilled. Let B_X be any measurable set in $\sigma(S_X)$. Note that, for fixed $x \in S_X$, $\int_{S_X} K(x, dy) = 1$ by the definition of a transition kernel and

$$\begin{aligned}\int_{S_X} K(y, B_X)\,\mu(dy) &= \int_{S_X}\int_{B_X} K(y, dx)\mu(dy) \\ &= \int_{B_X}\int_{S_X} K(x, dy)\mu(dx) \qquad = \int_{B_X}\mu(dx)\int_{S_X} K(x, dy) = \int_{B_X}\mu(dx).\end{aligned}$$

□

We thus want to devise a method to construct a Markov chain $\{U_k\}_{k\geq 0}$ such that its invariant measure has the density $\pi_{\mathbf{U}}^{\mathbf{y}}$. We can do so using the Metropolis-Hastings algorithm, which provides us a way to construct transition kernels that fulfill the balance condition (3.1) together with the posterior measure with density $\pi_{\mathbf{U}}^{\mathbf{y}}$.

3.2 Metropolis-Hastings algorithms

In this section, we will introduce the *Metropolis-Hastings transition kernels* and show how these transition kernel can be used to construct a Markov chain with a specified invariant measure. One iteration of a Metropolis-Hastings algorithm [33] is described below. A definition of the transition kernel of the Markov chain generated using a Metropolis-Hastings algorithm is then given.

Definition 3.2.1: Metropolis-Hastings procedure

Let the *draw* $\mathbf{u_k} \in \mathbb{R}^n$ be the k-th element in the Markov chain. Let the $\mathbb{R}^n$-valued random variable $\mathbf{V}$, the *proposal*, have a probability density function, π_K. In most cases, the function $\pi_K(\mathbf{u_k}, \mathbf{v}) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^+$ is the probability density function of $\mathbf{V}$ conditioned on $\mathbf{u_k}$. In the case where the density of π_K does not depend on $\mathbf{u_k}$, we will write $\pi_K(\mathbf{v}) : \mathbb{R}^n \to \mathbb{R}^+$. Let $\alpha(\mathbf{u_k}, \mathbf{v}) : \mathbb{R}^n \times \mathbb{R}^n \to [0, 1]$ be the acceptance probability.

1. **Generate proposal.** Draw a proposal $\mathbf{v}$ using the *proposal generation kernel* with the proposal probability density $\pi_K(\mathbf{u_k}, \mathbf{v})$.
2. **Find acceptance probability.** Compute $\alpha(\mathbf{u_k}, \mathbf{v})$. Later in this section, we will discuss how α is chosen in more depth.
3. **Accept-or-reject step.** Draw $\mathbf{a} \sim Unif(0, 1)$. If $\alpha(\mathbf{u_k}, \mathbf{v}) > \mathbf{a}$, accept proposal $\mathbf{v}$ as the draw from this iteration. Otherwise, reject $\mathbf{v}$ and treat $\mathbf{u_k}$ as the draw from this iteration.

The transition kernel of the Markov chain constructed using this procedure is given by [46]

$$K_{MH}(\mathbf{u_k}, B) = \int_B f_K(\mathbf{u_k}, \mathbf{v})\,d\mathbf{v} + r_K(\mathbf{u_k})\mathbb{1}_B(\mathbf{u_k}) \tag{3.2}$$

where $f_K(\mathbf{u_k},\mathbf{v}) = \pi_K(\mathbf{u_k},\mathbf{v})\alpha(\mathbf{u_k},\mathbf{v})$ and $r_K(\mathbf{u_k}) = 1 - \int_{\mathbb{R}^n} \pi_K(\mathbf{v},\mathbf{u_k})\alpha(\mathbf{v},\mathbf{u_k})\, d\mathbf{v}$. The function f_K represents the probability of moving to $\mathbf{v}$ from $\mathbf{u_k}$ and r represents the probability of remaining at $\mathbf{v}$.

When we carry out steps 1-3, we carry out one iteration of the Metropolis-Hastings algorithm. We repeat the procedure M times to collect M samples, and use M to denote the *number of iterations* in our algorithm. The detailed balance condition (3.1) for Metropolis-Hastings algorithms in terms of densities is stated below. This condition can be used to check that a Markov chain constructed using a Metropolis-Hastings procedure has an invariant distribution with the posterior density $\pi_{\mathbf{U}}^{\mathbf{y}}$.

Theorem 3.2.2: Balance condition for Metropolis-Hastings algorithms

Let π_K be a probability density function, $\alpha : \mathbb{R}^n \times \mathbb{R}^n \to [0,1]$, and Metropolis-Hastings procedure be as described in Definition 3.2.1. Let $\pi_{\mathbf{U}}^{\mathbf{y}}$ be the posterior probability density function with respect to ν as defined in (2.27). If there is a function $\alpha : \mathbb{R}^n \times \mathbb{R}^n \to [0,1]$ such that

$$\alpha(\mathbf{u_k},\mathbf{v})\frac{\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u_k})\pi_K(\mathbf{u_k},\mathbf{v})}{\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})\pi_K(\mathbf{v},\mathbf{u_k})} = \alpha(\mathbf{v},\mathbf{u_k}) \tag{3.3}$$

for any $(\mathbf{u_k},\mathbf{v}) \in \{(\mathbf{u_k},\mathbf{v}) \in \mathbb{R}^n \times \mathbb{R}^n : \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})\pi_K(\mathbf{v},\mathbf{u_k}) > 0\}$, then the invariant measure of the Markov chain characterised by the transition kernel (3.2) is the posterior probability distribution with the density function $\pi_{\mathbf{U}}^{\mathbf{y}}$ [46].

Proof. We write

$$r_K(\mathbf{v}) = 1 - \int_{\mathbb{R}^n} \pi_K(\mathbf{u_k},\mathbf{v})\alpha(\mathbf{u_k},\mathbf{v})\, d\mathbf{u_k}$$

and see that

$$\int_{\mathbb{R}^n} f_K(\mathbf{v},\mathbf{u_k})\, d\mathbf{u_k} = 1 - r_K(\mathbf{v}).$$

When the condition (3.3) is fulfilled, we have

$$\begin{aligned}\int_{\mathbb{R}^n} f_K(\mathbf{u_k},\mathbf{v})\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u_k})\, d\mathbf{u_k} &= \int_{\mathbb{R}^n} \pi_K(\mathbf{u_k},\mathbf{v})\alpha(\mathbf{u_k},\mathbf{v})\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u_k})\, d\mathbf{u_k}\\ &= \int_{\mathbb{R}^n} \pi_K(\mathbf{v},\mathbf{u_k})\alpha(\mathbf{v},\mathbf{u_k})\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})\, d\mathbf{u_k}\\ &= \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})\int_{\mathbb{R}^n} f_K(\mathbf{v},\mathbf{u_k})\, d\mathbf{u_k}. \qquad (3.4)\end{aligned}$$

Then,

$$\int_{\mathbb{R}^n} f_K(\mathbf{u_k},\mathbf{v})\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u_k})\, d\mathbf{u_k} = \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})(1 - r_K(\mathbf{v})). \tag{3.5}$$

For any $(\mathbf{u}, \mathbf{v}) \in \mathbb{R}^n \times \mathbb{R}^n$ and $B \in \sigma(\mathbb{R}^n)$, we then have

$$\begin{aligned}
\int_{\mathbb{R}^n} K_{MH}(\mathbf{u}, B)\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})\, d\mathbf{u} &= \int_{\mathbb{R}^n} \int_B f_K(\mathbf{u}, \mathbf{v})\, d\mathbf{v} + r_K(\mathbf{u})\mathbb{1}_B(\mathbf{u})\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})\, d\mathbf{u} \\
&= \int_{\mathbb{R}^n} \int_B f_K(\mathbf{u}, \mathbf{v})\, d\mathbf{v}\, \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})\, d\mathbf{v} + \int_{\mathbb{R}^n} r_K(\mathbf{u})\mathbb{1}_B(\mathbf{u})\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})\, d\mathbf{u} \\
&= \int_B \int_{\mathbb{R}^n} f_K(\mathbf{u}, \mathbf{v})\, d\mathbf{u}\; d\mathbf{v} + \int_B r_K(\mathbf{v})\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})\, d\mathbf{v} \\
&= \int_B \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})(1 - r_K(\mathbf{v})) + r_K(\mathbf{v})\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})\, d\mathbf{v} \\
&= \int_B \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})\, d\mathbf{v}. \qquad (3.6)
\end{aligned}$$

The transition kernel (3.2) thus characterises a Markov chain with an invariant measure whose Lebesgue density is $\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u})$. □

3.3 Random walk Metropolis-Hastings

The simplest Metropolis-Hastings variant is the random walk proposal. We first choose a random variable $\mathbf{W}$ with the probability density function $g(\mathbf{w})$. The random variable $\mathbf{W}$ is usually chosen so that it is easy to sample from. Uniform and Gaussian random variables, for example, are popular choices for $\mathbf{W}$. We define the random walk proposal

$$\mathbf{V} = \mathbf{u_k} + \beta\mathbf{W} \qquad (3.7)$$

where $\beta > 0$ is a parameter also called the *step size*. Informally, it represents how far away from the current draw $\mathbf{u_k}$ we want to 'walk' when we generate a proposal.

The proposal probability density function is then $\pi_K(\mathbf{u_k}, \mathbf{v}) = g\left(\frac{\mathbf{v}-\mathbf{u_k}}{\beta}\right)$. We can then set the acceptance probability function as

$$\alpha(\mathbf{u_k}, \mathbf{v}) = \min\left\{1, \frac{\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})g\left(\frac{\mathbf{u_k}-\mathbf{v}}{\beta}\right)}{\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u_k})g\left(\frac{\mathbf{v}-\mathbf{u_k}}{\beta}\right)}\right\} \qquad (3.8)$$

to fulfill the balance condition (3.3). The accept-reject mechanism then selects for proposals in areas with higher values of $\pi_{\mathbf{U}}^{\mathbf{y}}$. In the special case where π_K is symmetrical, meaning $\pi_K(\mathbf{u_k}, \mathbf{v}) = \pi_K(\mathbf{v}, \mathbf{u_k})$, the acceptance probability is

$$\alpha(\mathbf{u_k}, \mathbf{v}) = \min\left\{1, \frac{\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})}{\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u_k})}\right\}. \qquad (3.9)$$

The step size β affects the effectiveness of the sampling method. If the step size β is too small, the increments of the Markov chain will be small and it will take a large number of iterations to sample from the invariant measure. If the step size β is too large, the Metropolis-Hastings algorithm will reject many of the proposals.

When random walk Metropolis-Hastings is used to sample from the posterior measure of a Bayesian inverse problem, the step size β has to be of order $\mathcal{O}(1/n)$ [47]. Otherwise, the step size is too large and very few proposals are accepted. The number of random walk Metropolis-Hastings iterations we need to carry out to sample from the invariant measure is $\mathcal{O}(n)$.

This poses a problem when n is large, which is often the case in inverse problems. If we wish to deblur a picture taken using an iPhone 15, we would need to work with a 1179 × 2556 pixel image. Then, $n > 3\,000\,000$. We would need to run the random walk Metropolis-Hastings algorithm for at least $\mathcal{O}(10^6)$ iterations, drawing a 3 000 000-dimensional sample at each iteration.

The method presented in the next section addresses this limitation.

3.4 Preconditioned Crank-Nicolson

The preconditioned Crank-Nicolson (pCN) Metropolis-Hastings algorithm [28] was designed for sampling from invariant measures of the form (2.29) where the prior measure $\mu_{\mathbf{U}}$ is a centered Gaussian probability measure. The pCN proposal is derived from a Crank-Nicolson discretisation of the equation

$$\frac{dU}{ds} = -(\mathcal{C}^{-1}U + \nabla\phi(U)) + \sqrt{2}\frac{dB}{ds} \tag{3.10}$$

where B is the standard Brownian motion on a Hilbert space B_u, $\mathcal{C}$ is the covariance operator of a Gaussian prior measure, and ϕ is the potential defined in (2.28). The pCN algorithm was constructed to build a Markov chain with function-valued elements by discretising (3.10). As the algorithm was originally designed for function spaces with infinite dimensions, it is also suitable for high-dimensional problems.

Suppose our invariant measure $\mu_{\mathbf{U}}^{\mathbf{y}}$ is absolutely continuous with respect to a prior measure $\mu_{\mathbf{U}}$ with the Lebesgue density

$$\pi_{\mathbf{U}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2}\langle \mathbf{u}, \mathbf{L}\mathbf{u}\rangle\right) \tag{3.11}$$

for some $\mathbf{L} \in \mathbb{R}^{n\times n}$, where $\langle\cdot,\cdot\rangle$ is the standard inner product on $\mathbb{R}^n$. By Bayes' Theorem (2.26), the Radon-Nikodym derivative of $\mu_{\mathbf{U}}^{\mathbf{y}}$ with respect to $\mu_{\mathbf{U}}$ is

$$\frac{d\mu_{\mathbf{U}}^{\mathbf{y}}}{d\mu_{\mathbf{U}}}(\mathbf{u}) = \exp\left(-\phi(\mathbf{u})\right)$$

for a Bayesian inverse problem with Gaussian noise and the potential (2.28) ϕ. The pCN proposal $\mathbf{V}$ is defined as

$$\mathbf{V} = \sqrt{1-\beta^2}\mathbf{u}_{\mathbf{k}} + \beta\mathbf{W}, \tag{3.12}$$

where $\beta \in [0,1]$ and $\mathbf{W}$ has the density specified in (3.11). Then we can write the proposal density of $\mathbf{V}$ as

$$\pi_K(\mathbf{u}_{\mathbf{k}}, \mathbf{v}) = (\beta\sqrt{2\pi})^{-n}\exp\left(-\frac{1}{2\beta^2}\left\langle \mathbf{v} - \sqrt{1-\beta^2}\mathbf{u}_{\mathbf{k}}, \mathbf{L}(\mathbf{v} - \sqrt{1-\beta^2}\mathbf{u}_{\mathbf{k}})\right\rangle\right). \tag{3.13}$$

Then

$$\begin{aligned}\frac{\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u_k})\pi_K(\mathbf{u_k},\mathbf{v})}{\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{v})\pi_K(\mathbf{v},\mathbf{u_k})} &= \exp\Bigg(-\frac{1}{2}\left\langle \mathbf{u_k}, \mathbf{L}\mathbf{u_k}\right\rangle - \phi(\mathbf{u_k}) + \frac{1}{2}\left\langle \mathbf{v}, \mathbf{L}\mathbf{v}\right\rangle + \phi(\mathbf{v}) \\ &\quad - \frac{1}{2\beta^2}\left\langle \mathbf{v} - \sqrt{1-\beta^2}\mathbf{u_k}, \mathbf{L}(\mathbf{v} - \sqrt{1-\beta^2}\mathbf{u_k})\right\rangle \\ &\quad + \frac{1}{2\beta^2}\left\langle \mathbf{u_k} - \sqrt{1-\beta^2}\mathbf{v}, \mathbf{L}(\mathbf{u_k} - \sqrt{1-\beta^2}\mathbf{v})\right\rangle \Bigg) \\ &= \exp\Bigg(\left\langle -\frac{1}{2}\mathbf{u_k} + \frac{\sqrt{1-\beta^2}}{2\beta^2}(\mathbf{v} - \sqrt{1-\beta^2}\mathbf{u_k}) + \frac{1}{2\beta^2}(\mathbf{u_k} - \sqrt{1-\beta^2}\mathbf{v}), \mathbf{L}\mathbf{u_k}\right\rangle \\ &\quad - \left\langle -\frac{1}{2}\mathbf{v} + \frac{\sqrt{1-\beta^2}}{2\beta^2}(\mathbf{u_k} - \sqrt{1-\beta^2}\mathbf{v}) + \frac{1}{2\beta^2}(\mathbf{v} - \sqrt{1-\beta^2}\mathbf{u_k}), \mathbf{L}\mathbf{v}\right\rangle + \phi(\mathbf{v}) - \phi(\mathbf{u_k}) \Bigg) \\ &= \exp\left(\phi(\mathbf{v}) - \phi(\mathbf{u_k})\right).\end{aligned}$$

We then set the acceptance probability function

$$\alpha(\mathbf{u_k}, \mathbf{v}) = \min\left\{1, \exp\left(\phi(\mathbf{u_k}) - \phi(\mathbf{v})\right)\right\} \tag{3.14}$$

to fulfill the balance condition (3.3). The acceptance probability (3.14) is a special case of the acceptance probability function for random variables taking values in general Hilbert spaces, which was derived in [47]. The pCN algorithm can be considered a generalisation of the random walk discussed in Section 3.3 to function spaces.

The parameter β, the step size, needs to be tuned to ensure the algorithm performs efficiently. In [28], the parameter β is tuned for individual test problems in order to achieve an average acceptance ratio of 25%. Choosing an optimal step size for pCN is an open problem. We will discuss how β affects the performance of the pCN algorithm for our 1D deconvolution problem in the next chapter.

3.5 Randomise-then-optimise

So far, we have focused on MCMC methods for inverse problems with linear forward operators. We now consider an MCMC method for inverse problems with non-linear forward operators, as the prior transformations introduced in Section 4.1 recast linear inverse problems with non-Gaussian priors as non-linear inverse problems with Gaussian priors. We present randomise-then-optimise (RTO), a Metropolis-Hastings algorithm formulated for inverse problems with non-linear forward operators [29], [48].

Consider the non-linear measurement model

$$\mathbf{Y} = \mathcal{F}(\mathbf{U}) + \mathbf{E} \tag{3.15}$$

where $\mathbf{U} : \Omega_1 \to \mathbb{R}^n$, $\mathbf{E} : \Omega_2 \to \mathbb{R}^m$, $\mathbf{Y} : \Omega \to \mathbb{R}^m$, and $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^m$ is a non-linear function. We want to condition $\mathbf{U}$ on a single realisation of $\mathbf{Y}$. Let the prior density function of $\mathbf{U}$ be an n-variate Gaussian density function with mean $\mathbf{U}_0$ and the identity matrix as its covariance and let the noise $\mathbf{E}$ follow a standard m-variate Gaussian distribution. The posterior density function of $\mathbf{U}$ given $\mathbf{y}$ is then

$$\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2}||\tilde{\mathcal{F}}(\mathbf{u}) - \tilde{\mathbf{y}}||^2\right) \tag{3.16}$$

where

$$\tilde{\mathcal{F}}(\mathbf{u}) = \begin{pmatrix} \mathbf{u} \\ \mathcal{F}(\mathbf{u}) \end{pmatrix}, \quad \tilde{\mathbf{y}} = \begin{pmatrix} \mathbf{u}_0 \\ \mathbf{y} \end{pmatrix}.$$

More generally, RTO is a method for sampling from densities of the form

$$\pi(\mathbf{u}) \propto \exp\left(-\frac{1}{2}||\tilde{\mathcal{F}}(\mathbf{u}) - \tilde{\mathbf{y}}||^2\right) \tag{3.17}$$

where $\tilde{\mathcal{F}} : \mathbb{R}^n \to \mathbb{R}^{m+n}$, $\tilde{\mathbf{y}} \in \mathbb{R}^{m+n}$ with $m > 0$.

Before applying RTO, we first determine three components that are present in every iteration of the algorithm. The first is a linearisation point $\bar{\mathbf{u}}_{MAP}$, which is fixed throughout the method. In [31], [48] the MAP estimator is chosen as the linearisation point. The second component is the Jacobian of $\mathbf{F}$, denoted by $\mathbf{J}_{\mathbf{F}}(\mathbf{u})$. The third component is $\bar{\mathbf{Q}} \in \mathbb{R}^{(m+n)\times n}$, the matrix of orthonormal basis vectors for the column space of $\mathbf{J}_{\mathbf{F}}(\bar{\mathbf{u}}_{MAP})$. The matrix $\bar{\mathbf{Q}}$ can be found by taking a thin-QR factorisation of $\mathbf{J}_{\mathbf{F}}(\bar{\mathbf{u}}_{MAP})$.

We can now define the RTO proposal [48]

$$\mathbf{V} = \underset{\mathbf{v}\in\mathbb{R}^n}{\operatorname{argmin}}\, ||\bar{\mathbf{Q}}^\intercal(\tilde{\mathcal{F}}(\mathbf{v}) - (\tilde{\mathbf{y}} + \mathbf{W}))||^2 \tag{3.18}$$

where $\mathbf{W} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

We now derive the density of the proposal $\mathbf{V}$. First, we assume that the MAP estimation problem

$$\bar{\mathbf{u}}_{MAP} = \underset{\mathbf{u}\in\mathbb{R}^n}{\operatorname{argmin}}\, \frac{1}{2}||\tilde{\mathcal{F}}(\mathbf{u}) - \tilde{\mathbf{y}}||^2 \tag{3.19}$$

has a unique solution. Additionally, we assume that $\tilde{\mathcal{F}}$ is continuously differentiable and the Jacobian $\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{u})$ is rank n for all $\mathbf{u}$ in the domain of $\tilde{\mathcal{F}}$. The first order optimality condition is then given by

$$\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP})^\intercal(\tilde{\mathbf{y}} - \tilde{\mathcal{F}}(\bar{\mathbf{u}}_{MAP})) = \mathbf{0}. \tag{3.20}$$

The QR-factorisation of $\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP})$ is

$$\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP}) = \begin{pmatrix} \bar{\mathbf{Q}} & \tilde{\mathbf{Q}} \end{pmatrix} \begin{pmatrix} \bar{\mathbf{R}} \\ \mathbf{0} \end{pmatrix}.$$

The columns of $\tilde{\mathbf{Q}} \in \mathbb{R}^{(m+n)\times n}$ are orthonormal basis vectors for the orthogonal complement of the column space of $\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP})$. The matrix $\bar{\mathbf{R}} \in \mathbb{R}^{n\times n}$ is upper triangular and $\mathbf{0} \in \mathbb{R}^{m\times n}$ is the zero matrix. The thin QR-factorisation of $\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP})$ is then given as $\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP}) = \bar{\mathbf{Q}}\bar{\mathbf{R}}$. We make one more assumption about $\tilde{\mathcal{F}}$, which will be important later. We assume that $\bar{\mathbf{Q}}^\intercal\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{u})$ is invertible for all $\mathbf{u}$ in the domain of $\bar{\mathbf{Q}}^\intercal\tilde{\mathcal{F}}(\mathbf{u})$.

Since $\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP})$ is rank n, $\bar{\mathbf{R}}$ is invertible. Then, the condition (3.20) implies

$$\bar{\mathbf{Q}}^\intercal(\tilde{\mathbf{y}} - \tilde{\mathcal{F}}(\bar{\mathbf{u}}_{MAP})) = \mathbf{0}. \tag{3.21}$$

Define

$$\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}(\mathbf{u}) = \bar{\mathbf{Q}}^\intercal\tilde{\mathcal{F}}(\mathbf{u}).$$

The range of $\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}$ is denoted by $B_{\tilde{\mathcal{F}}}$. Then (3.20) is equivalent to writing $\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}(\bar{\mathbf{u}}_{MAP}) = \bar{\mathbf{Q}}^\intercal\tilde{\mathbf{y}}$. By assumption, $\bar{\mathbf{u}}_{MAP}$ is the unique solution of the problem (2.30), and so $\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}$ has an inverse at $\bar{\mathbf{Q}}^\intercal\tilde{\mathbf{y}}$, denoted by $\tilde{\mathcal{F}}^{-1}_{\bar{\mathbf{u}}_{MAP}}(\bar{\mathbf{Q}}^\intercal\tilde{\mathbf{y}}) = \bar{\mathbf{u}}_{MAP}$.

Since $\tilde{\mathcal{F}}$ is continuously differentiable with respect to $\mathbf{u}$ and $\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}} : \mathbb{R}^n \to \mathbb{R}^n$, the inverse function theorem guarantees that $\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}$ is invertible in a neighborhood of $\bar{\mathbf{Q}}^\intercal\tilde{\mathbf{y}}$. We can then define a random variable using the inverse map $\tilde{\mathcal{F}}^{-1}_{\bar{\mathbf{u}}_{MAP}}$. Let

$$\mathbf{V} = \tilde{\mathcal{F}}^{-1}_{\bar{\mathbf{u}}_{MAP}}(\mathbf{S}), \quad \text{where } \mathbf{S} = \bar{\mathbf{Q}}^\intercal(\tilde{\mathbf{y}} + \mathbf{W}) \tag{3.22}$$

and $\mathbf{W} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Then, $\mathbf{S}$ has the probability density function

$$\pi_{\mathbf{S}}(\mathbf{s}) \propto \exp\left(-\frac{1}{2}||\mathbf{s} - \bar{\mathbf{Q}}^\intercal\tilde{\mathbf{y}}||^2\right).$$

To ensure that $\mathbf{S}$ is supported only in the range of $\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}$, we define another random variable, $\mathbf{S}'$, with the density function

$$\pi_{\mathbf{S}'}(\mathbf{s}') \propto \mathbb{1}_{B_{\tilde{\mathcal{F}}}}(\mathbf{s}')\pi_{\mathbf{S}}(\mathbf{s}').$$

We can thus replace (3.22) with

$$\mathbf{V} = \tilde{\mathcal{F}}^{-1}_{\bar{\mathbf{u}}_{MAP}}(\mathbf{S}'), \quad \text{where } \mathbf{S}' \text{ has the probability density function } \pi_{\mathbf{S}'}(\mathbf{s}'). \tag{3.23}$$

Since $\bar{\mathbf{Q}}^\intercal\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{u})$ is invertible for all $\mathbf{u}$ in the domain of $\bar{\mathbf{Q}}^\intercal\tilde{\mathcal{F}}(\mathbf{u})$ by assumption, the inverse function theorem guarantees that function $\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}$ is one-to-one. Therefore, the mapping in (3.23) is well-defined. We can now derive the probability density $\pi_K(\mathbf{v})$ of the random variable $\mathbf{V}$ defined by (3.23). Following the theory of transformations of multivariate random variables, we have

$$\begin{aligned}
\pi_K(\mathbf{v}) &= |\mathbf{J}_{\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}}(\mathbf{v})|\pi_{\mathbf{S}'}(\tilde{\mathcal{F}}_{\bar{\mathbf{u}}_{MAP}}(\mathbf{v})) \\
&\propto |\bar{\mathbf{Q}}^\intercal\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{v})| \exp\left(-\frac{1}{2}||\bar{\mathbf{Q}}^\intercal(\tilde{\mathcal{F}}(\mathbf{v}) - \tilde{\mathbf{y}})||^2\right) \\
&= |\bar{\mathbf{Q}}^\intercal\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{v})| \exp\left(\frac{1}{2}||\tilde{\mathbf{Q}}^\intercal(\tilde{\mathcal{F}}(\mathbf{v}) - \tilde{\mathbf{y}})||^2 - \frac{1}{2}||\tilde{\mathcal{F}}(\mathbf{v}) - \tilde{\mathbf{y}}||^2\right) \\
&= c(\mathbf{v})\pi^{\mathbf{y}}_{\mathbf{U}}(\mathbf{v})
\end{aligned} \tag{3.24}$$

with the posterior density $\pi^{\mathbf{y}}_{\mathbf{U}}$ as given in (3.16) and

$$\begin{aligned}
c(\mathbf{v}) &= |\bar{\mathbf{Q}}^\intercal\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{v})| \exp\left(\frac{1}{2}||\tilde{\mathbf{Q}}^\intercal(\tilde{\mathcal{F}}(\mathbf{v}) - \tilde{\mathbf{y}})||^2\right) \\
&= |\bar{\mathbf{Q}}^\intercal\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{v})| \exp\left(\frac{1}{2}||\tilde{\mathcal{F}}(\mathbf{v}) - \tilde{\mathbf{y}}||^2 - \frac{1}{2}||\bar{\mathbf{Q}}^\intercal(\tilde{\mathcal{F}}(\mathbf{v}) - \tilde{\mathbf{y}})||^2\right).
\end{aligned} \tag{3.25}$$

The assumptions and above results are summarised in Theorem 3.5.1 (Theorem 3.1 in [48]).

Theorem 3.5.1: Randomise-then-optimise validity conditions

Let $\mathbf{V}$ be the random variable defined by (3.18) and suppose the following conditions are fulfilled:

(1) the problem $\underset{\mathbf{u}\in\mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2}||\tilde{\mathcal{F}}(\mathbf{u}) - \tilde{\mathbf{y}}||^2$ has a unique solution $\bar{\mathbf{u}}_{MAP}$,

(2) the function $\tilde{\mathcal{F}}$ is continuously differentiable,

(3) the Jacobian $\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{u})$ is rank n for all $\mathbf{u}$ in the domain of $\tilde{\mathcal{F}}$,

(4) and $\bar{\mathbf{Q}}^\intercal \mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{u})$ is invertible for all $\mathbf{u}$ in the domain of $\bar{\mathbf{Q}}^\intercal \tilde{\mathcal{F}}(\mathbf{u})$.

Then $\mathbf{V}$ has the density (3.24).

We then set the acceptance probability function

$$\alpha(\mathbf{u_k}, \mathbf{v}) = \min\left\{1, \frac{c(\mathbf{u_k})}{c(\mathbf{v})}\right\} \tag{3.26}$$

to fulfill the balance condition (3.3).

The next chapter applies the concepts discussed in Chapters 2 and 3 to the one-dimensional deconvolution problem to prepare for MCMC sampling.

Chapter 4

The posterior density

Section 4.1 covers the derivation of the posterior distribution using Bayes' formula (2.26), Section 4.2 presents the algorithms used to sample from the posterior distribution.

4.1 Bayesian inversion

In this section, we obtain the solution of our 1D deconvolution problem. Three priors from the prior classes described in Section 2.3 are constructed. The first prior is a Gaussian prior which encodes assumptions about the smoothness of the true signal. The second prior is the total variation prior for one-dimensional functions. The third prior is the Besov prior (2.45) truncated to n terms. We then derive the likelihood of the 1D deconvolution problem and apply Bayes' Theorem for inverse problems (Theorem 2.2.1) to obtain three posterior density functions.

Together with the total variation and Besov priors, we present two non-linear transformations from the non-Gaussian priors to standard Gaussian random variables. These transformations enable the application of the methods presented in Chapter 3 to sample from linear inverse problems governed by TV and Besov priors. The MCMC methods in Chapter 3 were originally formulated for Gaussian priors [28], [48]. By applying prior transformations, these MCMC methods can be modified for sampling from posterior densities obtained with non-Gaussian priors.

4.1.1 Gaussian smoothness prior

In 1D deconvolution, we want to suppress measurement noise and deconvolve the function. This can be done by modeling the smoothness properties of a continuous function. The first derivative of u is discretised and each point of this discretised derivative is modeled as a normally-distributed random variable. The expression for the smoothness model is given by

$$\frac{1}{\Delta x}(\mathbf{U}_j - \mathbf{U}_{j-1}) = \mathbf{Z}_j \tag{4.1}$$

for $j = 1, 2, \ldots, n-1$, where $\{\mathbf{Z}_j\}_{j=0}^{n-1}$ are i.i.d. $\mathcal{N}\left(0, \sigma_u^2\right), \sigma_u > 0$. To apply the discretisation (4.1) to the entire domain $[0, 1)$, we set $\mathbf{U}_n = \mathbf{U}_0 = 0$ and define $\mathbf{D}$ as in (2.24).

The prior density function of $\mathbf{U}$ is then

$$\pi_{\mathbf{U}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2\sigma_u^2}\left\langle \mathbf{u}, \mathbf{D}^\intercal \mathbf{D}\mathbf{u}\right\rangle\right). \tag{4.2}$$

We henceforth refer to the random variable $\mathbf{U}$ with the density (4.2) as the Gaussian smoothness prior with the standard deviation σ_u.

4.1.2 Total variation prior

We define the total variation prior on $\mathbf{U}$ as given in Definition 2.3.7

$$\pi_{\mathbf{U}}(\mathbf{u}) \propto \exp\left(-\lambda \sum_{j=1}^{n} |(\mathbf{D}\mathbf{u})_j|\right) \tag{4.3}$$

where $\lambda > 0$ and $\mathbf{D}$ is as defined in (2.24). We henceforth refer to the random variable $\mathbf{U}$ with the density (4.3) as the TV smoothness prior.

The transformation from the TV prior to a standard Gaussian random variable is introduced in [31]. Let $\Phi(z)$ denote the standard Gaussian cumulative distribution function, i.e. the c.d.f of $\mathbf{Z} \sim \mathcal{N}(0, 1)$. Define

$$g_\lambda(z) = -\frac{1}{\lambda}\operatorname{sgn}(z)\log\left(1 - 2\left|\Phi(z) - \frac{1}{2}\right|\right),$$

which transforms a standard one-dimensional Gaussian random variable $\mathbf{Z}$ to a Laplace-distributed random variable. The TV prior is on the derivative. We thus write

$$\mathbf{D}\mathbf{u} = G_\lambda(\mathbf{z})$$

with $\mathbf{D}$ being the difference matrix (2.24) and $G_\lambda = \begin{pmatrix} g_\lambda(\mathbf{z}_0) & g_\lambda(\mathbf{z}_1) & \ldots & g_\lambda(\mathbf{z}_{n-2}) & g_\lambda(\mathbf{z}_{n-1}) \end{pmatrix}^\intercal$. The TV prior transformation $T_\lambda : \mathbb{R}^n \to \mathbb{R}^n$ is then

$$T_\lambda(\mathbf{z}) = \mathbf{D}^{-1} G_\lambda(\mathbf{z}). \tag{4.4}$$

The transformation (4.4) is the composition of a linear operator $\mathbf{D}^{-1}$ and a non-linear function G_λ. The transformation T_λ is continuously differentiable and invertible (see Appendix B).

4.1.3 Besov priors

The 1D Besov prior (2.45) is a random process defined as the infinite sum of orthonormal basis functions for $L^2(\mathbb{T})$ [27], [38], [41]. For practical purposes, the series (2.45) must be truncated to a finite number of terms. We now derive the density of the discretised Besov prior used in 1D deconvolution.

Let $J \in \mathbb{Z}$ be such that $n = 2^J$. Let $\mathcal{K}_j = \{0, 1, \ldots, 2^j - 1\}$ for all $j = 0, 1, \ldots, J$. Let $\mathbf{W} \in \mathbb{R}^{n \times n}$ be the matrix whose columns are the wavelet coefficients associated with the standard basis vectors for $\mathbb{R}^n$.

In this project, the fast discrete wavelet transform [43], [49] is done using the `wavedec` function from the `PyWavelets` [50] package in Python. We set the `mode` as `'periodization'`, as $\psi_j(x)$ is the periodisation of the wavelet basis function.

The Haar and Daubechies-8 wavalets are used in this project, following the choices of these wavelet functions in [41]. The Haar wavelet function ψ is constructed from the composition of step functions (see Figure 2.8a). As such, the Haar wavelet basis functions possess local discontinuities. The Daubechies-8 wavelets are smooth, compact wavelets (see Figure 2.8b). As our true signal (Figure 2.2) contains discontinuities and smooth regions, we wish to see whether one wavelet family can represent the features of the true signal more accurately than the other.

Let the diagonal matrix $\mathbf{S} \in \mathbb{R}^{n\times n}$ of Besov weights be given by

$$\mathbf{S}_{1,1} = 1, \mathbf{S}_{l,l} = 2^{jk}$$

for $2^j \leq l < 2^{j+1}$ and for $j = 0, 1, \ldots, J-1$.

The n-term truncation of the Besov random variable (2.45) is given by

$$\mathbf{SWU} = \begin{pmatrix} \xi_0 + \sum_{j=0}^{J} \sum_{k\in\mathcal{K}_j} \delta^{-\frac{1}{q}} \gamma_j \xi_{j,k} \psi_j(\mathbf{x}_0 - k2^{-j}) \\ \xi_0 + \sum_{j=0}^{J} \sum_{k\in\mathcal{K}_j} \delta^{-\frac{1}{q}} \gamma_j \xi_{j,k} \psi_j(\mathbf{x}_1 - k2^{-j}) \\ \vdots \\ \xi_0 + \sum_{j=0}^{J} \sum_{k\in\mathcal{K}_j} \delta^{-\frac{1}{q}} \gamma_j \xi_{j,k} \psi_j(\mathbf{x}_{n-2} - k2^{-j}) \\ \xi_0 + \sum_{j=0}^{J} \sum_{k\in\mathcal{K}_j} \delta^{-\frac{1}{q}} \gamma_j \xi_{j,k} \psi_j(\mathbf{x}_{n-1} - k2^{-j}) \end{pmatrix}. \tag{4.5}$$

Then, as in Appendix C of [31] and Section 2.2 of [41], the discrete approximation of the Besov norm (2.47) is given by

$$||U||_{B^s_{qq}(\mathbb{T})} \approx ||\mathbf{SWU}||_q$$

and we write the discrete Besov prior density

$$\pi_{\mathbf{U}}(\mathbf{u}) \propto \exp\left(-\kappa||\mathbf{SWu}||_q^q\right), \tag{4.6}$$

where $\kappa = \frac{\delta}{2}$ as defined in (2.48). A transformation can now be defined to map a Besov prior to a standard Gaussian prior.

The generalised Gaussian distribution [51] parametrised by τ and q is the distribution with the p.d.f.

$$\pi_{\tau,q}(x) = \frac{q}{2\tau\Gamma\left(\frac{1}{q}\right)} \exp\left(-\left|\frac{x}{\tau}\right|^q\right) \tag{4.7}$$

and the c.d.f.

$$\Phi_{\tau,q}(x) = \begin{cases} \frac{1}{2}Q\left(\frac{1}{q}, \left(-\frac{x}{\tau}\right)^q\right) & x \leq 0 \\ 1 - \frac{1}{2}Q\left(\frac{1}{q}, \left(\frac{x}{\tau}\right)^q\right) & x > 0. \end{cases}$$

Here, $Q(a, x) = \left(\frac{1}{\Gamma(a)}\Gamma(a, x)\right)$ denotes the normalised incomplete gamma function, with $\Gamma(a, x) = \int_x^\infty t^{a-1} \exp(-t)\, dt$. The generalised Gaussian distribution extends the Gaussian and Laplace distributions; when $q = 1$, we obtain the Laplace distribution and when $q = 2$, we obtain the Gaussian distribution. Besov priors with $q = 1$ model similar behavior to the total variation prior, and so we use $q = 1$ below unless otherwise specified. This value is chosen so that deconvolution results obtained using the Besov prior can be compared to results obtained using the total variation prior.

Using an inverse cumulative distribution function method, we can define a function that transforms a standard Gaussian random variable to a generalised Gaussian random variable. Let this function be

$$\begin{aligned} g_{\tau,q}(z) &= \Phi_{\tau,q}^{-1}(\Phi(z)) \\ &= \tau \mathrm{sgn}(z) \left(Q^{-1} \left(\frac{1}{q}, 1 + \mathrm{sgn}(z) - 2\mathrm{sgn}(z)\Phi(z) \right) \right)^{\frac{1}{q}} \end{aligned} \tag{4.8}$$

where $\Phi(z)$ is the standard Gaussian c.d.f. and $Q^{-1}(a, y)$ is the inverse normalised incomplete gamma function. By setting $\tau = \kappa^{-\frac{1}{q}}$, we transform a standard Gaussian random variable z to a random variable $\xi = g_{\tau,q}(z)$ with

$$\pi(\xi) \propto \exp\left(-\kappa|\xi|^q\right).$$

We thus write

$$\mathbf{SWu} = G_{\tau,q}(\mathbf{z})$$

with $G_{\tau,q} = \begin{pmatrix} g_{\tau,q}(\mathbf{z}_0) & g_{\tau,q}(\mathbf{z}_1) & \dots & g_{\tau,q}(\mathbf{z}_{n-2}) & g_{\tau,q}(\mathbf{z}_{n-1}) \end{pmatrix}^{\mathsf{T}}$. The Besov prior transformation is then

$$T_{\tau,q}(\mathbf{z}) = (\mathbf{SW})^{-1} G_{\tau,q}(\mathbf{z}). \tag{4.9}$$

Note that $\mathbf{W}^{-1}$ is the matrix representation of the inverse wavelet transform and $\mathbf{S}$ is a diagonal matrix, hence it is invertible. Therefore, $(\mathbf{SW})^{-1}$ exists. The transformation (4.9) is the composition of a linear operator $(\mathbf{SW})^{-1}$ and a non-linear function $G_{\tau,q}$. The transformation $T_{\tau,q}$ is invertible (see Appendix B).

4.1.4 Likelihood density

Assuming the noise at each grid point $\mathbf{x}_j$ is independent and normally distributed with mean 0 and standard deviation $\sigma_{\mathbf{E}}$, we can model the noise as an $\mathbb{R}^m$-valued Gaussian random variable with mean 0 and covariance $\Gamma_{\mathbf{E}} = \sigma_{\mathbf{E}}^2 \mathbf{I}$. Our likelihood density function is then

$$\pi_{\mathbf{E}}(\mathbf{y} - \mathbf{Au}) \propto \exp\left(-\frac{1}{2\sigma_{\mathbf{E}}^2} ||\mathbf{y} - \mathbf{Au}||^2\right).$$

4.1.5 Posterior densities

By applying Bayes' formula (2.27), we obtain three posterior density functions of $\mathbf{U}$ when the priors described in Sections 4.1.1-4.1.3 are used. The posterior densities obtained using the Gaussian smoothness prior, the TV prior, and the discrete Besov prior are

$$\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2\sigma_{\mathbf{E}}^2} ||\mathbf{y} - \mathbf{Au}||^2 - \frac{1}{2\sigma_u^2} \mathbf{u}^{\mathsf{T}} \mathbf{D}^{\mathsf{T}} \mathbf{Du}\right), \tag{4.10}$$

$$\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2\sigma_{\mathbf{E}}^2} ||\mathbf{y} - \mathbf{Au}||^2 - \lambda \sum_{j=1}^{n} |(\mathbf{Du})_j|\right), \tag{4.11}$$

and

$$\pi^{\mathbf{y}}_{\mathbf{U}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2\sigma^2_{\mathbf{E}}}||\mathbf{y}-\mathbf{A}\mathbf{u}||^2 - \kappa||\mathbf{S}\mathbf{W}\mathbf{u}||_q^q\right) \tag{4.12}$$

respectively, with $\mathbf{A}$ being the discrete convolution matrix (2.21).

4.2 Algorithms

The sampling methods discussed in Section 3.2 are implemented in Python with the library `JAX` [52]. In Subsection 4.2.1, we present Algorithm 1 for pCN sampling from a posterior density obtained using a Gaussian prior and Algorithm 2 for pCN sampling from a posterior density obtained using a non-Gaussian prior. In Subsection 4.2.2, we present Algorithm 3 for RTO sampling.

4.2.1 Preconditioned Crank-Nicolson

An expression of the potential function ϕ, as defined in (2.28), is needed for the implementation of the preconditioned Crank-Nicolson sampling method. The potential in the posterior density (4.10) is

$$\phi(\mathbf{u}) = \frac{1}{2\sigma^2_{\mathbf{E}}}||\mathbf{y}-\mathbf{A}\mathbf{u}||^2. \tag{4.13}$$

We present the pCN algorithm [28] for sampling from the posterior density (4.10) with the prior density (4.2).

Algorithm 1 pCN algorithm (with Gaussian prior)

```
Choose initial draw $\mathbf{u_0}$, fix $\beta \in (0,1]$.
for $k \in \{0, 2, \dots, M-1\}$ do
    procedure Generate proposal($\beta, \sigma_u^2(\mathbf{D}^\intercal\mathbf{D})^{-1}, \mathbf{u_k}$)
        Draw a realisation $\mathbf{w}$ from $\mathbf{W} \sim \mathcal{N}\left(\mathbf{0}, \sigma_u^2(\mathbf{D}^\intercal\mathbf{D})^{-1}\right)$
        Compute proposal $\mathbf{v} \leftarrow \sqrt{1-\beta^2}\mathbf{u_k} + \beta\mathbf{w}$
    end procedure
    procedure Accept-reject($\mathbf{u_k}, \mathbf{v}, \phi(\cdot)$)
        Compute acceptance probability $\alpha(\mathbf{u_k}, \mathbf{v}) \leftarrow \min\{1, \exp(\phi(\mathbf{u_k}) - \phi(\mathbf{v}))\}$
        Draw $\mathbf{a} \sim \text{Unif}(0,1)$.
        if $\alpha(\mathbf{u_k}, \mathbf{v}) > \mathbf{a}$ then
            $\mathbf{u_{k+1}} \leftarrow \mathbf{v}$
        else
            $\mathbf{u_{k+1}} \leftarrow \mathbf{u_k}$
        end if
    end procedure
end for
```

Algorithm 1 can be modified to sample from the densities (4.11) and (4.12) by a change in variables [30]. Let $T = T_\lambda$ (4.4) or $T = T_{\tau,q}$ (4.9). By Proposition B.1, posterior probability density function as defined in (4.11) or (4.12) can be rewritten as a function of $\mathbf{z}$,

$$\begin{aligned}\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{z}) &\propto \exp\left(-\frac{1}{2}||\mathbf{A}(T(\mathbf{z}))-\mathbf{y}||^2-\frac{1}{2}||\mathbf{z}||^2\right)\\ &= \exp\left(-\phi(T(\mathbf{z}))-\frac{1}{2}||\mathbf{z}||^2\right). \qquad (4.14)\end{aligned}$$

Note $\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{z})$ is the Lebesgue density of a measure $\mu_{\mathbf{U}}^{\mathbf{y}}$ with the Radon-Nikodym derivative

$$\frac{d\mu_{\mathbf{U}}^{\mathbf{y}}}{d\mu_{\mathbf{Z}}}(\mathbf{z}) = \exp\left(-\phi(T(\mathbf{z}))\right),$$

where $\mu_{\mathbf{Z}}$ has the Lebesgue density $\pi_{\mathbf{Z}}(\mathbf{z}) \propto \exp\left(-\frac{1}{2}||\mathbf{z}||^2\right)$. We can thus sample from $\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{z})$ using Algorithm 2, presented below.

Algorithm 2 Whitened pCN algorithm (with non-Gaussian prior)

```
Choose initial draw $\mathbf{z_0}$, fix $\beta \in (0,1]$.
for $k \in \{0,2,\dots,M-1\}$ do
    procedure GENERATE PROPOSAL($\beta, \mathbf{z_k}$)
        Draw a realisation $\mathbf{w}$ from $\mathbf{W} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$
        Compute proposal $\mathbf{v} \leftarrow \sqrt{1-\beta^2}\mathbf{z_k} + \beta\mathbf{w}$
    end procedure
    procedure ACCEPT-REJECT($\mathbf{z_k}, \mathbf{v}, \phi(\cdot), T(\cdot)$)
        Compute acceptance probability $\alpha(\mathbf{z_k},\mathbf{v}) \leftarrow \min\{1, \exp(\phi(T(\mathbf{z_k})) - \phi(T(\mathbf{v})))\}$
        Draw $\mathbf{a} \sim \mathrm{Unif}(0,1)$.
        if $\alpha(\mathbf{z_k},\mathbf{v}) > \mathbf{a}$ then
            $\mathbf{u_{k+1}} \leftarrow \mathbf{v}$
        else
            $\mathbf{u_{k+1}} \leftarrow \mathbf{z_k}$
        end if
    end procedure
end for
```

When Algorithm 2 is used, the samples $\{\mathbf{z_k}\}_{k=1}^{K}$ have to be transformed to the non-Gaussian random variables. The collection of samples is then $\{\mathbf{u_k}\}_{k=1}^{K} = \{T(\mathbf{z_k})\}_{k=1}^{K}$.

4.2.2 Randomise-then-optimise

Recall that, to implement RTO, we need the posterior density to be in the form (3.17). When the prior is the Gaussian smoothness prior (4.2), this can be achieved by letting

$$\tilde{\mathcal{F}}(\mathbf{u}) = \begin{pmatrix}\frac{1}{\sigma_u}\mathbf{D}^{\intercal}\mathbf{D}\mathbf{u}\\ \frac{1}{\sigma_{\mathbf{E}}^2}\mathbf{A}\mathbf{u}\end{pmatrix}, \quad \tilde{\mathbf{y}} = \begin{pmatrix}\mathbf{0}\\ \frac{1}{\sigma_{\mathbf{E}}^2}\mathbf{y}\end{pmatrix}.$$

To sample from the posterior distributions (4.11) and (4.12), we ensure the posterior density has the RTO form by applying prior transformations [31]. Let T be a continuously differentiable and invertible transformation of an n-variate standard Gaussian random variable $\mathbf{Z}$ to $\mathbf{U}$, a $\mathbb{R}^n$-valued random variable with the density (4.3) or (4.6) (i.e. let $T = T_\lambda$ (4.4) or $T = T_{\tau,q}$ (4.9).) By Proposition B.1, the posterior probability density function can be rewritten as a function of $\mathbf{z}$,

$$\begin{aligned} \pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{z}) &\propto \exp\left(-\frac{1}{2}||\mathbf{A}(T(\mathbf{z}))-\mathbf{y}||^2 - \frac{1}{2}||\mathbf{z}||^2\right) \\ &= \exp\left(\frac{1}{2}||\tilde{\mathcal{F}}(\mathbf{z})-\tilde{\mathbf{y}}||^2\right) \end{aligned} \tag{4.15}$$

where

$$\tilde{\mathcal{F}}(\mathbf{z}) = \begin{pmatrix} \mathbf{z} \\ \frac{1}{\sigma_{\mathbf{E}}^2}\mathbf{A}(T(\mathbf{z})) \end{pmatrix} \text{ and } \tilde{\mathbf{y}} = \begin{pmatrix} 0 \\ \frac{1}{\sigma_{\mathbf{E}}^2}\mathbf{y} \end{pmatrix}. \tag{4.16}$$

The proof is included in Appendix B. The RTO validity conditions in Theorem 3.5.1 are checked for RTO with a prior transformation in Appendix B.

The non-linear prior transformation is included in the objective function of the stochastic optimisation problem (3.18). As such, only one randomise-then-optimise algorithm is needed to sample from the posterior densities in Subsection 4.1.5. The approximate Jacobian is computed using the automatic differentiation features in the library `JAX`.

Algorithm 3 RTO-MH algorithm

```
Choose initial draw $\mathbf{u_0}$, compute $\bar{\mathbf{Q}}$.
for $k \in \{0, 2, \dots, K-1\}$ do
    procedure GENERATE PROPOSAL($\bar{\mathbf{Q}}, \tilde{\mathcal{F}}(\cdot), \tilde{\mathbf{y}}$)
        Draw a realisation $\mathbf{w}$ from $\mathbf{W} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
        Compute proposal $\mathbf{v}_k \leftarrow \underset{\mathbf{v}\in\mathbb{R}^n}{\operatorname{argmin}} ||\bar{\mathbf{Q}}^\intercal(\tilde{\mathcal{F}}(\mathbf{v}) - (\tilde{\mathbf{y}} + \mathbf{W}))||^2$
    end procedure
end for
for $k \in \{0, 2, \dots, M-1\}$ do
    procedure ACCEPT-REJECT($\mathbf{u_k}, \mathbf{v}, c(\cdot)$)
        Compute acceptance probability $\alpha(\mathbf{u_k}, \mathbf{v_k}) \leftarrow \min\left\{1, \frac{c(\mathbf{u_k})}{c(\mathbf{v_k})}\right\}$ with $c$ as defined in (3.25).
        Draw $\mathbf{a} \sim \text{Unif}(0, 1)$.
        if $\alpha(\mathbf{u_k}, \mathbf{v_k}) > \mathbf{a}$ then
            $\mathbf{u_{k+1}} \leftarrow \mathbf{v_k}$
        else
            $\mathbf{u_{k+1}} \leftarrow \mathbf{u_k}$
        end if
    end procedure
end for
```

When Algorithm 3 is used with a prior transformation, the samples $\{\mathbf{z_k}\}_{k=1}^{K}$ have to be transformed to the non-Gaussian random variables. We return $\{T(\mathbf{u_k})\}_{k=1}^{K}$. In the next chapter, we present sampling results from the posterior densities derived in Section 4.1. We sample using algorithms in Section 4.2, implemented in the Python library `JAX`, with automatic differentiation used to compute approximate Jacobians. Optimisation problems are solved using the Adam [53] optimiser implemented in `optax` [54] with a tolerance level of 1×10^{-6}. To find the MAP estimator (3.19), $\mathbf{u}_{MAP}$, the maximum number of optimiser iterations is 1×10^5. To solve the RTO proposal optimisation problem (3.18), the maximum number of optimiser iterations is 1×10^3.

Chapter 5

Numerical results

In this chapter, we present results obtained using the sampling methods described Chapter 3 for the one-dimensional deconvolution problem. The measurement $\mathbf{y}$ is given in Section 5.1. Parameter choices for the priors and sampling methods are made in Section 5.2. Sampling results with total variation priors are shown in Section 5.3. Sampling results with Besov priors are shown in Section 5.4. In Section 5.5, summaries of error metrics and measures of computational efficiency are shown and discussed. The effect of problem dimensions are briefly studied in Section 5.6.

5.1 True signal and measurement

In our 1D deconvolution problem, the linear forward operator is the discrete convolution matrix $\mathbf{A}$ (2.21) constructed using the quartic PSF (2.12). The noise $\mathbf{E}$ is modeled as an n-variate centered Gaussian distribution with the covariance matrix $\mathbf{\Gamma_E} = \sigma_{\mathbf{E}}^2\mathbf{I}$ with $\sigma_{\mathbf{E}} = 0.02$.

When synthesising data, it is important to keep consider mind that it is unrealistic to assume we know the exact PSF and PSF parameter a in real-life deconvolution problems. Overlooking these issues may lead to committing an *inverse crime*. An inverse crime occurs when synthetic data is generated using the same theoretical information we incorporate into our formulation of our inverse problem. When we commit an inverse crime and apply a method to solve an inverse problem, we may obtain a very close approximation to the true signal which do not reflect how well our method would perform on problems where we do not have perfect knowledge of how the observations were generated.

To avoid committing an inverse crime, the observation $\mathbf{y}$ is generated as follows. We set n such that our discretised function $\mathbf{u}$ will be in $\mathbb{R}^n$. We then define $n_1 > n$, where n_1 is the largest integer such that $n_1 \leq 5.25n < n_1 + 1$ and discretise our true signal u on an n_1-point grid. Then, we choose $a = 0.05 > 0$ and construct a discrete convolution matrix as in (2.21) using the quartic PSF (2.18) parametrised by $a_1 = 1.05a$. Discrete convolution is then performed on the fine n_1-point grid and linear interpolation is used to sample the result on the coarser n-point grid, yielding noiseless data $\mathbf{y}^*$. Measurement noise $\mathbf{e}$ is constructed by taking a realisation of an n-variate Gaussian random variable with mean zero and

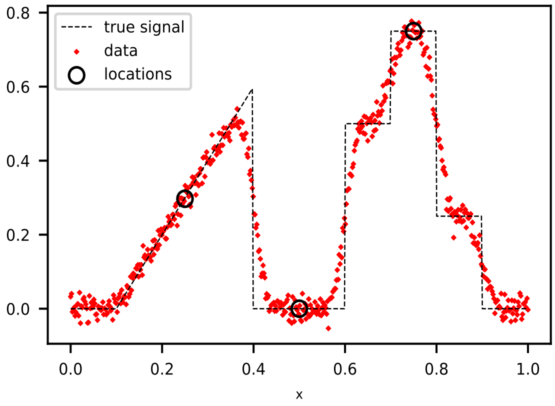


Figure 5.1: True signal and data. The relative error of the observation $\mathbf{y}$ is $E_{L2}(\mathbf{y}) = 0.20245$.

convariance matrix $\sigma_{\mathbf{E}}^2 \cdot \mathbf{I}$. The synthetic observation is

$$\mathbf{y} = \mathbf{y}^* + \mathbf{e}. \tag{5.1}$$

The data points are plotted together with the true signal in Figure 5.1. We aim to recover the true signal, marked in black, from the red data points plotted in Figure 5.1. The data and true signal are shown in Figure 5.1.

The x-coordinates marked by circles in Figure 5.1 are locations at which we plot the samples and auto-correlation functions (ACF). To obtain the sampling results in Sections 5.3 and 5.4, the initial draw $\mathbf{u_0}$ is $\begin{pmatrix} 0.4 & 0.4 & \dots & 0.4 & 0.4 \end{pmatrix}^\mathsf{T}$ is used. As in [41], 10^4 samples are generated.

5.2 Parameter choices

The standard deviation of the Gaussian smoothness prior σ_u (4.2), the total variation prior parameter λ (4.3), and the Besov prior parameter κ (4.6) need to be chosen. In Subsection 5.2.1, we choose values of the RW step size parameter β and the prior parameters σ_u, λ and κ. In Subsection 5.2.2, we choose values of the pCN step size parameter β and the prior parameters σ_u, λ and κ. In Subsection 5.2.3, we choose the prior parameters σ_u, λ and κ for RTO sampling.

We search for prior parameter values which minimise the L2 error of the estimated conditional mean $\bar{\mathbf{u}}_{est}$ relative to the true signal,

$$E_{L2}(\bar{\mathbf{u}}_{est}) = \frac{||\bar{\mathbf{u}}_{est} - \mathbf{u}_{true}||_2}{||\mathbf{u}_{true}||_2} \tag{5.2}$$

where $||\cdot||_2^2$ is the Euclidean 2-norm in $\mathbb{R}^n$. This is not a realistic method for choosing the prior parameters σ_u, λ, and κ, as the relative L2 error is not available practical settings, where the true signal is not known. In this project, the choice to minimise the relative L2 error is motivated by our goal of comparing relatively accurate estimators obtained using the different methods.

For random walk and preconditioned Crank-Nicolson, the step size parameter β needs to be chosen. This value is chosen as to ensure that the average acceptance probability (average α) is not too small or too large. If the average α value is close to zero ($\ll 0.1$), this indicates that the sampling method does not generate good proposals. A large average α value (> 0.9) may indicate that the sampling method is not exploring the posterior distribution effectively, and only drawing proposals from a small subset of $\mathbb{R}^n$.

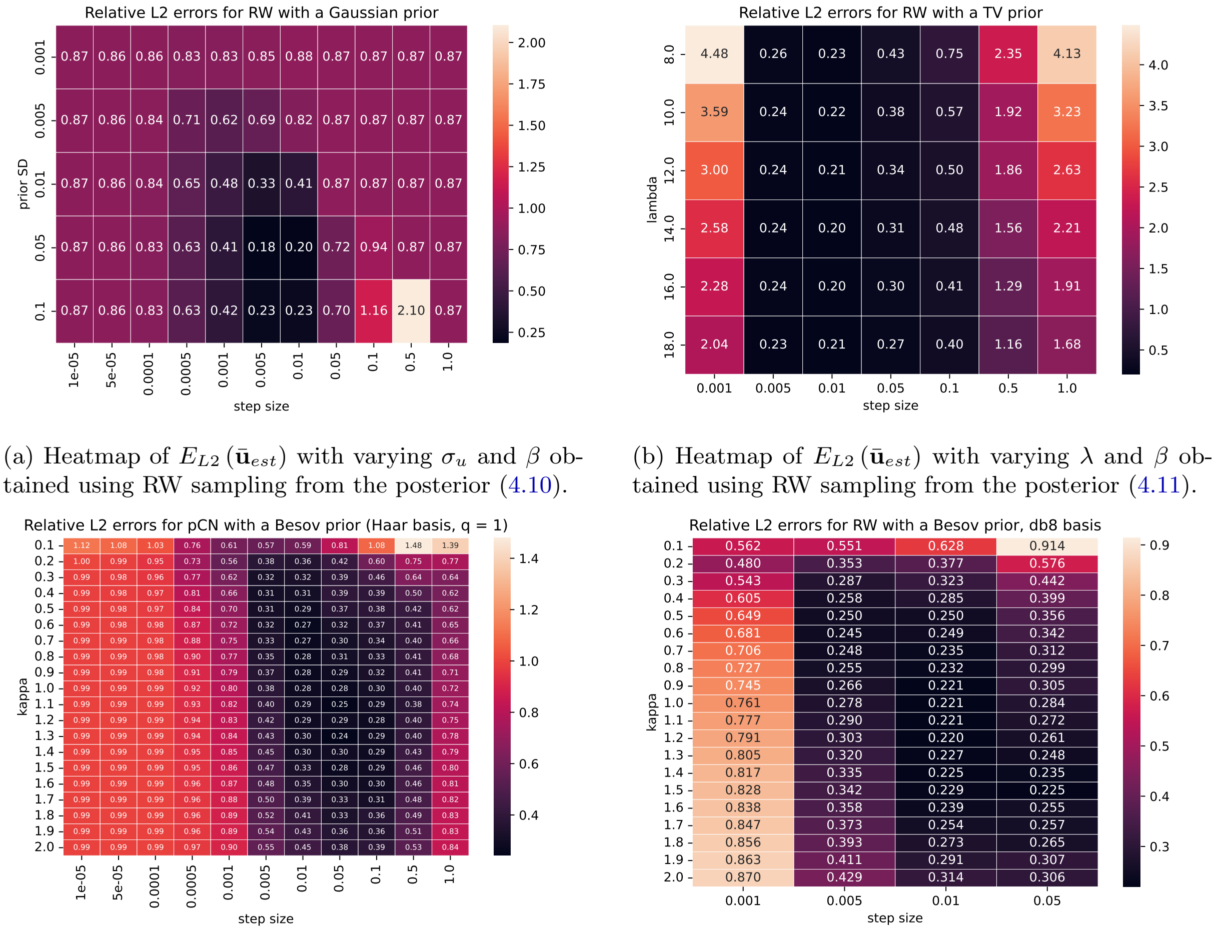
5.2.1 Parameter choices for random walk

Prior parameters

To choose the standard deviation σ_u for the Gaussian smoothness prior (4.2), the total variation prior parameter λ (4.3), and the Besov prior parameter κ (4.6) for RW, we perform grid searches over varying prior parameter values and step size β values. For each prior parameter value and β pair, we run the RW sampler for 10^4 iterations and compute the estimated conditional mean $\bar{\mathbf{u}}_{est}$ using the last 8000 samples. We show the heat maps obtained from our grid search in Figure 5.2 and summarise the chosen prior parameter values in Table 5.1.

Parameter	Parameter value	β	Average α	$E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$
σ_u	0.05	0.005	0.27791	0.18484
λ	16	0.01	0.19660	0.20314
κ, Haar	1.4	0.05	0.16475	0.27630
κ, db8	1.2	0.01	0.55279	0.21991

Table 5.1: Chosen prior parameter values for RW.



(a) Heatmap of $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ with varying σ_u and β obtained using RW sampling from the posterior (4.10).

(b) Heatmap of $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ with varying λ and β obtained using RW sampling from the posterior (4.11).

(c) Heatmap of $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ with varying κ and β obtained using RW sampling from the posterior (4.12) with $q = 1$ and the Haar wavelet basis functions.

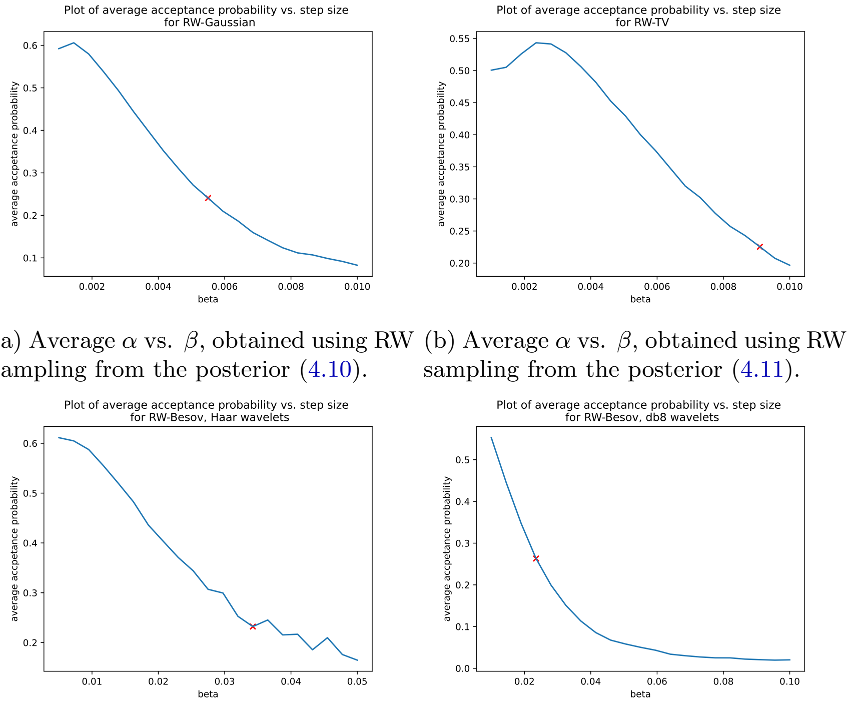
(d) Heatmap of $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ with varying κ and β obtained using RW sampling from the posterior (4.12) with $q = 1$ and the Daubechies-8 wavelet basis functions.

Figure 5.2: Heatmaps of $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ with varying prior parameters and β, obtained using RW.

Step size parameters

We continue tuning the step size parameter β. Following [28], we choose β such that the average acceptance probability is ≈ 0.234. This is a heuristic and not necessarily the optimal value of β.

We fix the prior parameter values in Table 5.1 and refine our grid search to tune the step size parameter β. For each grid search, we run the RW algorithm for 10^4 iterations and compute the average value of α over the iterations. The value of β minimising $|\alpha_{avg} - 0.234|$ is then chosen. In Table 5.1, the average α values ranged from 0.16475 to 0.55279. The finer grid search narrows down the average α values range to $0.22533 - 0.26311$.



(a) Average α vs. β, obtained using RW sampling from the posterior (4.10).

(b) Average α vs. β, obtained using RW sampling from the posterior (4.11).

(c) Average α vs. β, obtained using RW sampling from the posterior (4.12) with Haar wavelets and $q = 1$.

(d) Average α vs. β, obtained using RW sampling from the posterior (4.12) with Daubechies-8 wavelets and $q = 1$

Figure 5.3: Average acceptance probabilities vs. β.

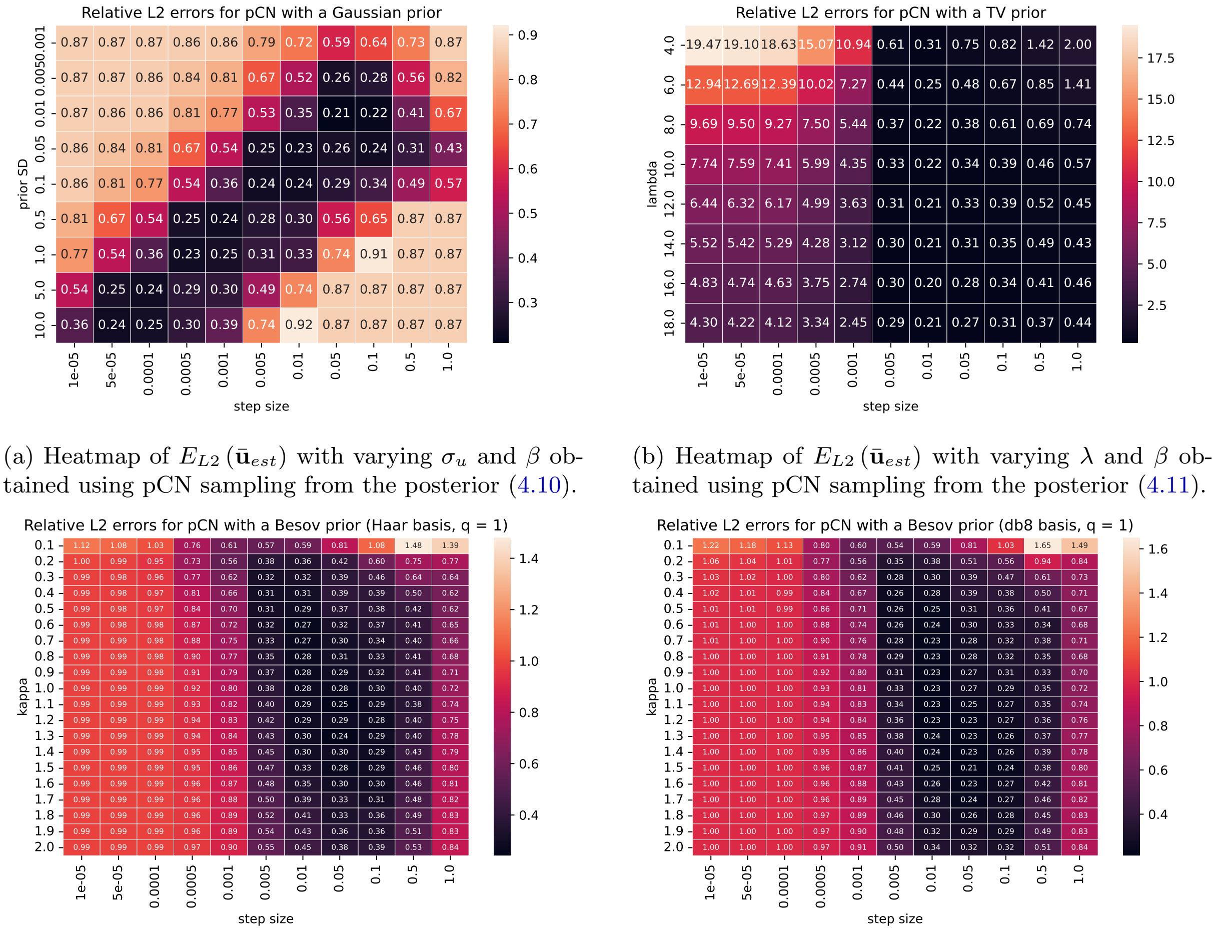
Parameter	Parameter value	β	Average α	$E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$
σ_u	0.05	0.0055	0.24083	0.18940
λ	16	0.0091	0.22533	0.19439
κ, Haar	1.4	0.0235	0.26311	0.22940
κ, db8	1.2	0.03425	0.23214	0.27246

Table 5.2: Values of β chosen for RW.

5.2.2 Parameter choices for preconditioned Crank-Nicolson

Prior parameters

To choose the standard deviation σ_u for the Gaussian smoothness prior (4.2), the total variation prior parameter λ (4.3), and the Besov prior parameter κ (4.6) for pCN, we perform grid searches over varying prior parameter values and step size β values. For each prior parameter value and β pair, we run the pCN sampler for 10^4 iterations and compute the estimated conditional mean $\bar{\mathbf{u}}_{est}$ using the last 8000 samples. We show the heat maps obtained from our grid search in Figure 5.4 and summarise the chosen prior parameter values in Table 5.3.



(a) Heatmap of $E_{L2}(\bar{\mathbf{u}}_{est})$ with varying σ_u and β obtained using pCN sampling from the posterior (4.10).

(b) Heatmap of $E_{L2}(\bar{\mathbf{u}}_{est})$ with varying λ and β obtained using pCN sampling from the posterior (4.11).

(c) Heatmap of $E_{L2}(\bar{\mathbf{u}}_{est})$ with varying κ and β obtained using pCN sampling from the posterior (4.12) with $q = 1$ and the Haar wavelet basis functions.

(d) Heatmap of $E_{L2}(\bar{\mathbf{u}}_{est})$ with varying κ and β obtained using pCN sampling from the posterior (4.12) with $q = 1$ and the Daubechies-8 wavelet basis functions.

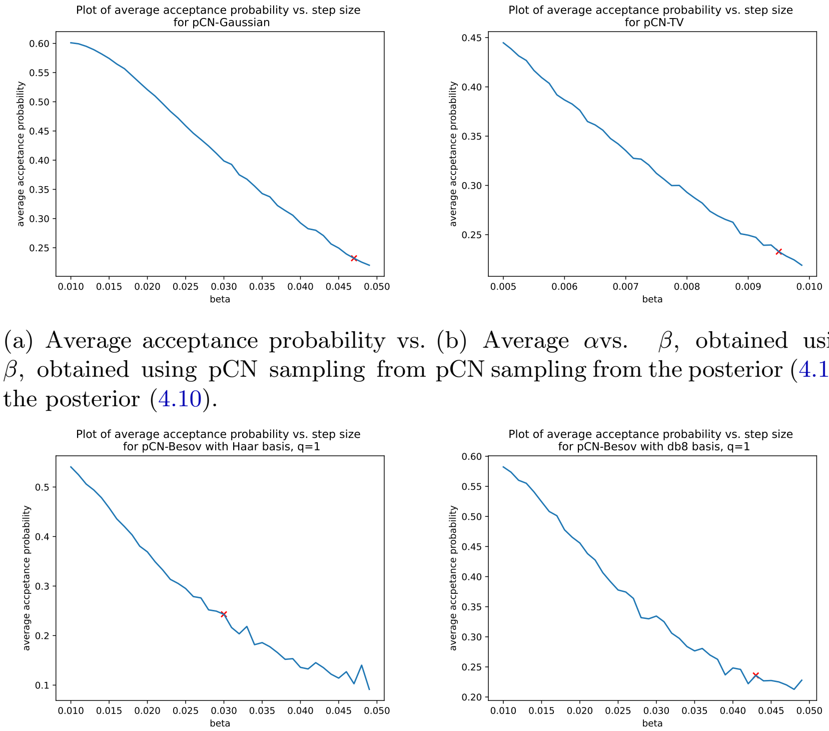
Figure 5.4: Heatmaps of $E_{L2}(\bar{\mathbf{u}}_{est})$ with varying prior parameters and β, obtained using pCN.

Parameter	Parameter value	β	Average α	$E_{L2}(\bar{\mathbf{u}}_{est})$
σ_u	0.01	0.05	0.21109	0.20895
λ	16	0.01	0.21714	0.19853
κ, Haar	1.3	0.05	0.10291	0.24154
κ, db8	1.5	0.05	0.23278	0.21248

Table 5.3: Chosen prior parameter values for pCN.

Step size parameters

We continue tuning the step size parameter β. Following [28], we choose β such that the average acceptance probability is ≈ 0.234. We fix the prior parameter values in Table 5.3 and refine our grid search to tune the step size parameter β. For each grid search, we run the pCN algorithm for 10^4 iterations and compute the average value of α over the iterations. The value of β minimising $|\alpha_{avg} - 0.234|$ is then chosen. In Table 5.1, the average α values ranged from 0.10291 to 0.23278. The finer grid search narrows down the average α values range to $0.23412 - 0.24154$.



(a) Average acceptance probability vs. β, obtained using pCN sampling from the posterior (4.10).

(b) Average αvs. β, obtained using pCN sampling from the posterior (4.11).

(c) Average α vs. β, obtained using pCN sampling from the posterior (4.12) with Haar wavelets and $q = 1$.

(d) Average α vs. β, obtained using pCN sampling from the posterior (4.12) with Daubechies-8 wavelets and $q = 1$

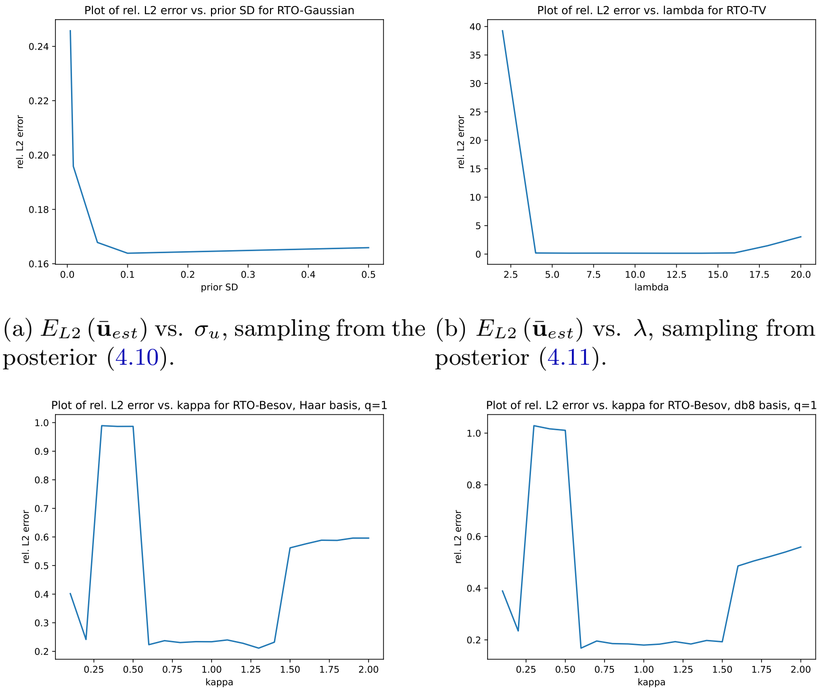
Figure 5.5: Average acceptance probabilities vs. β.

Parameter	Parameter value	β	Average α	$E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$
σ_u	0.01	0.04700	0.23412	0.21148
λ	16	0.00950	0.23276	0.20395
κ, Haar	1.3	0.03000	0.24154	0.27343
κ, db8	1.5	0.04300	0.23579	0.20903

Table 5.4: Values of β chosen for pCN.

5.2.3 Parameter choices for Randomise-Then-Optimise

To choose the standard deviation σ_u for the Gaussian smoothness prior (4.2), the total variation prior parameter λ (4.3), and the Besov prior parameter κ (4.6) for RTO, we perform grid searches which minimise $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ after 100 iterations. The step size parameter β is not present in the randomise-then-optimise sampling method. The magnitude of the stochastic perturbation in the optimisation problem (3.18) depends on $\bar{\mathbf{Q}}$, which is determined by the function $\tilde{\mathcal{F}}$. The function $\tilde{\mathcal{F}}$ is, in turn, parametrised by the prior parameters σ_u, λ, and κ. We show the plots obtained from our grid search in Figure 5.6 and summarise the chosen prior parameter values in Table 5.5.



(a) $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ vs. σ_u, sampling from the posterior (4.10).

(b) $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ vs. λ, sampling from the posterior (4.11).

(c) $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ vs. κ, sampling from the posterior (4.12) with Haar wavelets and $q = 1$.

(d) $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ vs. κ, sampling from the posterior (4.12) with Daubechies-8 wavelets and $q = 1$

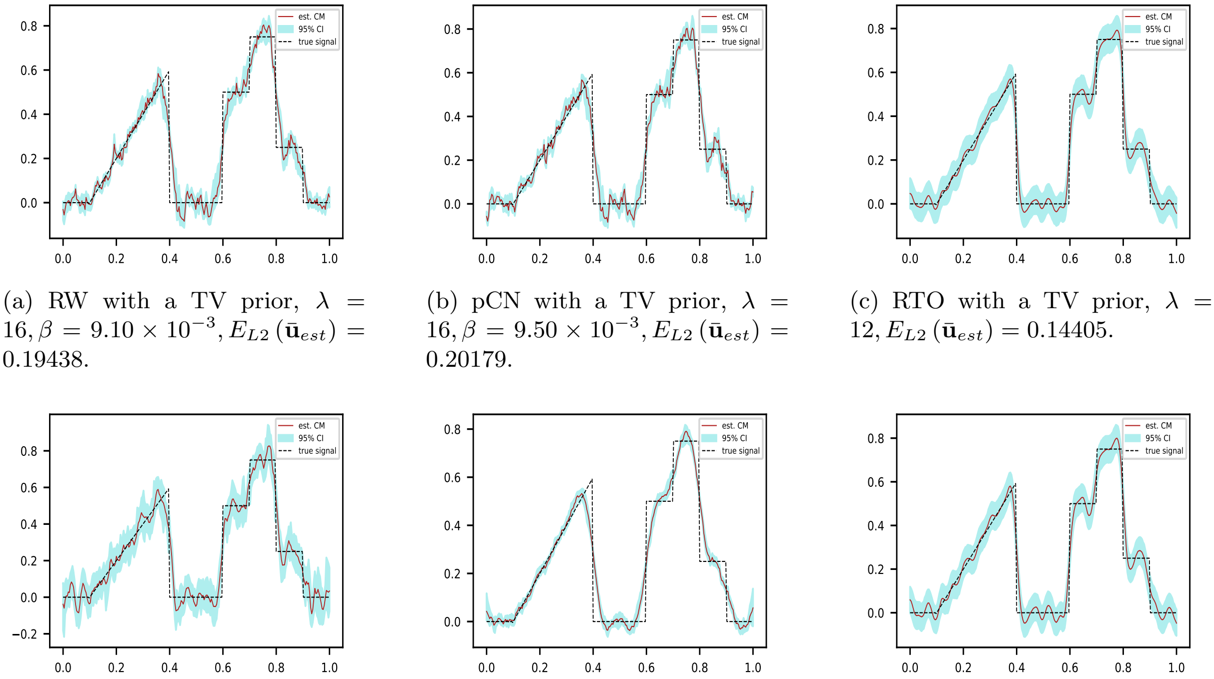
Figure 5.6: Relative L2 error $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ vs. various prior parameters obtained using RTO sampling.

Parameter	Parameter value	$E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$
σ_u	0.1	0.16386
λ	12	0.14449
κ, Haar	1.3	0.21121
κ, db8	0.6	0.16796

Table 5.5: Chosen prior parameter values for RTO.

5.3 Deconvolution with the total variation prior

The estimated conditional means and credible intervals obtained by sampling from the posterior distribution with the TV density (4.11) using RW, pCN, and RTO are presented in Figures 5.7a, 5.7b, and 5.7c. Estimated conditional means and credible intervals obtained by sampling from the posterior distribution with the density (4.10) using RW, pCN, and RTO are shown in Figures 5.7d, 5.7e, and 5.7f for comparison purposes.



(a) RW with a TV prior, $\lambda = 16, \beta = 9.10 \times 10^{-3}, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.19438$.

(b) pCN with a TV prior, $\lambda = 16, \beta = 9.50 \times 10^{-3}, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.20179$.

(c) RTO with a TV prior, $\lambda = 12, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.14405$.

(d) RW with a Gaussian smoothness prior, $\sigma_u = 0.05, \beta = 5.5 \times 10^{-3}, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.18940$.

(e) pCN with a Gaussian smoothness prior, $\sigma_u = 0.01, \beta = 4.7 \times 10^{-2}, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.20854$.

(f) RTO with Gaussian smoothness prior, $\sigma_u = 0.1, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.15660$.

Figure 5.7: Estimated posterior means computed from samples drawn using pCN and RTO with a TV prior, plotted together with the true signal and 95% estimated credible intervals.

The estimated conditional means obtained using RTO with the Gaussian and TV priors (Figures 5.7c and 5.7f, respectively) capture the features of the true signal more successfully than those in Figures 5.7a, 5.7b, 5.7d, and 5.7e.

Figure 5.7 indicates that the choices of β and σ_u for the TV and Gaussian priors in Section 5.2 are not optimal. The estimated credible intervals in Figures 5.7f and 5.7c contain the true signal, which cannot be said for the credible intervals in Figures 5.7a, 5.7b, 5.7d, and 5.7e. As β is small, the RW (3.7) and pCN (3.12) proposals are close to each other, resulting in smaller estimated credible intervals. The chosen prior standard deviation σ_u for pCN is also the smallest, affecting the size of the credible interval. The estimated conditional mean obtained using pCN with a Gaussian prior has a larger relative L2 error than the observation $\mathbf{y}$. The relative L2 error of the observation $\mathbf{y}$ is $E_{L2}(\mathbf{y}) = 0.20245$, whereas the relative L2 error of the estimated conditional mean obtained using pCN with a Gaussian prior is 0.20854.

The MAP estimator $\mathbf{u}_{MAP}$ in Figure 5.8 is the linearisation point (3.19) for the RTO algorithm. In comparison to the TV $\bar{\mathbf{u}}_{est}$ in Figure 5.7c from RTO, the MAP estimator $\mathbf{u}_{MAP}$ of the posterior density with the total variation prior and $\lambda = 12$ (shown in Figure 5.7f) is more successful at recovering the discontinuities of the true. The estimated conditional mean $\bar{\mathbf{u}}_{est}$ captures the true signal more accurately at $x = 0.4$, where the linear portion of the true signal reaches its peak. The RTO-TV $\bar{\mathbf{u}}_{est}$ is the mean of 10^4 independent samples, which may lead it to appear similar to the Gaussian $\bar{\mathbf{u}}_{est}$ in Figure 5.7f. Comparisons of the true conditional mean $\mathbf{u}_{CM}$ (2.35), MAP estimators, and estimated conditional means for the Gaussian and total variation priors with the parameters in Table 5.6 are found in Sections C.1 and C.2 of Appendix C.

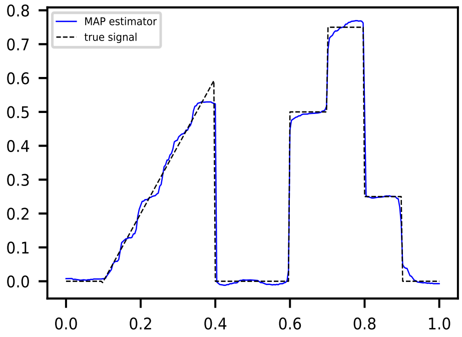
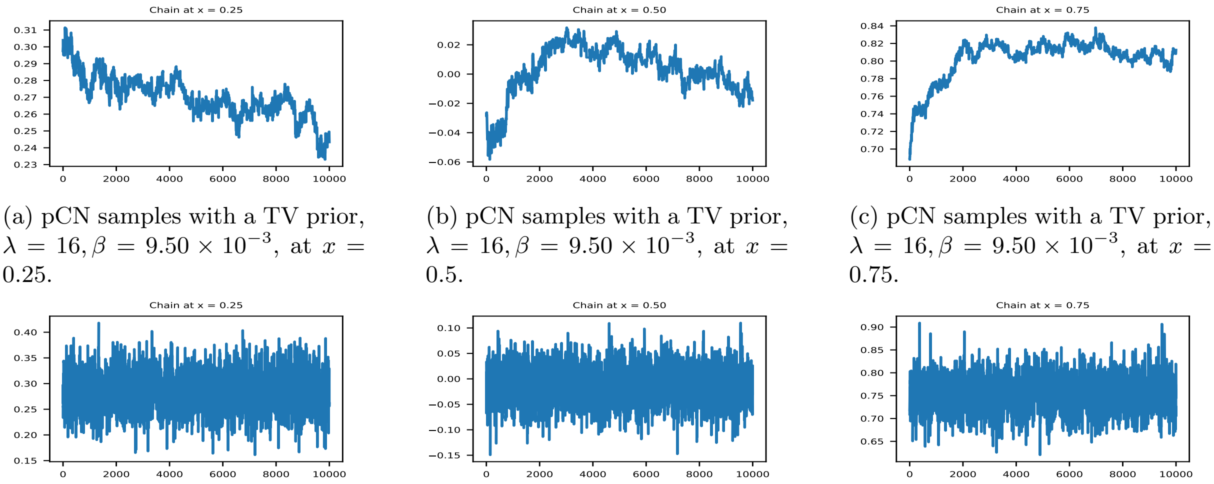


Figure 5.8: MAP estimator $\mathbf{u}_{MAP}$ of the TV posterior density (4.11) with $\lambda = 12$, corresponding to the estimated 95% CI and $\bar{\mathbf{u}}_{est}$ in Figure 5.7c.

The plots of pCN and RTO samples at selected locations, shown in Figure 5.9, indicate that values of the accepted samples obtained using pCN are much closer to each other than those obtained using RTO. This is expected, as the pCN proposal definition ensures that each proposal is correlated to the previous draw. This behavior was also seen in [28], samples drawn using pCN for a geophysics problem remain correlated after 10^6 iterations. The estimated conditional mean in Figure 5.7b is computed from correlated samples, leading to a less smooth estimate of the conditional mean compared to the estimate obtained using RTO (Figure 5.7c).



(a) pCN samples with a TV prior, $\lambda = 16, \beta = 9.50 \times 10^{-3}$, at $x = 0.25$.

(b) pCN samples with a TV prior, $\lambda = 16, \beta = 9.50 \times 10^{-3}$, at $x = 0.5$.

(c) pCN samples with a TV prior, $\lambda = 16, \beta = 9.50 \times 10^{-3}$, at $x = 0.75$.

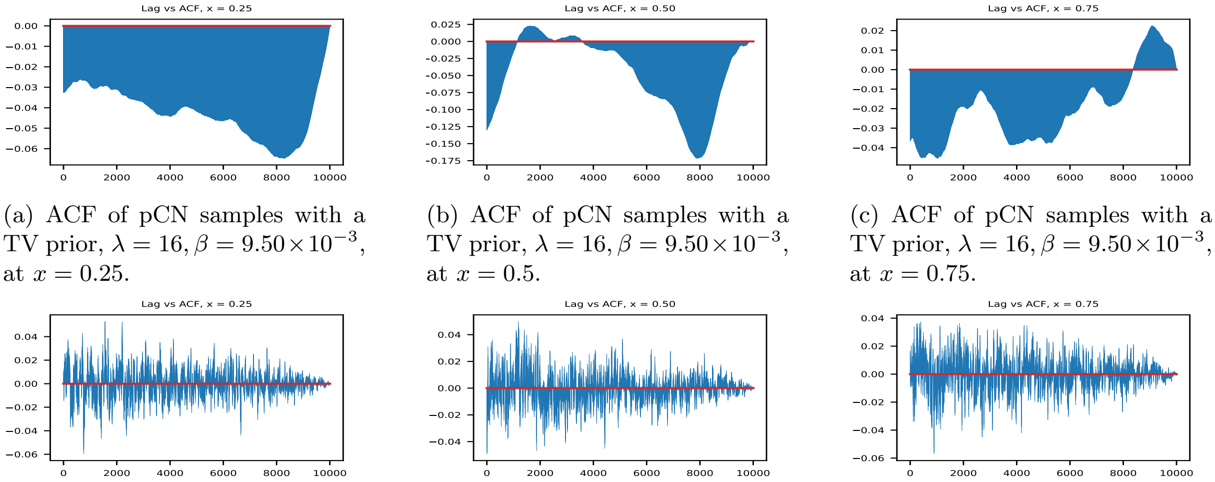
(d) RTO samples with a TV prior and $\lambda = 12$ at $x = 0.25$.

(e) RTO samples with a TV prior and $\lambda = 12$ at $x = 0.5$.

(f) RTO samples with a TV prior and $\lambda = 12$ at $x = 0.75$.

Figure 5.9: Plots of samples at the locations marked in Figure 5.1, obtained by sampling with pCN and RTO using a TV prior.

The behavior seen in the sample plots in Figure 5.9 correspond to the behavior in the ACF plots in Figure 5.10. The ACF plots of the RTO samples in Figures 5.10d, 5.10e, and 5.10f show that the trend of the ACF for RTO tends towards zero, while the ACF plots for the pCN samples in Figures 5.10a, 5.10b, and 5.10c show large autocorrelation at short lags. From Figures 5.10a, 5.10b, and 5.10c, it would appear that more than 10^4 iterations would be needed for pCN to sample for from the posterior density.



(a) ACF of pCN samples with a TV prior, $\lambda = 16, \beta = 9.50 \times 10^{-3}$, at $x = 0.25$.

(b) ACF of pCN samples with a TV prior, $\lambda = 16, \beta = 9.50 \times 10^{-3}$, at $x = 0.5$.

(c) ACF of pCN samples with a TV prior, $\lambda = 16, \beta = 9.50 \times 10^{-3}$, at $x = 0.75$.

(d) ACF of RTO samples with a TV prior and $\lambda = 12$ at $x = 0.25$.

(e) ACF of RTO samples with a TV prior and $\lambda = 12$ at $x = 0.5$.

(f) ACF of RTO samples with a TV prior and $\lambda = 12$ at $x = 0.75$.

Figure 5.10: ACF of samples at the locations marked in Figure 5.1, obtained by sampling with pCN and RTO using TV priors.

Prior	**Method**	**Prior parameter**	**Step size**	**Accepted samples** (%)	$E_{L2}(\bar{\mathbf{u}}_{MAP})$	$E_{L2}(\bar{\mathbf{u}}_{est})$
Gaussian	RW	0.05	0.0055	24.08	0.16369	0.18940
Gaussian	pCN	0.01	0.047	22.256	0.19539	0.20854
Gaussian	RTO	0.1	N/A	99.976	0.15713	0.15660
TV	RW	16	0.0091	22.552	0.11445	0.19438
TV	pCN	16	0.0095	21.256	0.11445	0.20179
TV	RTO	12	N/A	95.152	0.11476	0.14405

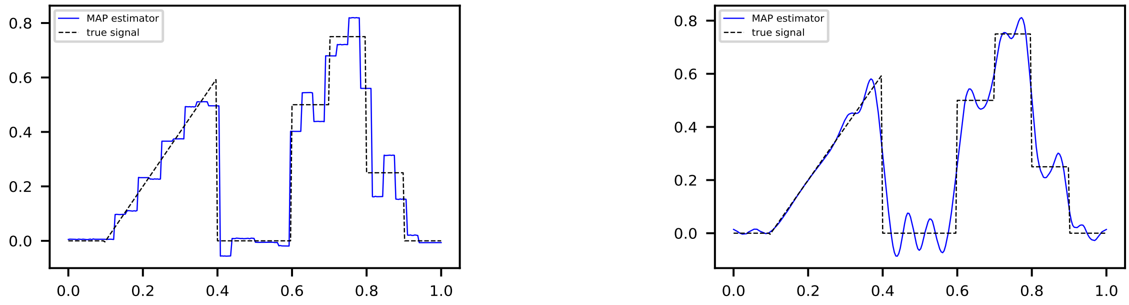
Table 5.6: Relative $L2$ errors of estimated conditional means obtained using the three sampling methods with Gaussian and TV priors. The prior parameters are the standard deviation of the Gaussian prior, σ_u, and the TV parameter, λ. The relative error of the observation $\mathbf{y}$ is $E_{L2}(\mathbf{y}) = 0.20245$.

The relative L2 errors of the estimated conditional means, $E_{L2}(\bar{\mathbf{u}}_{est})$, percentages of accepted samples, and L2 errors of the linearisation point, $E_{L2}(\bar{\mathbf{u}}_{MAP})$ are shown in Table 5.6. The estimated conditional means computed using samples generated with RTO have the smallest relative L2 errors, with $E_{L2}(\bar{\mathbf{u}}_{est}) = 0.15660$ for RTO with a Gaussian prior and $E_{L2}(\bar{\mathbf{u}}_{est}) = 0.14405$ for RTO with a TV prior. The estimators with the smallest relative L2 errors are the MAP estimators for the TV posterior density (4.11), which are shown in Figures C.4b, C.5b, and C.6b of Appendix C. As with the MAP estimator for the TV posterior density with $\lambda = 12$ shown in Figure 5.8, the MAP estimator for $\lambda = 16$ recovers the discontinuities of the true signal. The estimated conditional means $\bar{\mathbf{u}}_{est}$ are more successful at recovering the peak of the linear portion of the true signal at $x = 0.4$, as seen in Figures 5.7c and 5.7f.

The percentages of accepted samples for RW and pCN are close to the average α value used in the parameter tuning process in Sections 5.2.1-5.2.3. Meanwhile, the RTO acceptance rates are $> 90\%$. When a Gaussian prior is used, the RTO sampler is an efficient method to directly sample from the posterior density [29]. When a TV prior is used, all the assumptions in Theorem 3.5.1 are fulfilled, since the transformation T_λ (4.4) is invertible and continuously differentiable (see Appendix B.2). RTO then generates proposals around the linearisation point, in this case the MAP estimator $\bar{\mathbf{u}}_{est}$. As such, the proposals are drawn from regions of high probability density.

5.4 Deconvolution with Besov priors

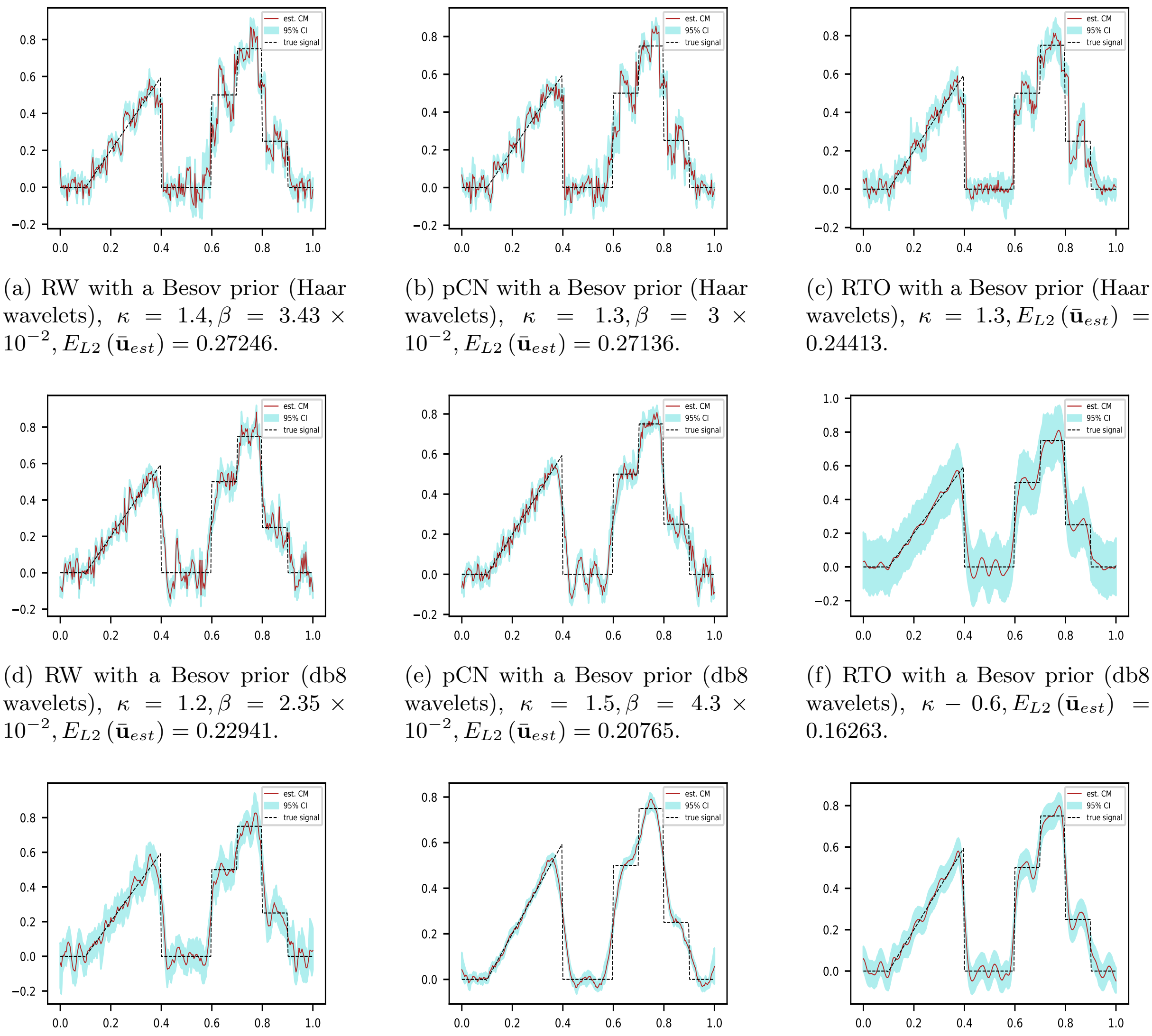
The estimated conditional means and credible intervals obtained by sampling from the posterior distribution with the posterior density obtained with a discretised Besov prior (4.12) using RW, pCN, and RTO are presented in Figure 5.12. The Besov prior parameter κ is obtained in Section 5.2. To construct Besov priors that behave similarly to TV priors, the parameters s, q from (2.45) are set as $s = 1$ and $q = 1$ to obtain the results in Figure 5.12. The estimated conditional means computed using RW and pCN samples do not de-noise the data effectively, as seen in Figures 5.12a, 5.12b, 5.12d, and 5.12a. Figures 5.12g, 5.12h, and 5.12i are the same plots as shown in Figures 5.7d, 5.7e, and 5.7f, repeated for comparison. The linearisation points for RTO, which are the MAP estimators of the posterior density (4.12) obtained using Besov priors, are plotted in Figures 5.11. Comparisons to the estimated conditional mean obtained from RW, pCN, and RTO are shown in Sections C.3 and C.4 of Appendix C.



(a) MAP estimator $\mathbf{u}_{MAP}$ of the Besov posterior density (4.12) (Haar wavelets) with $s = 1, q = 1, \kappa = 1.3$, corresponding to the estimated 95% CI and $\bar{\mathbf{u}}_{est}$ in Figure 5.12c.

(b) MAP estimator $\mathbf{u}_{MAP}$ of the Besov posterior density (4.12) (db8 wavelets) with $s = 1, q = 1, \kappa = 0.6$, corresponding to the estimated 95% CI and $\bar{\mathbf{u}}_{est}$ in Figure 5.12f

Figure 5.11: MAP estimators $\mathbf{u}_{MAP}$ of the Besov posterior densities sampled from using RTO.



(a) RW with a Besov prior (Haar wavelets), $\kappa = 1.4, \beta = 3.43 \times 10^{-2}, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.27246$.

(b) pCN with a Besov prior (Haar wavelets), $\kappa = 1.3, \beta = 3 \times 10^{-2}, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.27136$.

(c) RTO with a Besov prior (Haar wavelets), $\kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.24413$.

(d) RW with a Besov prior (db8 wavelets), $\kappa = 1.2, \beta = 2.35 \times 10^{-2}, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.22941$.

(e) pCN with a Besov prior (db8 wavelets), $\kappa = 1.5, \beta = 4.3 \times 10^{-2}, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.20765$.

(f) RTO with a Besov prior (db8 wavelets), $\kappa - 0.6, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.16263$.

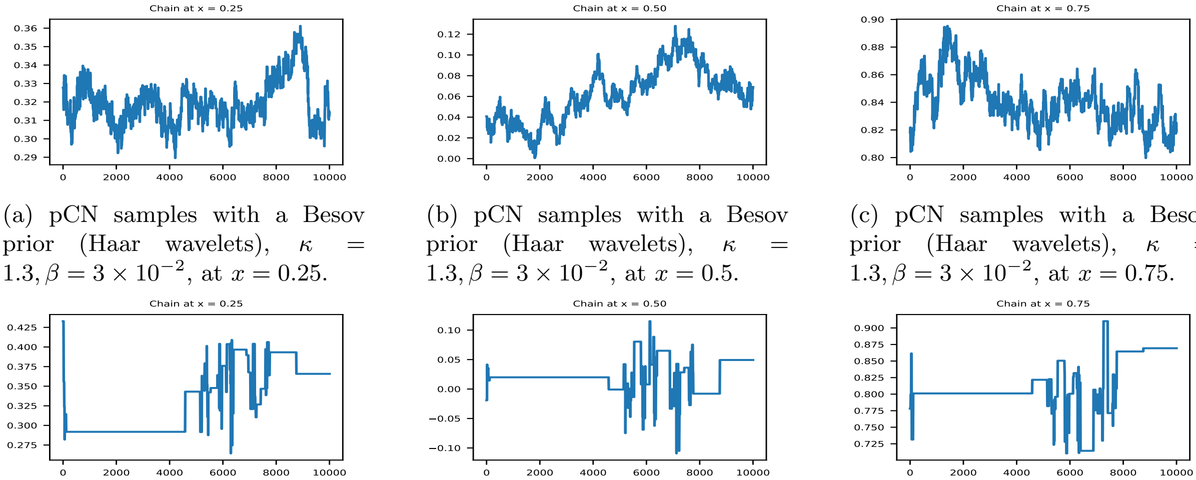
(g) RW with a Gaussian smoothness prior, $\sigma_u = 0.05, \beta = 5.5 \times 10^{-3}, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.18940$.

(h) pCN with a Gaussian smoothness prior, $\sigma_u = 0.01, \beta = 4.7 \times 10^{-2}, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.20854$.

(i) RTO with Gaussian smoothness prior, $\sigma_u = 0.1, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.15660$.

Figure 5.12: Estimated posterior means computed from samples drawn using pCN and RTO with Besov priors, plotted together with the true signal and 95% estimated credible intervals.

Figures 5.12c and 5.12f show that the choice of wavelet basis functions has a significant effect on the estimated conditional mean. The estimated conditional mean obtained using RTO with a Haar basis is considerably less smooth than the estimated conditional mean obtained using RTO with a db8 basis. This was also found to be the case in [41], where Besov priors were constructed using the Haar and db8 wavelets to solve the inpainting problem. Compared to the estimated conditional means obtained using the Gaussian smoothness prior (4.2) in Figures 5.12d, 5.12e, 5.12f, the estimated conditional mean obtained using the Besov prior constructed with Haar wavelets, shown in Figure 5.12c, is more successful at capturing sudden jumps. The estimated conditional mean obtained using the Besov prior constructed with db8 wavelets is more successful at capturing the smoother pieces of the function (Figure 5.12f).



(a) pCN samples with a Besov prior (Haar wavelets), $\kappa = 1.3, \beta = 3 \times 10^{-2}$, at $x = 0.25$.

(b) pCN samples with a Besov prior (Haar wavelets), $\kappa = 1.3, \beta = 3 \times 10^{-2}$, at $x = 0.5$.

(c) pCN samples with a Besov prior (Haar wavelets), $\kappa = 1.3, \beta = 3 \times 10^{-2}$, at $x = 0.75$.

(d) RTO samples with a Besov prior (Haar wavelets), $\kappa = 1.3$, at $x = 0.25$.

(e) RTO samples with a Besov prior (Haar wavelets), $\kappa = 1.3$, at $x = 0.5$.

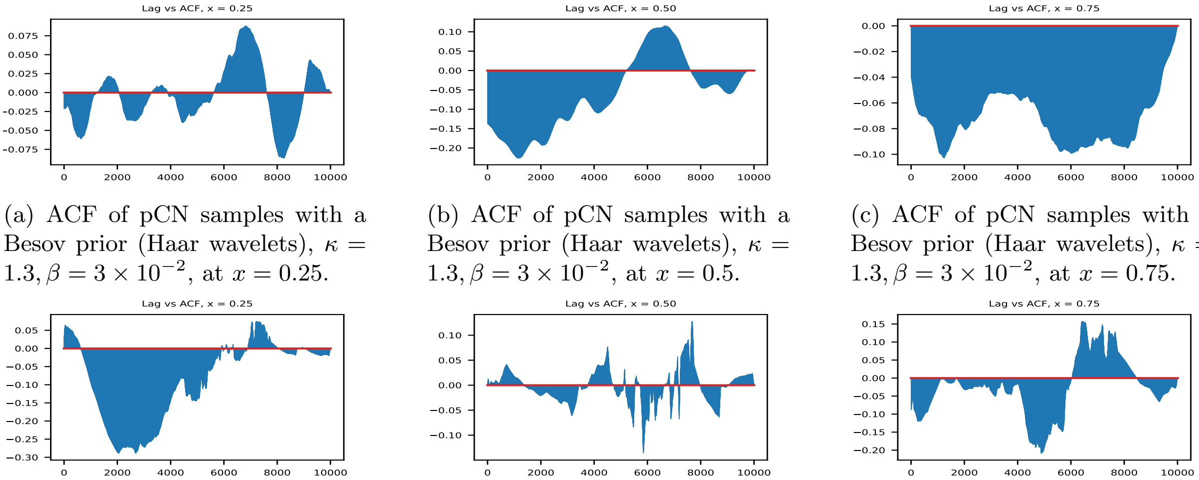
(f) RTO samples with a Besov prior (Haar wavelets), $\kappa = 1.3$, at $x = 0.75$.

Figure 5.13: Plots of samples at the locations marked in Figure 5.1, obtained by sampling with pCN and RTO using Besov priors (Haar wavelets).

When Haar wavelets are used to construct a Besov prior for the deconvolution problem, 0.880% of samples were accepted by RTO. Figures 5.13d, 5.13e, and 5.13f show that the samples remained constant for at least 4000 iterations. A local minimum may have been found when the optimisation problem (3.18) was solved in one of the iterations. In this implementation of RTO, the low percentage of accepted samples cannot be addressed by tuning a parameter that has a similar role to the step size β in RW and pCN. Whether or not the RTO sampling method can explore the posterior distribution effectively is therefore determined by the choice of prior. In contrast, the pCN sample plots in Figures 5.13a, 5.13b, and 5.13c show that pCN continues to accept samples, with 20.528% of samples being accepted. As in Figure 5.9, the samples appear correlated.

The ACF plots in Figures 5.14a, 5.14b, 5.14c, 5.15a, 5.15b, 5.15c show that there are autocorrelations at short lags for pCN samples when Haar and db8 wavelets are used. Together with the estimated conditional means shown in Figures 5.12b and 5.12e, this indicates the choices for the step size parameter β in Subsection 5.2.2 are not optimal. Additionally, the algorithm may not have been run for enough iterations.

The ACF plots in Figures 5.14d, 5.14e, 5.14f also show autocorrelations at short lags, which do not tend to approach zero. In contrast, the ACF plots in Figures 5.15d, 5.15e, 5.15f show steadily decreasing autocorrelations. Together with the sample plots in Figures 5.16d, 5.16e, 5.16f, which show samples that appear independent, this indicates that RTO samples more effectively from a posterior distribution obtained with a Besov (db8) prior than a posterior distribution obtained with a Besov (Haar) prior.



(a) ACF of pCN samples with a Besov prior (Haar wavelets), $\kappa = 1.3, \beta = 3 \times 10^{-2}$, at $x = 0.25$.

(b) ACF of pCN samples with a Besov prior (Haar wavelets), $\kappa = 1.3, \beta = 3 \times 10^{-2}$, at $x = 0.5$.

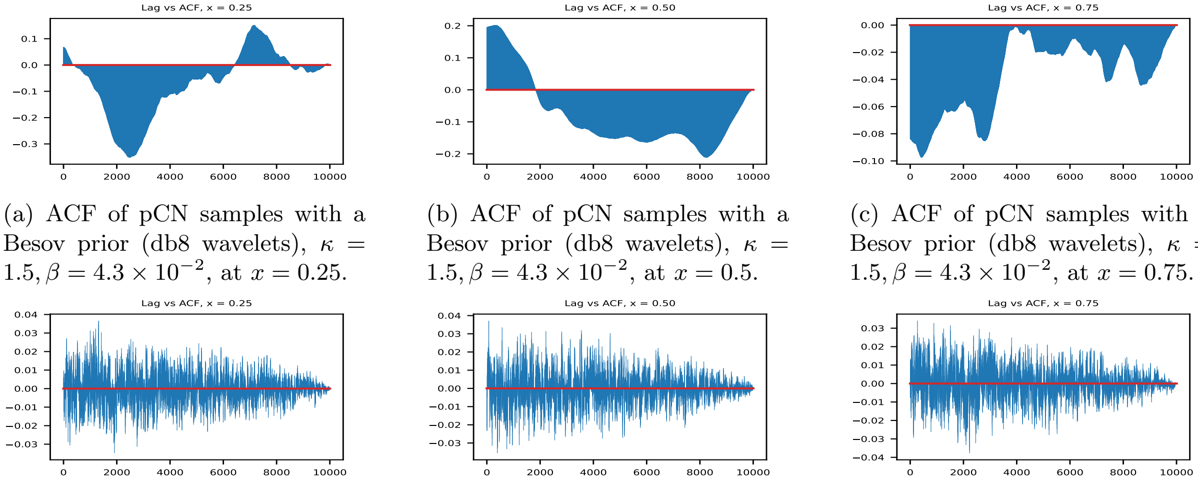
(c) ACF of pCN samples with a Besov prior (Haar wavelets), $\kappa = 1.3, \beta = 3 \times 10^{-2}$, at $x = 0.75$.

(d) ACF of RTO samples with a Besov prior (Haar wavelets), $\kappa = 1.3$, at $x = 0.25$.

(e) ACF of RTO samples with a Besov prior (Haar wavelets), $\kappa = 1.3$, at $x = 0.5$.

(f) ACF of RTO samples with a Besov prior (Haar wavelets), $\kappa = 1.3$, at $x = 0.75$.

Figure 5.14: ACF of samples at the locations marked in Figure 5.1, obtained by sampling with pCN and RTO using Besov priors (Haar wavelets).



(a) ACF of pCN samples with a Besov prior (db8 wavelets), $\kappa = 1.5, \beta = 4.3 \times 10^{-2}$, at $x = 0.25$.

(b) ACF of pCN samples with a Besov prior (db8 wavelets), $\kappa = 1.5, \beta = 4.3 \times 10^{-2}$, at $x = 0.5$.

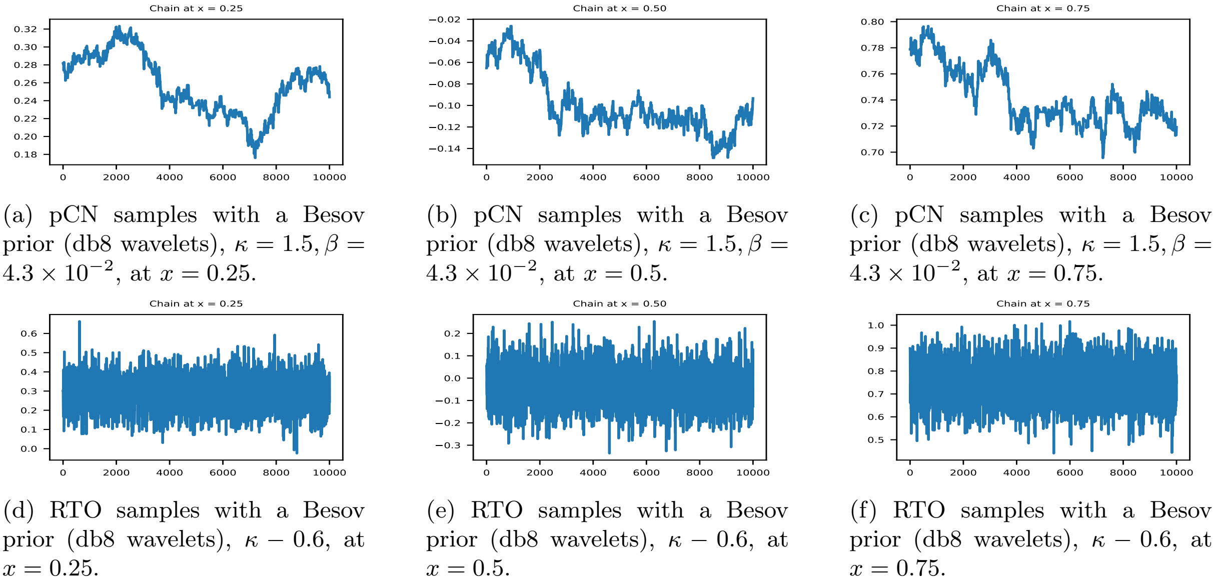
(c) ACF of pCN samples with a Besov prior (db8 wavelets), $\kappa = 1.5, \beta = 4.3 \times 10^{-2}$, at $x = 0.75$.

(d) ACF of RTO samples with a Besov prior (db8 wavelets), $\kappa - 0.6$, at $x = 0.25$.

(e) ACF of RTO samples with a Besov prior (db8 wavelets), $\kappa - 0.6$, at $x = 0.5$.

(f) ACF of RTO with a Besov prior (db8 wavelets), $\kappa - 0.6$, at $x = 0.75$.

Figure 5.15: ACF of samples at the locations marked in Figure 5.1, obtained by sampling with pCN and RTO using Besov priors (db8 wavelets).



(a) pCN samples with a Besov prior (db8 wavelets), $\kappa = 1.5, \beta = 4.3 \times 10^{-2}$, at $x = 0.25$.

(b) pCN samples with a Besov prior (db8 wavelets), $\kappa = 1.5, \beta = 4.3 \times 10^{-2}$, at $x = 0.5$.

(c) pCN samples with a Besov prior (db8 wavelets), $\kappa = 1.5, \beta = 4.3 \times 10^{-2}$, at $x = 0.75$.

(d) RTO samples with a Besov prior (db8 wavelets), $\kappa - 0.6$, at $x = 0.25$.

(e) RTO samples with a Besov prior (db8 wavelets), $\kappa - 0.6$, at $x = 0.5$.

(f) RTO samples with a Besov prior (db8 wavelets), $\kappa - 0.6$, at $x = 0.75$.

Figure 5.16: Plots of samples at the locations marked in Figure 5.1, obtained by sampling with pCN and RTO using Besov priors (db8 wavelets).

Prior	**Method**	κ	s	q	**Step size**	**Accepted samples** (%)	$E_{L2}(\bar{\mathbf{u}}_{MAP})$	$E_{L2}(\bar{\mathbf{u}}_{est})$
Besov (Haar)	RW	1.4	1.0	1	0.0343	18.184	0.24134	0.27246
Besov (Haar)	pCN	1.3	1.0	1	0.03	20.528	0.24080	0.27136
Besov (Haar)	RTO	1.3	1.0	1	N/A	0.880	0.24080	0.24413
Besov (db8)	RW	1.2	1.0	1	0.0235	26.472	0.18105	0.22941
Besov (db8)	pCN	1.5	1.0	1	0.043	14.424	0.18429	0.20765
Besov (db8)	RTO	0.6	1.0	1	N/A	51.656	0.17597	0.16263

Table 5.7: Relative $L2$ errors of estimated conditional means obtained using the three sampling methods with Besov priors. The relative error of the observation $\mathbf{y}$ is $E_{L2}(\mathbf{y}) = 0.20245$.

The relative L2 errors of the estimated conditional means $E_{L2}(\bar{\mathbf{u}}_{est})$, percentages of accepted samples, and relative L2 errors of the linearisation point, $E_{L2}(\bar{\mathbf{u}}_{MAP})$ are shown in Table 5.7. For all three methods, the relative L2 errors of the estimated conditional mean $E_{L2}(\bar{\mathbf{u}}_{est})$ are smaller when db8 wavelets are used to construct Besov priors compared to when Haar wavelets are used. In this section, the smallest $E_{L2}(\bar{\mathbf{u}}_{est})$ is obtained when RTO is used to sample from a posterior density (4.12) with a Besov prior constructed using db8 wavelets. Note that, with the exception of the estimated conditional mean $\bar{\mathbf{u}}_{est}$ obtained using RTO with a Besov prior and db8 wavelets, all the estimated conditional means in Table 5.7 have larger relative L2 errors than the observation $\mathbf{y}$. The MAP estimators $\mathbf{u}_{MAP}$ obtained with the Besov prior and db8 wavelets have smaller relative errors than the observation $\mathbf{y}$. They are plotted in Section C.4 of Appendix C.

The percentage of accepted samples for RTO with Besov priors is 0.880% when Haar wavelets are used and it is 51.656% when db8 wavelets are used. These are much lower values compared to the percentage of accepted samples for RTO with a Gaussian smoothness prior (99.976%) and RTO with a TV prior (95.152%). Note that, while the TV prior transformation T_λ is continuously differentiable and invertible, the Besov prior transformation $T_{\tau,q}$ is only invertible, as the derivative of $g_{\tau,q}$ is not continuous at zero (see Appendix (B.3)). Therefore, Assumption (2) of Theorem 3.5.1 is not fulfilled on the entirety of $\mathbb{R}^n$. In practice, the Jacobian of $\tilde{\mathcal{F}}$ is approximated using the Python library `JAX`, which handles discontinuous gradients by perturbing the function $\tilde{\mathcal{F}}$ slightly near the discontinuity and taking gradient values close to the discontinuity. Then, the optimisation problem (3.18) can still be solved to generate RTO proposals, although these samples may not be in high-density areas, leading to a smaller percentage being accepted.

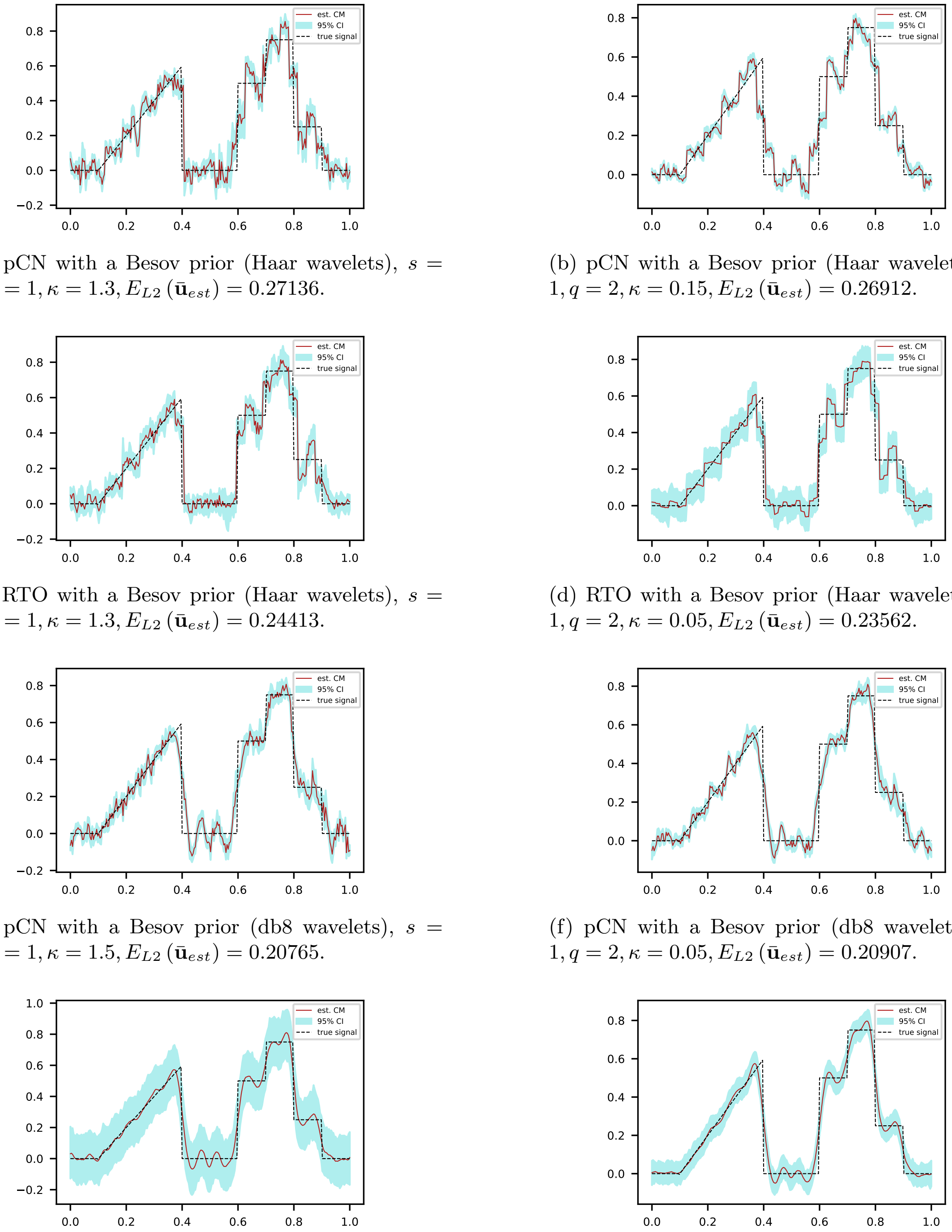
5.4.1 Influence of s and q parameters

The s parameter of a Besov random function (2.45) affects the regularity properties of the function by determining the decay of the deterministic coefficients. The parameter q characterises the generalised Gaussian distribution [51] in (4.7) and the Besov space $B^s_{qq}(\mathbb{T}^d)$. In Section 5.4, the values of these parameters are set as $s = 1$ and $q = 1$ to construct Besov priors that behave similarly to total variation priors. In this subsection, the effects of changing the s and q on parameters are investigated. The estimated conditional means obtained by varying q and s are plotted. Plots of MAP estimators can be found in Appendix C.

The q parameter

When $q = 1$, the generalised Gaussian density function (4.7) coincides with a Laplace density, and when $q = 2$ the generalised Gaussian density function (4.7) coincides with a Gaussian density function [51]. In Besov spaces $B^s_{qq}(\mathbb{T}^d)$, q is connected to the $L^q(\mathbb{T}^d)$ spaces where the elements of $B^s_{qq}(\mathbb{T}^d)$ reside (see Definition 2.3.12). For the figures below, the parameter κ is set by minimising the relative L2 error of the conditional mean $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ using a grid search, as in Section 5.2.

Estimated conditional means obtained by sampling with a Besov prior and Haar wavelets with $q = 2, s = 1$ are shown in Figures 5.17b and 5.17d. Estimated conditional means obtained by sampling with a Besov prior constructed using db8 wavelets with $q = 2, s = 1$ are shown in Figures 5.17f and 5.17h. Comparing Figures 5.17b, 5.17d, 5.17f, and 5.17h to Figures 5.18b, 5.18e, 5.19b, and 5.19e shows that sampling with the parameter $q = 2$ and corresponding κ values has similar effects as increasing s, although s and q parameters play different roles in the construction of the Besov priors. In [41], where the effects of q were studied for the inpainting problem, it was also found that increasing the s and q parameters do not lead to significantly different effects.



(a) pCN with a Besov prior (Haar wavelets), $s = 1, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.27136$.

(b) pCN with a Besov prior (Haar wavelets), $s = 1, q = 2, \kappa = 0.15, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.26912$.

(c) RTO with a Besov prior (Haar wavelets), $s = 1, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.24413$.

(d) RTO with a Besov prior (Haar wavelets), $s = 1, q = 2, \kappa = 0.05, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.23562$.

(e) pCN with a Besov prior (db8 wavelets), $s = 1, q = 1, \kappa = 1.5, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.20765$.

(f) pCN with a Besov prior (db8 wavelets), $s = 1, q = 2, \kappa = 0.05, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.20907$.

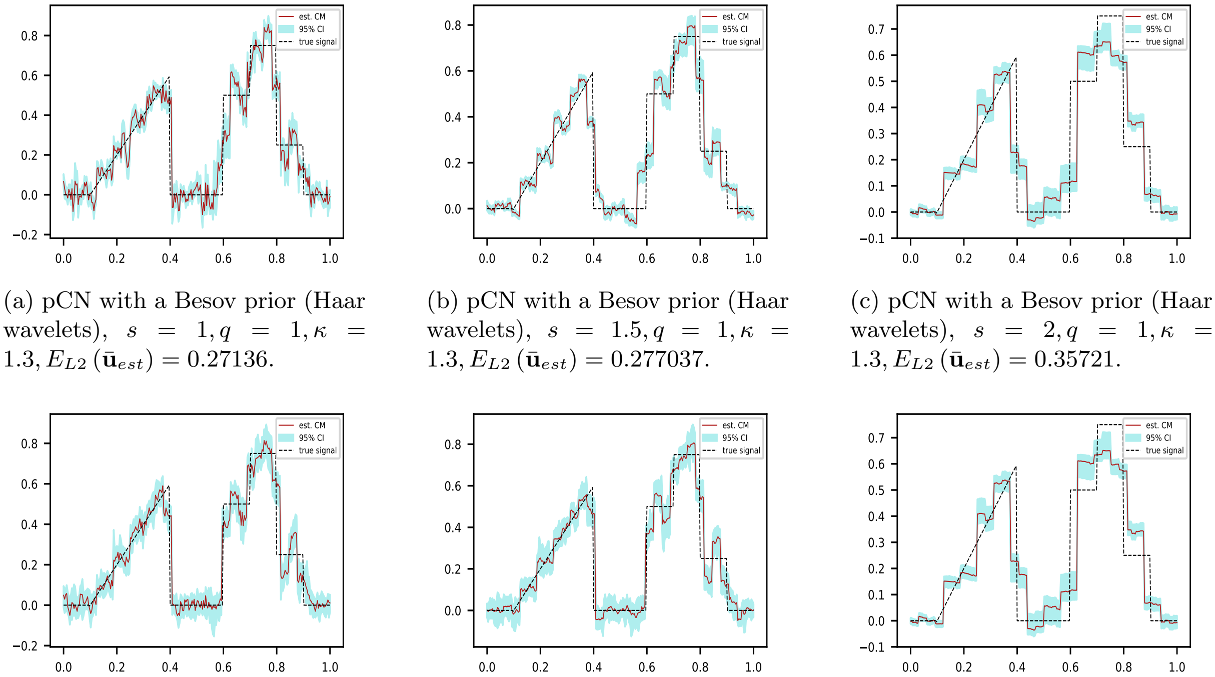
(g) RTO with a Besov prior (db8 wavelets), $s = 1, q = 1, \kappa = 0.6, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.16263$.

(h) RTO with a Besov prior (db8 wavelets), $s = 1, q = 2, \kappa = 0.1, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.16838$.

Figure 5.17: Deconvolution results obtained using pCN and RTO with Besov priors, $s = 1$ and different q values.

The s parameter

In Figure 5.18, estimated conditional means obtained by sampling with a Besov prior and Haar wavelets with $s \in \{1, 1.5, 2\}$ and $q = 1$ are shown.



(a) pCN with a Besov prior (Haar wavelets), $s = 1, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.27136$.

(b) pCN with a Besov prior (Haar wavelets), $s = 1.5, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.277037$.

(c) pCN with a Besov prior (Haar wavelets), $s = 2, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.35721$.

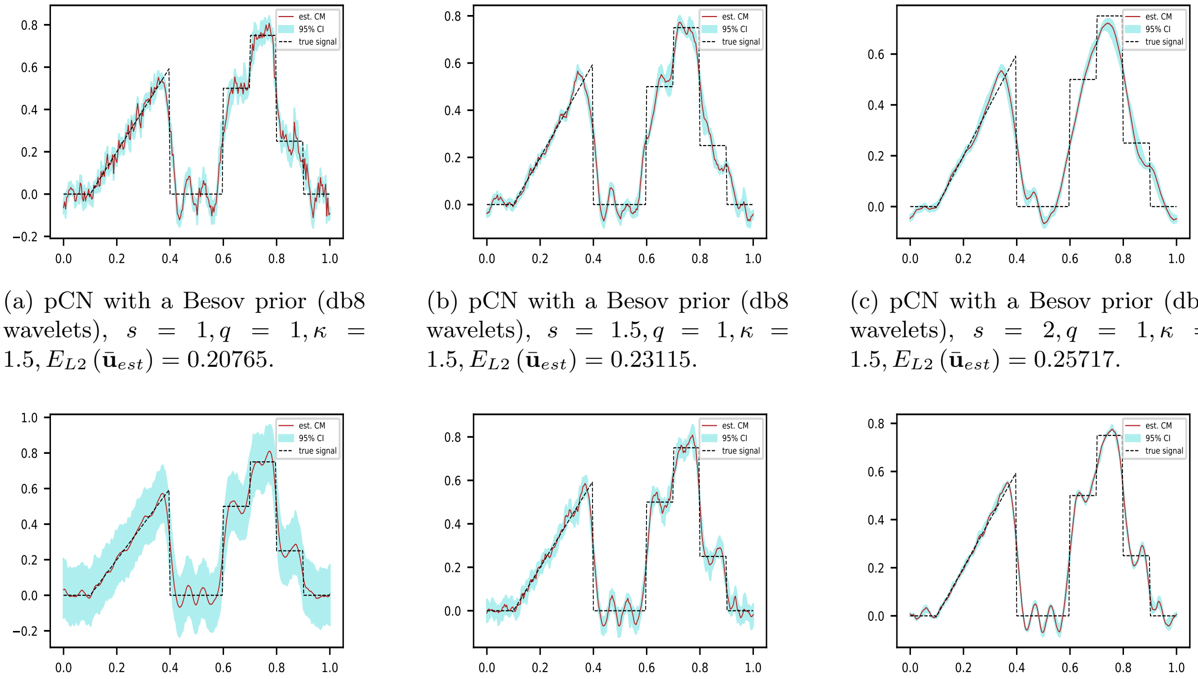
(d) RTO with a Besov prior (Haar wavelets), $s = 1, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.24413$.

(e) RTO with a Besov prior (Haar wavelets), $s = 1.5, q = 1, \kappa = 0.5, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.23629$.

(f) RTO with a Besov prior (Haar wavelets), $s = 2, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.41255$.

Figure 5.18: Deconvolution results obtained using Besov priors (Haar wavelets) with $q = 1$ and varying s.

In Figure 5.19, estimated conditional means obtained by sampling with a Besov prior and db8 wavelets with $s \in \{1, 1.5, 2\}$ and $q = 1$ are shown. Increasing s results in faster decay of deterministic coefficients in (2.45), similar to the effect observed in [41] when s was varied for the inpainting problem. When the Haar wavelets are used, the estimated conditional mean has a more step function-like appearance, as seen in Figures 5.18c and 5.18f. The estimated conditional means obtained using db8 wavelets and $s = 2$ are not significantly deconvolved, as seen in 5.19c and 5.19f. This is because, when the deterministic coefficients decay more rapidly, larger jumps between adjacent values are discouraged.



(a) pCN with a Besov prior (db8 wavelets), $s = 1, q = 1, \kappa = 1.5, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.20765$.

(b) pCN with a Besov prior (db8 wavelets), $s = 1.5, q = 1, \kappa = 1.5, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.23115$.

(c) pCN with a Besov prior (db8 wavelets), $s = 2, q = 1, \kappa = 1.5, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.25717$.

(d) RTO with a Besov prior (db8 wavelets), $s = 1, q = 1, \kappa = 0.6, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.16263$.

(e) RTO with a Besov prior (db8 wavelets), $s = 1.5, q = 1, \kappa = 0.6, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.16893$.

(f) RTO with a Besov prior (db8 wavelets), $s = 2, q = 1, \kappa = 0.6, E_{L2}(\bar{\mathbf{u}}_{est}) = 0.18705$.

Figure 5.19: Deconvolution results obtained using Besov priors (db8 wavelets) with $q = 1$ and varying s.

Prior	**Method**	κ	s	q	**Accepted samples** (%)	$E_{L2}(\bar{\mathbf{u}}_{MAP})$	$E_{L2}(\bar{\mathbf{u}}_{est})$
Besov (db8)	RTO	0.1	1.0	2	98.64	0.16810	0.16838
Besov (db8)	RTO	0.6	1.5	1	4.96	0.17937	0.16893
Besov (db8)	RTO	0.6	2.0	1	0.80	0.18645	0.18705

Table 5.8: Summary of errors of estimated conditional means obtained using Besov priors with varying s and q parameters with smaller $E_{L2}(\bar{\mathbf{u}}_{est})$ smaller than the relative error of the observation $\mathbf{y}, E_{L2}(\mathbf{y}) = 0.20245$.

Table 5.8 lists the parameter values for which the relative L2 errors of the estimated conditional means $E_{L2}(\bar{\mathbf{u}}_{est})$ are smaller than the relative L2 errors of the observation $E_{L2}(\mathbf{y}) = 0.20245$. The parameter values with the smallest $E_{L2}(\bar{\mathbf{u}}_{est})$ value are $s = 1.0, q = 2, \kappa = 0.1$. The value of $E_{L2}(\bar{\mathbf{u}}_{est})$ with these parameters is larger than the $E_{L2}(\bar{\mathbf{u}}_{est})$ value for RTO-Besov (db8) with $s = 1.0, q = 1, \kappa = 0.6$ and $E_{L2}(\bar{\mathbf{u}}_{est})$ value for RTO-TV with $\lambda = 12$.

5.5 Comparison of errors and computation times

The relative L2 errors of the estimated conditional means $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$, relative L2 errors of the RTO linearisation point (or posterior mode) $E_{L2}\left(\bar{\mathbf{u}}_{MAP}\right)$, and percentages of accepted samples are summarised in Table 5.9.

Prior	**Method**	**Prior parameter**	**Step size**	**Accepted samples** (%)	$E_{L2}\left(\bar{\mathbf{u}}_{MAP}\right)$	$E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$
Gaussian	RW	0.05	0.0055	24.08	0.16369	0.18940
Gaussian	pCN	0.01	0.047	22.256	0.19539	0.20854
Gaussian	RTO	0.1	N/A	99.976	0.15713	0.15660
TV	RW	16	0.0091	22.552	0.11445	0.194384
TV	pCN	16	0.0095	21.256	0.11445	0.20179
TV	RTO	12	N/A	95.152	0.11476	0.14405
Besov (Haar)	RW	1.4	0.0343	18.184	0.24134	0.27246
Besov (Haar)	pCN	1.3	0.03	20.528	0.24080	0.27136
Besov (Haar)	RTO	1.3	N/A	0.880	0.86676	0.24413
Besov (db8)	RW	1.2	0.0235	26.472	0.18105	0.22941
Besov (db8)	pCN	1.5	0.043	14.424	0.18429	0.20765
Besov (db8)	RTO	0.6	N/A	51.656	0.17597	0.16263

Table 5.9: Relative $L2$ errors of estimated conditional means obtained using the three sampling methods with Gaussian, TV, and Besov priors.

The relative L2 error of the observation $\mathbf{y}$ is $E_{L2}\left(\mathbf{y}\right) = 0.20245$. Not all the estimated conditional means result in reduced relative L2 error values. The estimated conditional means with the smallest relative L2 error values are RTO-TV, with $E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.14405$, RTO-Gaussian, with $E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.15660$, and RTO-Besov (db8) $E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.16263$. Relative to $E_{L2}\left(\mathbf{y}\right) = 0.20245$, the $E_{L2}\left(\bar{\mathbf{u}}_{est}\right)$ values for estimated conditional means computed using RTO-TV, RTO-Gaussian, and RTO-Besov (db8) are reduced by $28.85\%, 22.64\%$, and 19.67% respectively. The estimated credible intervals found using RTO-TV and RTO-Besov (db8) mostly contain the true signal.

The faster mixing and higher acceptance probabilities of RTO compared to pCN is not entirely surprising, as both the proposal (3.18) and acceptance probability (3.26) of RTO are formulated based on the posterior density, whereas for whitened pCN only the acceptance probability (3.14) takes into account the posterior density. In [55], comparisons between RTO and pCN support the idea that RTO mixes more rapidly than pCN, and pCN may fail to produce meaningful estimates of the posterior. While estimated conditional means computed from samples obtained using RTO have been more accurate, as sho wn in Tables 5.6 and 5.7, and RTO has resulted in better mixing (as shown in Figures 5.9, 5.13, and 5.16), pCN affords an additional control over the sampling process in the form of the step size parameter β, which can be tuned for a fixed prior.

Figures 5.7f, 5.7c, and 5.12f show that the estimated conditional means obtained using RTO-TV, RTO-Gaussian, and RTO-Besov (db8) recover features of the true signal successfully, although the edge-preserving priors (TV and Besov) do not capture the discontinuities in the signal. The results obtained using the Besov priors, in particular, are heavily influenced by the wavelet basis functions. The Besov (db8) estimated conditional mean computed using RTO samples is smooth, as the db8 wavelet is smooth. In contrast, the estimated conditional mean computed using the Haar basis functions was able to capture sharp jumps, but did not successfully recover the continuous linear piece of the function. Similar conclusions were drawn in [41] for one-dimensional deconvolution.

Computational efficiency is evaluated through CPU time required to generate 10^4 samples and effective sample size (ESS), the number of samples used by an independent Monte Carlo estimator with the same variance as the estimator computed by the correlated MCMC samples. The ESS is given [56] by

$$\text{ESS} = \frac{M}{1 + \sum_{t=1}^{\infty} \rho_t}$$

where M is the number of draws and ρ_t is the autocorrelation function at t. Approximate ESS is computed using the Python package `arviz`, which provides a built-in ESS function. The `ess` function in `arviz` computes approximate ESS, $\widehat{\text{ESS}}$, using the formula [57]

$$\widehat{\text{ESS}} = \frac{N_C \cdot M}{-1 + \sum_{t=0}^{K_\rho} \hat{\rho}_{2t} + \hat{\rho}_{2t+1}} \tag{5.3}$$

where N_C is the number of chains, M is the number of samples, $\hat{\rho}_t$ is the estimated autocorrelation function at t, and K_ρ is the last integer such that $\hat{\rho}_{2t} + \hat{\rho}_{2t+1} > 0$.

We distinguish between the CPU time taken to run a sequential algorithm and the total CPU time taken to generate the samples. This is done because the proposal generation phase of RTO is not necessarily sequential and can be parallelised. We can directly compare the inherently sequential algorithms, which are RW, pCN, and the accept-reject phase of RTO. Sequential CPU time per ESS is used in [55] to compare RTO and pCN, as it normalises the effect of multiple chains. The values of these measures are presented in Table 5.10.

Prior	**Method**	**Sequential CPU time (s)**	**Total CPU time (s)**	$\widehat{\textbf{ESS}}$	**Seq. CPU time per $\widehat{\textbf{ESS}}$ (s/sample)**
Gaussian	RW	0.54656	0.54656	276	0.00198
Gaussian	pCN	0.96877	0.96877	263	0.00368
Gaussian	RTO	5.04121	8.67822	281	0.01793
TV	RW	0.92411	0.92411	263	0.00351
TV	pCN	1.20382	1.20382	262	0.00460
TV	RTO	7.51210	365.36354	285	0.02631
Besov (Haar)	RW	5.43124	5.43124	263	0.02064
Besov (Haar)	pCN	9.27566	9.27566	263	0.03529
Besov (Haar)	RTO	33.31040	1930.50193	277	0.12005
Besov (db8)	RW	4.86251	4.86251	261	0.01863
Besov (db8)	pCN	4.33015	4.33015	262	0.01656
Besov (db8)	RTO	24.99918	1904.761905	310	0.08055

Table 5.10: Sequential CPU time, total CPU time, ESS, and sequential CPU time per ESS for the different methods.

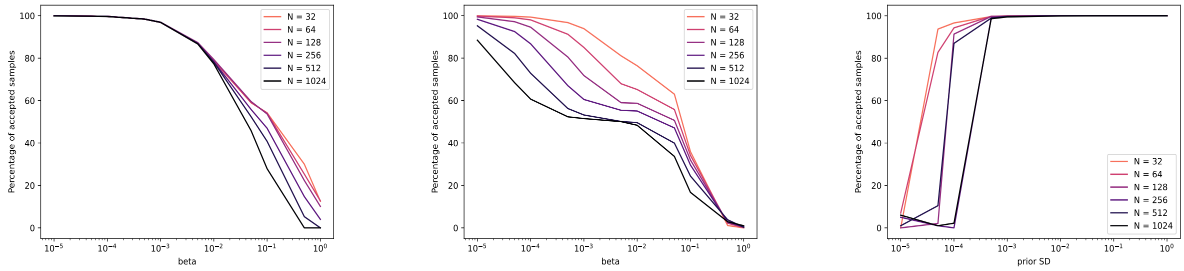
Table 5.10 shows that, terms of computational speed, pCN and RW outperform RTO, having smaller CPU time per ESS values. RTO takes up more total CPU time, as the proposal generation phase requires repeated solving of the optimisation problem (3.18). When only the sequential parts of the algorithms are evaluated, RTO is still slower than pCN and RW, as the accept-reject phase requires the computation of the determinant of $\bar{\mathbf{Q}}^\intercal \mathbf{J}_{\bar{\mathcal{F}}}(\mathbf{u_k})$ in (3.26), which is $\mathcal{O}(n^3)$. Although RTO is far more computationally costly, it also yields conditional mean estimates that are closer to the true signal, with effective noise suppression. This is not necessarily true for pCN, although it is possible that more samples have to be collected for pCN to sample from the posterior distribution.

5.6 Influence of problem dimensions

In 1D deconvolution, the unknown $u(x)$ is a piecewise continuous function, defined on a continuum. Sampling methods are used on discretisations of the problem. We may want to use finer discretisations, with larger n, to obtain closer approximations of the continuous $u(x)$. For the classic random walk Metropolis-Hastings algorithm presented in Section 3.3, the performance of the algorithm degrades as n

increases and more samples are needed to obtain an independent sample [58]. Ideally, the pCN and RTO algorithms would be robust to changes in n.

The percentages of accepted samples when $n = 32, 64, \dots, 1024$ are evaluated for RW, pCN, and RTO with the Gaussian, TV, and Besov priors and shown in Figures 5.20, 5.21, and 5.22.

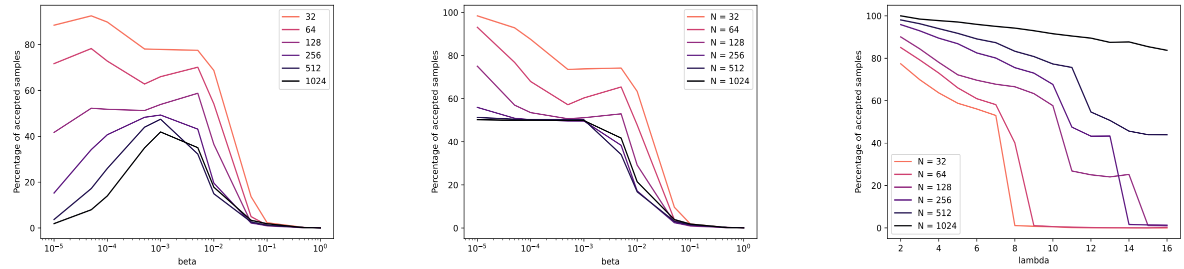


(a) Percentages of accepted samples obtained using RW with a Gaussian smoothness prior for $n = 32, 64, \dots, 1024$.

(b) Percentages of accepted samples obtained using pCN with a Gaussian smoothness prior for $n = 32, 64, \dots, 1024$.

(c) Percentages of accepted samples obtained using RTO with a Gaussian smoothness prior for $n = 32, 64, \dots, 1024$.

Figure 5.20: Plots of percentages of accepted samples vs. grid size for RW, pCN, and RTO with Gaussian priors.

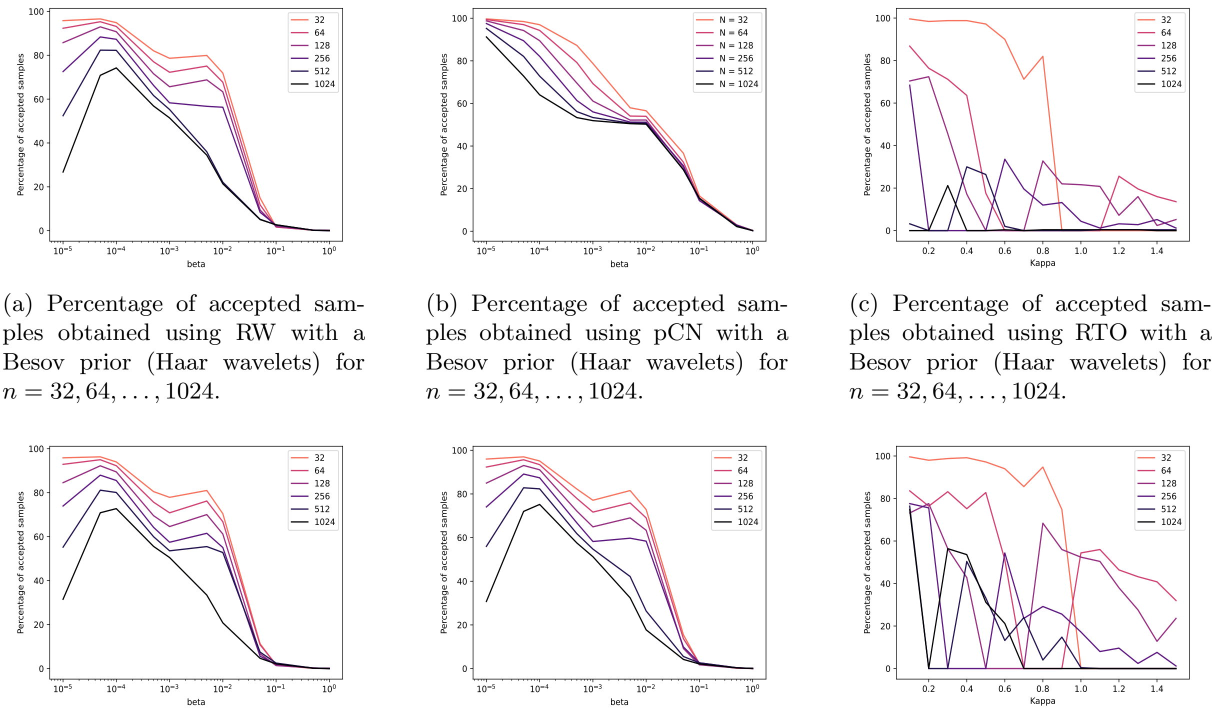


(a) Percentage of accepted samples obtained using RW with a TV prior for $n = 32, 64, \dots, 1024$.

(b) Percentage of accepted samples obtained using pCN with a TV prior for $n = 32, 64, \dots, 1024$.

(c) Percentage of accepted samples obtained using RTO with a TV prior for $n = 32, 64, \dots, 1024$.

Figure 5.21: Plots of percentages of accepted samples vs. grid size for RW, pCN, and RTO with TV priors.



(a) Percentage of accepted samples obtained using RW with a Besov prior (Haar wavelets) for $n = 32, 64, \ldots, 1024$.

(b) Percentage of accepted samples obtained using pCN with a Besov prior (Haar wavelets) for $n = 32, 64, \ldots, 1024$.

(c) Percentage of accepted samples obtained using RTO with a Besov prior (Haar wavelets) for $n = 32, 64, \ldots, 1024$.

(d) Percentage of accepted samples obtained using RW with a Besov prior (db8 wavelets) for $n = 32, 64, \ldots, 1024$.

(e) Percentage of accepted samples obtained using RW with a Besov prior (db8 wavelets) for $n = 32, 64, \ldots, 1024$.

(f) Percentage of accepted samples obtained using RW with a Besov prior (db8 wavelets) for $n = 32, 64, \ldots, 1024$.

Figure 5.22: Plots of percentages of accepted samples vs. grid size for RW, pCN, and RTO with Besov priors.

Figures 5.20a, 5.21a, 5.22a, and 5.22d show that, for all step size β values, random walk Metropolis-Hastings accepts fewer samples as n increases. The deterioriation in performance is particularly severe when the total variation prior is used, as seen in Figure 5.21a.

The percentage of accepted values decreases with increasing n for pCN, which can be seen in Figures 5.20b, 5.21b, 5.22b, and 5.22e. Compared to the random walk, the decreases are more gradual. Figures 5.21b and 5.22b show that, as n increases, the percentage of accepted samples curves grow closer to each other. The probelms sizes shown here are relatively small. In [28], pCN acceptance probabilities plotted against step size are found to be the same for $n = 100, 400, \ldots, 250000$. The maximum n value of 1024 is not large enough to evaluate dimension independence.

For random walk and pCN, the percentages of accepted samples are plotted against the step size parameter β. In RTO, the step size parameter is not present. Instead, the percentages of accepted samples for $n = 32, 64, \ldots, 1024$ are plotted against the prior parameters σ_u, λ and κ. The ranges of values of prior parameters are the same as those used in Chapter 4. In Figures 5.20c and 5.21c, RTO acceptance probabilities are very high ($> 50\%$) for certain prior parameter values and can drop drastically as parameter values change. This indicates that choosing suitable prior parameters are key for RTO performance when the TV and Gaussian priors are used. The performance of RTO with Besov priors is more difficult to parse. In Figures 5.22c and 5.22f, the percentages of accepted samples fluctuate between zero and non-zero values. This behavior explains the sample plots shown in Figures 5.13d, 5.13e, and 5.13f, where RTO often fails to accept proposals for a large proportion ($\approx 40\%$) of the iterations iterations. The estimated posterior mode $\bar{\mathbf{u}}_{MAP}$ serves as the RTO linearisation point and is computed before running RTO. This initial point may be a local minimum of the functional minimised in (3.18), and RTO may have difficulty generating proposals with higher acceptance probabilities.

Chapter 6

Conclusions and discussion

6.1 Conclusions

This thesis addresses the problem of sampling from posterior distributions obtained by solving one-dimensional deconvolution using the Bayesian approach to inverse problems. The one-dimensional deconvolution problem in this thesis is the problem of recovering a piecewise continuous function from noisy measurements of a convolved function. As the true piecewise continuous function has discontinuities, the thesis focuses on the construction of priors that promote sharp features in the solution, namely total variation (TV) priors and Besov space priors. A Gaussian smoothness prior is used as a point of comparison.

To address the computational challenge of sampling from the resulting posteriors, three Markov chain Monte Carlo (MCMC) methods were implemented: the classical random walk Metropolis-Hastings algorithm [18], the preconditioned Crank–Nicholson (pCN) algorithm [28] and the randomise-then-optimise (RTO) [48] algorithm. The latter two methods were originally developed for Bayesian inverse problems with Gaussian priors. Consequently, suitable prior transformations from the literature [31], [41] were used to modify them for sampling with edge-preserving priors. The prior transformations are compositions of invertible linear operators with non-linear multivariate functions. The invertible linear operator is a matrix modeling structural information about the prior in the form of a difference matrix (2.24) or wavelet transformation [43]. The non-linear multivariate functions are compositions of inverse cumulative distribution functions of generalised Gaussian distributions [51] with the cumulative distribution function of the standard n-variate Gaussian distribution. These prior transformations allow total variation and Besov space priors to be transformed to standard Gaussian random variables. The pCN and RTO methods can then be implemented for the inverse problem, which has been restated as an inverse problem with a non-linear forward operator. The non-linearity is introduced by the prior transformation.

Prior parameters were chosen with the aim of comparing the methods using parameters that would lead to the most accurate estimated conditional mean. The most accurate point estimator found in this thesis was the MAP estimator with the TV prior and $\lambda = 16$ with $E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.11445$. The most accurate estimated conditional mean found in this thesis was $\bar{\mathbf{u}}_{est}$ from RTO sampling with a TV prior and $\lambda = 12$ with $E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.14405$. The Gaussian prior, which was used as a point of comparison, was also found to deconvolve the signal, though it does not recover discontinuities and flat regions with the same effectiveness as the TV prior. The Gaussian prior also yields estimated credible intervals that contain some of the true signal for sufficiently large σ_u. The MAP estimators of the posterior densities found using TV priors successfully captured the discontinuities of the true signal, with the exception of the peak of a linear portion of the true signal. This feature was captured more successfully by the estimated conditional means of the posterior densities found using Besov (db8) and TV priors. In this thesis, the TV prior (with the MAP estimator) was found to be the most successful at edge preservation.

The main contribution of the thesis is the comparison of the sampling methods. The numerical results highlight a trade-off between the computational efficiency of the sampling methods and the accuracy of the estimators computed from the samples. When combined with prior transformations, RTO generated independent samples. The means computed using these samples are approximations of the posterior means corresponding to the densities (4.10), (4.11), and (4.12). The estimated conditional means are able to recover the features of the true signal and most of the true signal can be found in estimated 95% credible intervals obtained from the RTO samples. As in the work of [41], the estimated conditional means and credible intervals obtained by sampling with Besov priors are heavily influenced by the choice of wavelet basis functions, with db8 wavelets producing more accurate estimates than Haar wavelets. The effectiveness of RTO comes at a significant computational cost. Each proposal requires the solution of a stochastic optimisation problem, and the acceptance step involves the evaluation of a Jacobian determinant, both of which are expensive to compute. Moreover, the performance of RTO was observed to be highly sensitive to the choice of prior. In the absence of a suitable prior, the algorithm can behave unpredictably. In contrast, the pCN algorithm was found to be computationally efficient and straightforward to implement. It is more robust to changes in discretisation level compared with standard random walk Metropolis-Hastings, which confirms some of its theoretical advantages. Due to the correlations between the samples, it is difficult to determine when pCN has sampled from the posterior distribution. The resulting conditional mean estimates were not close approximations of the true signal and the true signal did not lie in the estimated 95% credible intervals found using pCN, which indicates that the prior parameters were not optimally chosen. Overall, RTO with a prior transformation produces estimated conditional means with smaller relative L2 errors and has better mixing properties than pCN.

Although it is theoretically dimension-independent, pCN performance is affected by the problem dimensions in one-dimensional deconvolution for the problem dimensions tested in this thesis, which are small ($n \leq 1024$) and do not provide enough information about limiting behavior. For $n = 32, 64, \ldots, 1024$, pCN accepts fewer proposals as n increases. For RTO, the effect of the problem dimensions are not as clear from the numerical results in this thesis. The results indicate that the performance of RTO depends on the prior parameters, as RTO does not have a step size parameter that can be tuned for different dimensions.

6.2 Discussion

There are aspects of this thesis that could be improved on to obtain more comprehensive answers to the research questions. The first would be more focus on the factors related to the effectiveness of the sampling methods. In particular, the number of samples needed for RTO and pCN to converge could have been chosen with more attention, for example by choosing the step sizes β of RW and pCN to maximise ESS or finding a number of iterations M_{opt} such that the estimated conditional mean $\bar{\mathbf{u}}_{est}$ no longer changes significantly after the algorithm has been run for M_{opt} iterations. The second would be more thorough study of the Besov priors. A search for an optimal combination of s, q, and κ would have yielded more insight into the edge-preserving properties of the Besov priors. In this thesis and [41], the choice of wavelet basis functions was found to have a significant effect on the point estimators of the Besov posterior density (4.12). While the total variation prior was found to have more significant edge-preserving properties than the Besov priors in this thesis, only the Haar and db8 wavelets were tested. More wavelet basis functions, such as the biorthogonal or other Daubechies wavelets, could have been tested. The third factor would be the number of problem numbers chosen to investigate the second research question. The maximum number of dimensions in this thesis, $n = 1024$, was chosen due to memory limitations. Larger problem dimensions, such as values of n up to 250 000 as in [28], may come closer to showing limiting behavior.

In this thesis, RTO with a prior transformation produced more accurate point estimators of the true signal and has better mixing properties than pCN. However, there are still factors that may make RTO a suboptimal choice for other Bayesian inverse problems, and these factors can provide directions for future work. It would be beneficial to address the effect of problem dimension on performance, as many Bayesian inverse problems concern fine discretisations of parameters defined on a continuum. In [55], the RTO method is extended to function space to establish theoretical dimension-independence and a new subspace strategy is proposed for high-dimensional RTO. The second issue to address, which may be related to the first, is a prior-independent modification to RTO that allows more control over the acceptance probability of the method. The third direction is towards deeper theoretical understanding of the method. In this thesis, the results of sampling with Besov space priors with db8 wavelets show that samples generated using RTO are still useful for uncertainty quantification and point estimation when Assumption (1) in Theorem 3.5.1 is not fulfilled everywhere on the domain of $\tilde{\mathcal{F}}$, as a Jacobian approximation is used when the analytical Jacobian is not available. More investigation into how Jacobian approximations may be used in combination with RTO to tackle inverse problems with more complex forward operators can also make RTO applicable to an even wider class of problems. Additionally, the effects of using linearisation points other than the MAP estimator (3.19) may be investigated. The MAP estimator is a natural choice of linearisation point, as it leads to sampling from high-density areas. Though its existence and uniqueness are key assumptions in the derivation of the RTO proposal density (3.24), the MAP estimator does not always exist and may not be unique. A fourth direction is the potential to make RTO more efficient through parallelisation, as the entire proposal-generation phase of Algorithm 3 can be parallelised.

For preconditioned Crank-Nicholson, an interesting direction of future work would be the study of sample sizes needed to begin sampling from the posterior distribution, as the method has not produced samples that are representative of the posterior distribution in this thesis. For example, a promising starting point would be [59], where a terminating framework for multivariate MCMC is proposed. As pCN with a prior transformation is computationally efficient, relatively simple to implement, derivative-free, and theoretically dimension-independent, it would be a powerful method for sampling from non-Gaussian priors when combined with a more reliable way to evaluate convergence to the posterior distribution. The pCN method has also been extended to take advantage of the geometric properties of the posterior [60]. This modification can be useful for sampling using the non-linear prior transformations used in this thesis.

Bibliography

[1] A. C. Mamourian, "History and physics of ct imaging," in *CT Imaging: Practical Physics, Artifacts, and Pitfalls*, Oxford University Press, Feb. 2013, ISBN: 9780199782604. DOI: 10.1093/med/9780199782604.003.0001. eprint: https://academic.oup.com/book/0/chapter/304214253/chapter-ag-pdf/44503517/book_35487_section_304214253.ag.pdf. [Online]. Available: https://doi.org/10.1093/med/9780199782604.003.0001.

[2] J. G. McNally, T. Karpova, J. Cooper, and J. A. Conchello, "Three-dimensional imaging by deconvolution microscopy," *Methods*, vol. 19, no. 3, pp. 373–385, 1999, ISSN: 1046-2023. DOI: https://doi.org/10.1006/meth.1999.0873. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1046202399908733.

[3] M. Prato, R. Cavicchioli, L. Zanni, P. Boccacci, and M. Bertero, "Efficient deconvolution methods for astronomical imaging: Algorithms and idl-gpu codes," *Astronomy&Astrophysics*, vol. 539, no. A133, 2012. DOI: https://doi.org/10.1051/0004-6361/201118681.

[4] W. A. Mousa, "Seismic deconvolution," in *Advanced Digital Signal Processing of Seismic Data*. Cambridge University Press, 2020, pp. 285–297.

[5] Ö. Yilmaz, "Deconvolution," in *Seismic Data Analysis*. Society of Exploration Geophysicists, 2001, pp. 159–271.

[6] O. Kochukhov, V. Makaganiuk, and N. Piskunov, "Least-squares deconvolution of the stellar intensity and polarization spectra," *Astronomy&Astrophysics*, vol. 524, no. A5, 2010. DOI: https://doi.org/10.1051/0004-6361/201015429.

[7] G. Vivó-Truyols, J. Torres-Lapasió, R. Caballero, and M. García-Alvarez-Coque, "Peak deconvolution in one-dimensional chromatography using a two-way data approach," *Journal of Chromatography A*, vol. 958, no. 1, pp. 35–49, 2002, ISSN: 0021-9673. DOI: https://doi.org/10.1016/S0021-9673(02)00409-0. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021967302004090.

[8] M. D. Hämälginen, Y.-z. Liang, O. Kvalheim, and R. Andersson, "Deconvolution in one-dimensional chromatography by heuristic evolving latent projections of whole profiles retention time shifted by simplex optimization of cross-correlation between target peaks," *Analytica Chimica Acta*, vol. 271, no. 1, pp. 101–114, 1993, ISSN: 0003-2670. DOI: https://doi.org/10.1016/0003-2670(93)80557-2. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0003267093805572.

[9] J. L. Müller and S. Siltanen, *Linear and Nonlinear Inverse Problems with Practical Applications*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2012. DOI: 10.1137/1.9781611972344. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611972344. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611972344.

[10] J. P. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*. Springer-Verlag, 2005. DOI: https://doi.org/10.1007/b138659.

[11] J. Hadamard, *Sur les problèmes aux dérivés partielles et leur signification physique*, 1902.

[12] J. Hadamard, *Lectures on Cauchy's problem in linear partial differential equations*. New Haven Yale University Press, 1923. [Online]. Available: https://archive.org/details/lecturesoncauchy00hadauoft/.

[13] P. C. Hansen, "4. computational aspects: Regularization methods," in *Discrete Inverse Problems*, pp. 53–83. DOI: 10.1137/1.9780898718836.ch4. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9780898718836.ch4. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9780898718836.ch4.

[14] D. L. Phillips, "A technique for the numerical solution of certain integral equations of the first kind," *Journal of the ACAM*, vol. 9, pp. 84–97, 1 1962. DOI: https://doi.org/10.1145/321105.321114.

[15] A. Tikhonov, *Nonlinear Ill-Posed Problems* (Applied Mathematical Sciences). Springer Dordrecht, 1998, ISBN: 978-94-017-5169-8.

[16] A. M. Stuart, "Inverse problems: A bayesian perspective," *Acta Numerica*, vol. 19, pp. 451–559, 2010. DOI: 10.1017/S0962492910000061.

[17] W. K. Hastings, "Monte carlo sampling methods using markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970, ISSN: 00063444, 14643510. [Online]. Available: http://www.jstor.org/stable/2334940 (visited on 08/18/2025).

[18] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, Jun. 1953, ISSN: 0021-9606. DOI: 10.1063/1.1699114. eprint: https://pubs.aip.org/aip/jcp/article-pdf/21/6/1087/18802390/1087_1_online.pdf. [Online]. Available: https://doi.org/10.1063/1.1699114.

[19] A. E. Gelfand and A. F. M. Smith, "Sampling-based approaches to calculating marginal densities," *Journal of the American Statistical Association*, vol. 85, no. 410, pp. 398–409, 1990, ISSN: 01621459, 1537274X. [Online]. Available: http://www.jstor.org/stable/2289776 (visited on 08/18/2025).

[20] J. P. Kaipio, V. Kolehmainen, E. Somersalo, and M. Vauhkonen, "Statistical inversion and monte carlo sampling methods in electrical impedance tomography," *Inverse Problems*, vol. 16, no. 5, p. 1487, Oct. 2000. DOI: 10.1088/0266-5611/16/5/321. [Online]. Available: https://dx.doi.org/10.1088/0266-5611/16/5/321.

[21] H. Haario, E. Saksman, and J. Tamminen, "An adaptive Metropolis algorithm," *Bernoulli*, vol. 7, no. 2, pp. 223–242, 2001.

[22] H. Haario, M. Laine, and A. Mira, " DRAM: Efficient adaptive MCMC," *Statistics & Computing*, vol. 6, pp. 339–354, 2006.

[23] G. Roberts and O. Stramer, "Langevin diffusions and metropolis-hastings algorithms," *Methodology And Computing In Applied Probability*, vol. 4, pp. 337–357, Jan. 2002. DOI: 10.1023/A:1023562417138.

[24] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng, *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, May 2011, ISBN: 9780429138508. DOI: 10.1201/b10905. [Online]. Available: http://dx.doi.org/10.1201/b10905.

[25] M. Lassas and S. Siltanen, "Can one use total variation prior for edge-preserving bayesian inversion?" *Inverse Problems*, vol. 20, no. 5, Aug. 2004. DOI: 10.1088/0266-5611/20/5/013.

[26] L. I. Rudin, "Images, numerical analysis of singularities and shock filters," Ph.D. dissertation, California Institute of Technology, 1987.

[27] M. Lassas, E. Saksman, and S. Siltanen, "Discretization-invariant bayesian inversion and besov space priors," *Inverse Problems and Imaging*, vol. 3, no. 1, pp. 87–122, 2009, ISSN: 1930-8337. DOI: 10.3934/ipi.2009.3.87. [Online]. Available: https://www.aimsciences.org/article/id/2c169f6d-50f3-4e01-9fc0-24c836905147.

[28] S. Cotter, G. Roberts, A. Stuart, and D. White, "Mcmc methods for functions: Modifying old algorithms to make them faster," *Statistical Science*, vol. 28, Feb. 2012. DOI: 10.1214/13-STS421.

[29] J. M. Bardsley, "Mcmc-based image reconstruction with uncertainty quantification," *SIAM Journal on Scientific Computing*, vol. 34, Jan. 2012. DOI: 10.1137/11085760X.

[30] V. Chen, M. Dunlop, O. Papaspiliopoulos, and A. Stuart, "Dimension-robust mcmc in bayesian inverse problems," Mar. 2018.

[31] Z. Wang, J. M. Bardsley, A. Solonen, T. Cui, and Y. M. Marzouk, "Bayesian inverse problems with l_1 priors: A randomize-then-optimize approach," *SIAM Journal on Scientific Computing*, vol. 39, no. 5, S140–S166, 2017. DOI: 10.1137/16M1080938. eprint: https://doi.org/10.1137/16M1080938. [Online]. Available: https://doi.org/10.1137/16M1080938.

[32] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical Mathematics*, 2nd ed. Heidelberg: Springer Berlin, 2007.

[33] C. Robert and G. Casella, *Monte Carlo Statistical Methods*. Springer-Verlag, 2004.

[34] M. A. Lifshits, *Gaussian Random Functions* (Mathematics and Its Applications), 1st ed. Springer Dordrecht, 1995.

[35] M. Lifshits, *Lectures on Gaussian Processes* (SpringerBriefs in Mathematics), 1st ed. Heidelberg: Springer Berlin, 2012, ISBN: 978-3-642-24939-6. DOI: https://doi.org/10.1007/978-3-642-24939-6.

[36] M. Dashti and A. M. Stuart, "The bayesian approach to inverse problems," in *Handbook of Uncertainty Quantification*, R. Ghanem, D. Higdon, and H. Owhadi, Eds. Cham: Springer International Publishing, 2017, pp. 311–428, ISBN: 978-3-319-12385-1. DOI: 10.1007/978-3-319-12385-1_7. [Online]. Available: https://doi.org/10.1007/978-3-319-12385-1_7.

[37] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259–268, 1992, ISSN: 0167-2789. DOI: https://doi.org/10.1016/0167-2789(92)90242-F. [Online]. Available: https://www.sciencedirect.com/science/article/pii/016727899290242F.

[38] M. Dashti, S. Harris, and A. Stuart, "Besov priors for bayesian inverse problems," *Inverse Problems and Imaging*, vol. 6, no. 2, pp. 183–200, 2012, ISSN: 1930-8337. DOI: 10.3934/ipi.2012.6.183. [Online]. Available: https://www.aimsciences.org/article/id/900b5bce-4a6c-4da1-8b93-0b87da3f7df9.

[39] A. Cohen, "Wavelet methods in numerical analysis," in *Solution of Equation in $\mathbb{R}^n$ (Part 3), Techniques of Scientific Computing (Part 3)*, ser. Handbook of Numerical Analysis, vol. 7, Elsevier, 2000, pp. 417–711. DOI: https://doi.org/10.1016/S1570-8659(00)07004-6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1570865900070046.

[40] Y. Meyer, *Wavelets and Operators* (Cambridge Studies in Advanced Mathematics), D. H. Salinger, Ed. Cambridge University Press, 1993.

[41] A. Horst, B. M. Afkham, Y. Dong, and J. Lemvig, "Uncertainty quantification for linear inverse problems with besov prior: A randomize-then-optimize method," *Statistics and Computing*, vol. 35, no. 101, 2025. DOI: https://doi.org/10.1007/s11222-025-10638-2.

[42] I. Daubechies, *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, 1992. DOI: 10.1137/1.9781611970104. eprint: https://epubs.siam.org/doi/pdf/10.1137/1.9781611970104. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611970104.

[43] S. Mallat, *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*, 3rd. USA: Academic Press, Inc., 2008, ISBN: 0123743702.

[44] MathWorks, *Introduction to wavelet families - matlab & simulink*. [Online]. Available: https://uk.mathworks.com/help/wavelet/gs/introduction-to-the-wavelet-families.html.

[45] H. Triebel, *Theory of Function Spaces III* (Monographs in Mathematics), 1st ed. Basel: Birkhäuser Basel, 2006, ISBN: 978-3-7643-7581-2. DOI: https://doi.org/10.1007/3-7643-7582-5.

[46] L. Tierney, "A note on metropolis-hastings kernels for general state spaces," *The Annals of Applied Probability*, vol. 8, no. 1, pp. 1–9, 1998, ISSN: 10505164. [Online]. Available: http://www.jstor.org/stable/2667233 (visited on 08/19/2024).

[47] A. Beskos, G. Roberts, and A. Stuart, "Optimal scalings for local Metropolis–Hastings chains on nonproduct targets in high dimensions," *The Annals of Applied Probability*, vol. 19, no. 3, pp. 863–898, 2009. DOI: 10.1214/08-AAP563. [Online]. Available: https://doi.org/10.1214/08-AAP563.

[48] J. M. Bardsley, A. Solonen, H. Haario, and M. Laine, "Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems," *SIAM Journal on Scientific Computing*, vol. 36, no. 4, A1895–A1910, 2014. DOI: 10.1137/140964023. eprint: https://doi.org/10.1137/140964023. [Online]. Available: https://doi.org/10.1137/140964023.

[49] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3rd ed. USA: Cambridge University Press, 2007, ISBN: 0521880688.

[50] G. R. Lee, R. Gommers, F. Wasilewski, K. Wohlfahrt, and A. O'Leary, "Pywavelets: A python package for wavelet analysis," *Journal of Open Source Software*, vol. 4, no. 36, p. 1237, 2019. DOI: https://doi.org/10.21105/joss.01237.

[51] S. Nadarajah, "A generalized normal distribution," *Journal of Applied Statistics*, vol. 32, no. 7, pp. 685–694, 2005. DOI: 10.1080/02664760500079464. eprint: https://doi.org/10.1080/02664760500079464. [Online]. Available: https://doi.org/10.1080/02664760500079464.

[52] J. Bradbury, R. Frostig, P. Hawkins, *et al.*, *JAX: Composable transformations of Python+NumPy programs*, version 0.3.13, 2018. [Online]. Available: http://github.com/jax-ml/jax.

[53] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, 2017. arXiv: 1412.6980 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1412.6980.

[54] DeepMind, I. Babuschkin, K. Baumli, *et al.*, *The DeepMind JAX Ecosystem*, 2020. [Online]. Available: http://github.com/google-deepmind.

[55] J. M. Bardsley, T. Cui, Y. M. Marzouk, and Z. Wang, "Scalable optimization-based sampling on function space," *SIAM Journal on Scientific Computing*, vol. 42, no. 2, A1317–A1347, 2020. DOI: 10.1137/19M1245220. eprint: https://doi.org/10.1137/19M1245220. [Online]. Available: https://doi.org/10.1137/19M1245220.

[56] C. J. Geyer, "Introduction to markov chain monte carlo," in *Handbook of Markov Chain Monte Carlo*, S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng, Eds. Chapman Hall CRC, 2011, pp. 3–48.

[57] R. Kumar, C. Carroll, A. Hartikainen, and O. Martin, "Arviz a unified library for exploratory analysis of bayesian models in python," *Journal of Open Source Software*, vol. 4, no. 33, p. 1143, 2019. DOI: 10.21105/joss.01143. [Online]. Available: https://doi.org/10.21105/joss.01143.

[58] A. Gelman, W. R. Gilks, and G. O. Roberts, "Weak convergence and optimal scaling of random walk Metropolis algorithms," *The Annals of Applied Probability*, vol. 7, no. 1, pp. 110–120, 1997. DOI: 10.1214/aoap/1034625254. [Online]. Available: https://doi.org/10.1214/aoap/1034625254.

[59] D. Vats, J. M. Flegal, and G. L. Jones, "Multivariate output analysis for markov chain monte carlo," *Biometrika*, vol. 106, no. 2, pp. 321–337, Apr. 2019, ISSN: 0006-3444. DOI: 10.1093/biomet/asz002. eprint: https://academic.oup.com/biomet/article-pdf/106/2/321/28575440/asz002.pdf. [Online]. Available: https://doi.org/10.1093/biomet/asz002.

[60] A. Beskos, M. Girolami, S. Lan, P. E. Farrell, and A. M. Stuart, "Geometric mcmc for infinite-dimensional inverse problems," *Journal of Computational Physics*, vol. 335, pp. 327–351, 2017, ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2016.12.041. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999116307033.

[61] E. Çinlar, *Probability and Stochastics*. Springer-Verlag, 2011.

[62] A. Gut, *Probability: A Graduate Course*, 2nd. ed. New York: Springer New York, NY, 2013.

Appendix A

Probability theory

For more details, the reader may with to consult [61] and [62].

Definition A.0.1: Probability measure

Let μ be a measure on a measurable space $(E, \sigma(E))$. It is called a probability measure if $\mu(E) = 1$.

Definition A.0.2: Probability space

A probability space is a triplet $(\Omega, \sigma(\Omega), \mathbb{P})$ where Ω is a set (called the *sample space*, consisting of elements called *outcomes*), $\sigma(\Omega)$ is a σ-algebra over Ω (consisting of elements called *events*), and $\mathbb{P}$ is a probability measure on $(\Omega, \sigma(\Omega))$.

Definition A.0.3: Random variable

Let $(E, \sigma(E))$ be a measurable space. A mapping $X : \Omega \to E$ is called a random variable taking values in $(E, \sigma(E))$ provided that it is measurable relative to $\sigma(\Omega)$ and $\sigma(E)$, meaning that $X^{-1}A = \{\omega \in \Omega : X(\omega) \in A\}$ is an event in $\sigma(\Omega)$ for every A in $\sigma(E)$. If the $\sigma(E)$ is understood from context, we can say that X takes values in E or is E-valued.

Definition A.0.4: Probability distribution

Let X be a random variable taking values in some measurable space $(E, \sigma(E))$ and let μ be a mapping such that

$$\mu(A) = \mathbb{P}(X^{-1}A) = \mathbb{P}\{X \in A\} \;\; \forall A \in \sigma(E). \tag{A.1}$$

Then μ is a probability measure on E and it is called the distribution of X.

Definition A.0.5: Absolutely continuous measure

Let μ and ν be measures on a measurable space $(E, \sigma(E))$. Then, ν is said to be absolutely continuous with respect to μ if, for every set $A \in \sigma(E)$,

$$\mu(A) = 0 \implies \nu(A) = 0$$

Definition A.0.6: Density

Let $(E, \sigma(E), \mu)$ be a measure space. Let p be a positive $\sigma(E)$-measurable function. Define

$$\nu(A) = \mu(p(x)\mathbb{1}_A(x)) = \int_A p(x)\,\mu(dx) \quad A \in \sigma(E). \tag{A.2}$$

The function p is the density of ν relative to μ.

Definition A.0.7: Radon-Nikodym derivative

Suppose that μ is σ-finite, and ν is absolutely continuous with respect to μ. Then, there exists a positive $\sigma(E)$-measurable function p such that

$$\int_E f(x)\,\nu(dx) = \int_E f(x)p(x)\,\mu(dx) \tag{A.3}$$

for every positive, $\sigma(E)$-measurable f. Moreover, p is unique up to equivalence: if (A.3) holds for another positive, $\sigma(E)$-measurable $\tilde{p}$, then $\tilde{p}(x) = p(x)$ for μ-almost every x in E. The function p is the Radon-Nikodym derivative and can be denoted by $\frac{d\nu}{d\mu}$.

Definition A.0.8: Conditional expectation

Let $(\Omega, \sigma(\Omega), \mathbb{P})$ be a probability space and let $\mathcal{S}$ be a sub-σ-algebra of $\sigma(\Omega)$. The conditional expectation $\mathbb{E}\left(X|\mathcal{S}\right)$ of an integrable random variable X relative to a $\mathcal{S}$ is any $\mathcal{S}$-measurable, integrable random variable Z in the equivalence class of random variables, such that

$$\int_\Lambda Z\,\mathbb{P}(dx) = \int_\Lambda X\,\mathbb{P}(dx) \tag{A.4}$$

for any $\Lambda \in \mathcal{S}$.

Definition A.0.9: Conditional probability

Let $(\Omega, \sigma(\Omega), \mathbb{P})$ be a probability space. Let $\mathcal{F}$ be a sub-σ-algebra of $\sigma(\Omega)$. For each event $A \in \sigma(\Omega)$,

$$\mathbb{P}(A|\mathcal{F}) = \mathbb{E}(\mathbb{1}_A(x)|\mathcal{F}) \tag{A.5}$$

is called the conditional probability of A given $\mathcal{F}$.

Appendix B

Prior transformation properties

B.1 Transformed posterior

The following proposition is used in Section 4.2 to show that the pCN and RTO sampling methods can be used to sample from posterior densities obtained using non-Gaussian priors. It follows from the theory of transfomations of $\mathbb{R}^n$-valued random variables.

Proposition B.1.1

Consider the discrete linear measurement model (2.25). Suppose the random variable $\mathbf{U} : \Omega_1 \to \mathbb{R}^n$ has the prior density $\pi_{\mathbf{U}}(\mathbf{u})$ and the posterior density

$$\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2}||\mathbf{A}(\mathbf{u}) - \mathbf{y}||^2\right) \cdot \pi_{\mathbf{U}}(\mathbf{u}) \tag{B.1}$$

given $\mathbf{y}$, a realisation of $\mathbf{Y} : \Omega \to \mathbb{R}^m$. Suppose $T : \mathbb{R}^n \to \mathbb{R}^n$ is an invertible transformation such that $T(\mathbf{Z}) = \mathbf{U}$, with $\mathbf{Z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The posterior density (B.1) can be written as a function of $\mathbf{z}$,

$$\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2}||\mathbf{A}(T(\mathbf{z})) - \mathbf{y}||^2 - \frac{1}{2}||\mathbf{z}||^2\right).$$

Proof. As T is invertible, the prior density can be written as

$$\pi_{\mathbf{U}}(\mathbf{u}) \propto \exp\left(-\frac{1}{2}||T^{-1}(\mathbf{u})||^2\right) |\mathbf{J}_{T^{-1}}(\mathbf{u})| .$$

Substituting this into the posterior density (B.1), we obtain

$$\begin{aligned}\pi_{\mathbf{U}}^{\mathbf{y}}(\mathbf{z}) &\propto \exp\left(-\frac{1}{2}||\mathbf{A}(T(\mathbf{z}))-\mathbf{y}||^2\right)\pi_{\mathbf{U}}(T(\mathbf{z}))\,|\mathbf{J}_T(\mathbf{z})| \\ &\propto \exp\left(-\frac{1}{2}||\mathbf{A}(T(\mathbf{z}))-\mathbf{y}||^2\right)\exp\left(-\frac{1}{2}||\mathbf{z}||^2\right)|\mathbf{J}_{T^{-1}}(\mathbf{u})|\,|\mathbf{J}_T(\mathbf{z})| \\ &= \exp\left(-\frac{1}{2}||\mathbf{A}(T(\mathbf{z}))-\mathbf{y}||^2-\frac{1}{2}||\mathbf{z}||^2\right).\end{aligned} \tag{B.2}$$

□

We now show that the assumptions in Proposition B.1 are satisfied by the transformations T_λ (4.4) and $T_{\tau,q}$ (4.9).

B.2 Total variation prior transformation properties

The transformation T_λ (4.4) is invertible. Note that $g_\lambda(z) = \mathcal{L}^{-1}(\Phi(z))$, where $\mathcal{L}$ is the Laplace cumulative distribution function. The inverse of $g_\lambda(z)$ thus exists and is written as $g_\lambda^{-1}(u) = \Phi^{-1}(\mathcal{L}(u))$. Let

$$G_\lambda^{-1}(\mathbf{x}) = \begin{pmatrix} g_\lambda^{-1}(\mathbf{x}_0) & g_\lambda^{-1}(\mathbf{x}_1) & \dots & g_\lambda^{-1}(\mathbf{x}_{n-2}) & g_\lambda^{-1}(\mathbf{x}_{n-1})\end{pmatrix}^{\intercal}. \tag{B.3}$$

The inverse of T_λ is then

$$T_\lambda^{-1}(\mathbf{u}) = G_\lambda^{-1}(\mathbf{D}\mathbf{u}).$$

This allows us to use Proposition B.1 to rewrite the posterior density (4.11) in the forms (4.14) and (4.15).

The Jacobian of the transformation (4.4) is needed for RTO. The function g_λ (4.1.2) is differentiable and

$$g_\lambda'(z) = \frac{\Phi'(z)}{\lambda\Phi(-|z|)}.$$

This function is continuous on $\mathbb{R}$. The Jacobian of $T_\lambda(\mathbf{z})$ is thus given by

$$\mathbf{J}_{T_\lambda}(\mathbf{z}) = \mathbf{D}^{-1}\begin{pmatrix} g_\lambda'(\mathbf{z}_0) & 0 & \cdots & 0 & 0 \\ 0 & g_\lambda'(\mathbf{z}_1) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & g_\lambda'(\mathbf{z}_{n-2}) & 0 \\ 0 & 0 & \cdots & 0 & g_\lambda'(\mathbf{z}_{n-1})\end{pmatrix}.$$

The transformation T_λ is continously differentiable.

B.3 Besov prior transformation properties

The transformation $T_{\tau,q}$ (4.9) is invertible. Note that $g_{\tau,q}(z) = \Phi_{\tau,q}^{-1}(\Phi(z))$. The inverse of $g_{\tau,q}(z)$ thus exists and is written as $g_{\tau,q}^{-1}(u) = \Phi^{-1}(\Phi_{\tau,q}(u))$. Let

$$G_{\tau,q}^{-1}(\mathbf{x}) = \begin{pmatrix} g_{\tau,q}^{-1}(\mathbf{x}_0) & g_{\tau,q}^{-1}(\mathbf{x}_1) & \dots & g_{\tau,q}^{-1}(\mathbf{x}_{n-2}) & g_{\tau,q}^{-1}(\mathbf{x}_{n-1}) \end{pmatrix}^{\mathsf{T}}. \tag{B.4}$$

The inverse of $T_{\tau,q}$ is then

$$T_{\tau,q}^{-1}(\mathbf{u}) = G_{\tau,q}^{-1}(\mathbf{SWu}).$$

This allows us to use Proposition B.1 to rewrite the posterior density (4.12) in the forms (4.14) and (4.15).

The Jacobian of the transformation (4.4) is needed for RTO. The function $g_{\tau,q}$ (4.8) is differentiable and its derivative is

$$g_{\tau,q}'(z) = \frac{\tau}{q}\Gamma\left(\frac{1}{q}\right)\exp\left(Q^{-1}\left(\frac{1}{q}, 1 + \operatorname{sgn}(z) - 2\operatorname{sgn}(z)\Phi(z)\right)\right)(2\operatorname{sgn}(z)\Phi'(z) - \operatorname{sgn}(z)).$$

We note that the derivative $g_{\tau,q}'(z)$ is not continuous at $z = 0$. The Jacobian of $T_{\tau,q}(\mathbf{z})$ is

$$\mathbf{J}_{T_{\tau,q}}(\mathbf{z}) = (\mathbf{SW})^{-1}\begin{pmatrix} g_{\tau,q}'(\mathbf{z}_0) & 0 & \cdots & 0 & 0 \\ 0 & g_{\tau,q}'(\mathbf{z}_1) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & g_{\tau,q}'(\mathbf{z}_{n-2}) & 0 \\ 0 & 0 & \cdots & 0 & g_{\tau,q}'(\mathbf{z}_{n-1}) \end{pmatrix}.$$

B.4 RTO conditions with a prior transformation

We check the RTO validity conditions 3.5.1 for sampling with prior transformations. This is Theorem 3.2 in [31]. It is stated for the posterior density (4.11) with the TV transformation (4.4).

Theorem B.4.1: RTO validity conditions with a prior transformation

Let (4.11) specify the posterior density of a Bayesian inference problem with pa- rameters $\mathbf{U}$, and let the forward model in (4.11) be linear and denoted by $\mathbf{A}$. After the prior transformation (4.4), the RTO algorithm described by Algorithm 3 generates proposal samples with probability density given in (3.24) where $\tilde{\mathcal{F}}$ is as written in (4.16), with $T = T_\lambda$.

Proof. We check that the assumptions in Theorem 3.5.1 are fulfilled. Since T_λ is continuously differentiable, conditions (1) and (2) are fulfilled. Note that the Jacobian of $\tilde{\mathcal{F}}$ is

$$\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{z}) = \begin{pmatrix} \mathbf{I} \\ \frac{1}{\sigma_e}\mathbf{A}\mathbf{J}_{T_\lambda}(\mathbf{z}) \end{pmatrix}. \tag{B.5}$$

The identity matrix in the first n rows of $\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{z})$ ensure that the columbs of $\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{z})$ are linearly independent regardless of $\mathbf{J}_{T_\lambda}(\mathbf{z})$. Assumption (4) in Theorem 3.5.1 will now be checked. Let

$$\mathbf{J}_{G_\lambda}(\mathbf{z}) = \begin{pmatrix} g'_\lambda(\mathbf{z}_0) & 0 & \cdots & 0 & 0 \\ 0 & g'_\lambda(\mathbf{z}_1) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & g'_\lambda(\mathbf{z}_{n-2}) & 0 \\ 0 & 0 & \cdots & 0 & g'_\lambda(\mathbf{z}_{n-1}) \end{pmatrix}. \tag{B.6}$$

For any $\mathbf{v} \in \mathbb{R}^n, \mathbf{w} \in \mathbb{R}^n$,

$$\begin{aligned} \mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{v})^\intercal \mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{w}) &= \mathbf{I} + \frac{1}{\sigma_\mathbf{e}^2}\mathbf{J}_{G_\lambda}(\mathbf{v})\mathbf{D}^{-\intercal}\mathbf{A}^\intercal\mathbf{A}\mathbf{D}^{-1}\mathbf{J}_{G_\lambda}(\mathbf{w}) \\ &= \mathbf{J}_{G_\lambda}(\mathbf{v})(\mathbf{J}_{G_\lambda}(\mathbf{v})^{-1}\mathbf{J}_{G_\lambda}(\mathbf{w})^{-1} + \frac{1}{\sigma_\mathbf{e}^2}\mathbf{D}^{-\intercal}\mathbf{A}^\intercal\mathbf{A}\mathbf{D}^{-1})\mathbf{J}_{G_\lambda}(\mathbf{w}). \end{aligned}$$

□

For any $\mathbf{z} \in \mathbb{R}^n$, the matrix $\mathbf{J}_{G_\lambda}(\mathbf{z})$ is a positive diagonal matrix. The matrix $\frac{1}{\sigma_\mathbf{e}^2}\mathbf{D}^{-\intercal}\mathbf{A}^\intercal\mathbf{A}\mathbf{D}^{-1}$ is symmetric positive definite. The matrix

$$(\mathbf{J}_{G_\lambda}(\mathbf{v})^{-1}\mathbf{J}_{G_\lambda}(\mathbf{w})^{-1} + \frac{1}{\sigma_\mathbf{e}^2}\mathbf{D}^{-\intercal}\mathbf{A}^\intercal\mathbf{A}\mathbf{D}^{-1})$$

is then symmetric positive definite. Therefore, $\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{v})^\intercal\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{w})$ is a product of three invertible matrices and is also invertible. The matrix $\bar{\mathbf{Q}}$ is obtained from the thin-QR decomposition of $\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP})$ at the posterior mode $\bar{\mathbf{u}}_{MAP}$. The matrix $\mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{u})^\intercal\bar{\mathbf{Q}} = \mathbf{J}_{\tilde{\mathcal{F}}}(\mathbf{u})^\intercal\mathbf{J}_{\tilde{\mathcal{F}}}(\bar{\mathbf{u}}_{MAP})\bar{\mathbf{R}}^{-1}$ is invertible for any $\mathbf{u} \in \mathbb{R}^n$, showing that Assumption (4) holds. The assumptions of Theorem 3.5.1 are therefore fulfilled.

Appendix C

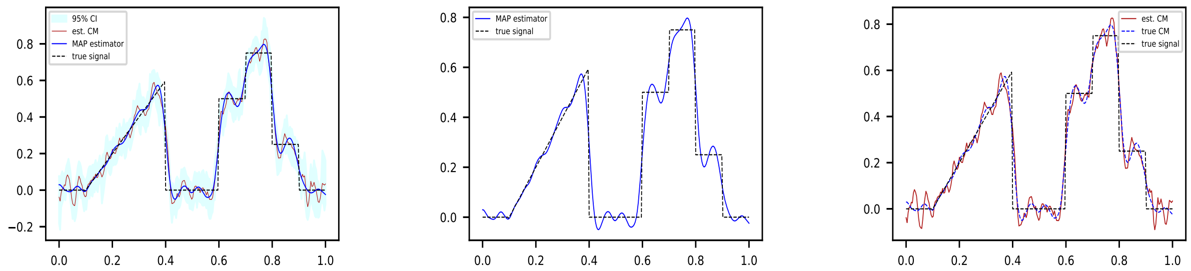
Additional point estimator plots

Plots of point estimators with the parameters in Table 5.9 are presented in this Appendix. In addition to the L2 error relative to the true signal, another metric will be shown with some of the plots. This is the L2 error of $\mathbf{a}$ relative to $\mathbf{b}$, $E_{L2}\left(\mathbf{a}, \mathbf{b}\right)$, which is used to compare different point estimators in this Appendix. The formula for $E_{L2}\left(\mathbf{a}, \mathbf{b}\right)$ is given by

$$E_{L2}\left(\mathbf{a}, \mathbf{b}\right) = \frac{||\mathbf{a} - \mathbf{b}||_2}{||\mathbf{b}||_2}. \tag{C.1}$$

Note that, for the Gaussian posterior density (4.10), the conditional mean $\mathbf{u}_{CM}$ (2.35) is the same as the MAP estimator $\mathbf{u}_{MAP}$. In this Appendix, they are treated as distinct only due to the methods used to compute them; the true conditional mean $\mathbf{u}_{CM}$ is calculated using the formula (2.35) and the MAP estimator $\mathbf{u}_{MAP}$ is found by using the `Adam` optimiser from the Python library `optax` to solve the MAP estimation problem (2.30).

C.1 Point estimators from Gaussian priors

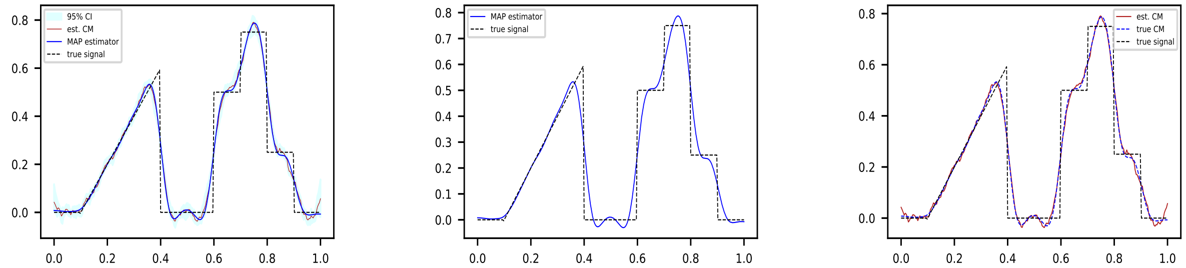


(a) RW $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Gaussian smoothness prior, $\sigma_u = 0.05, \beta = 5.5 \times 10^{-3}, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.09668$.

(b) Gaussian smoothness prior $\mathbf{u}_{MAP}$, $\sigma_u = 0.05, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.16369$.

(c) Comparison of RW $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{CM}$ with a Gaussian smoothness prior, $\sigma_u = 0.05, \beta = 5.5 \times 10^{-3}, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{CM}\right) = 0.09668$.

Figure C.1: Point estimators of the posterior density with a Gaussian prior (4.10), with $\bar{\mathbf{u}}_{est}$ found using RW samples.

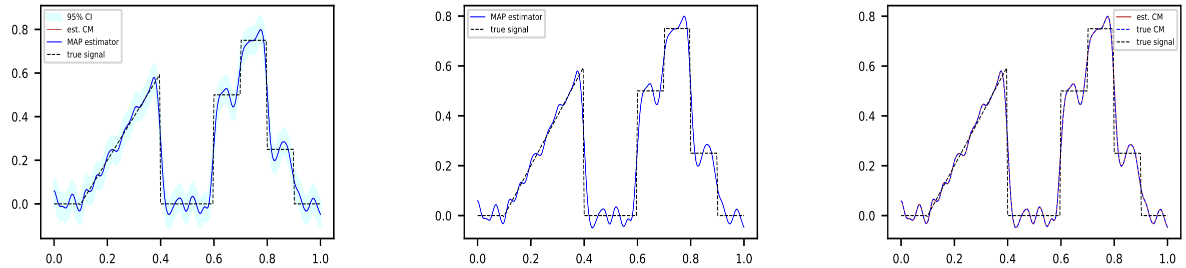


(a) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Gaussian smoothness prior, $\sigma_u = 0.01, \beta = 7.7 \times 10^{-2}, E_{L2}(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}) = 0.035770$.

(b) Gaussian smoothness prior $\mathbf{u}_{MAP}$, $\sigma_u = 0.01, E_{L2}(\mathbf{u}_{MAP}) = 0.19539$.

(c) Comparison of pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{CM}$ with a Gaussian smoothness prior, $\sigma_u = 0.01, \beta = 5.5 \times 10^{-3}, E_{L2}(\bar{\mathbf{u}}_{est}, \mathbf{u}_{CM}) = 0.035770$

Figure C.2: Point estimators of the posterior density with a Gaussian prior (4.10), with $\bar{\mathbf{u}}_{est}$ found using pCN samples.



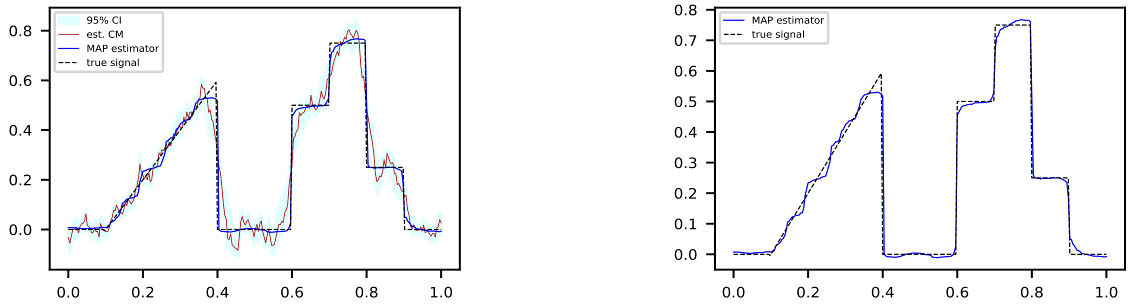
(a) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Gaussian smoothness prior, $\sigma_u = 0.1, E_{L2}(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}) = 0.00274$

(b) Gaussian smoothness prior $\mathbf{u}_{MAP}$, $\sigma_u = 0.1, E_{L2}(\mathbf{u}_{MAP}) = 0.15713$

(c) Comparison of RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{CM}$ with a Gaussian smoothness prior, $\sigma_u = 0.1, E_{L2}(\bar{\mathbf{u}}_{est}, \mathbf{u}_{CM}) = 0.00274$.

Figure C.3: Point estimators of the posterior density with a Gaussian prior (4.10), with $\bar{\mathbf{u}}_{est}$ found using RTO samples.

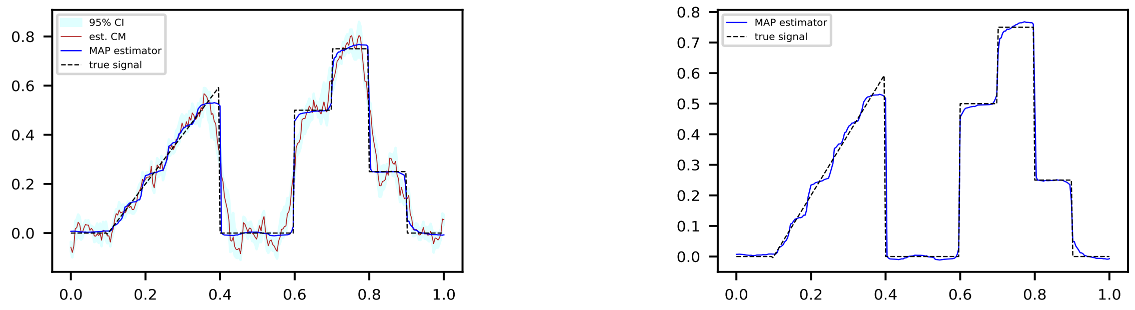
C.2 Point estimators from total variation priors



(a) RW $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a TV prior, $\lambda = 16, \beta = 9.10 \times 10^{-3}, E_{L2}(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}) = 0.16833$

(b) TV $\mathbf{u}_{MAP}, \lambda = 16, E_{L2}(\mathbf{u}_{MAP}) = 0.11445$.

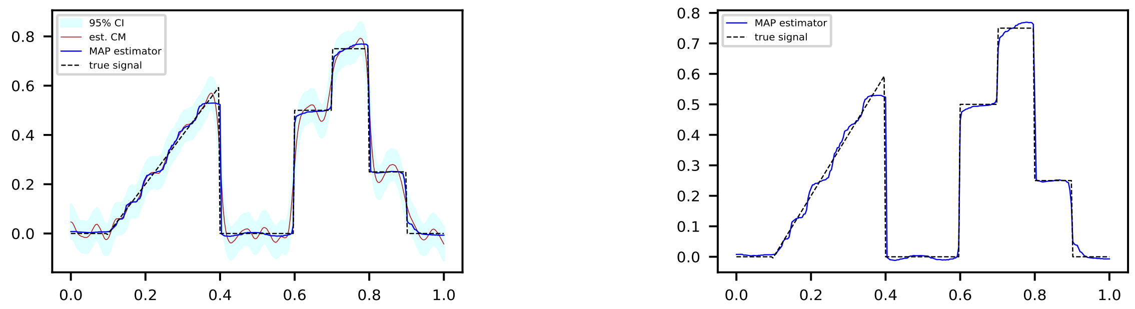
Figure C.4: Point estimators of the posterior density with a TV prior (4.11), with $\bar{\mathbf{u}}_{est}$ found using RW samples.



(a) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a TV prior, $\lambda = 16, \beta = 9.50 \times 10^{-3}, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.17137$

(b) TV $\mathbf{u}_{MAP}, \lambda = 16, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.11445.$

Figure C.5: Point estimators of the posterior density with a TV prior (4.11), with $\bar{\mathbf{u}}_{est}$ found using pCN samples.

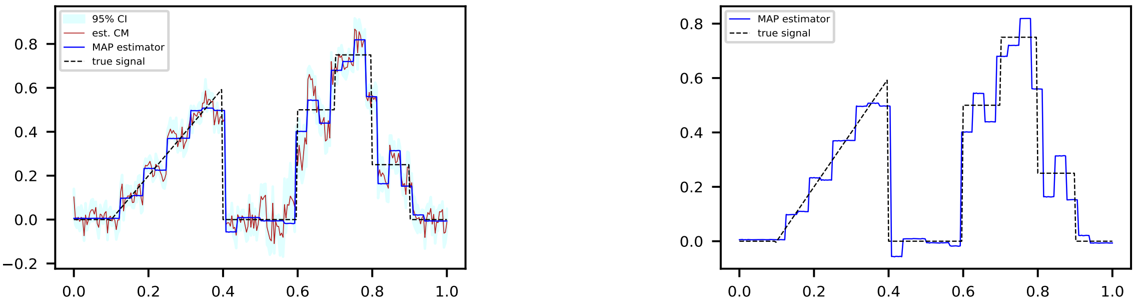


(a) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a TV prior, $\lambda = 12, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.11046.$

(b) TV $\mathbf{u}_{MAP}, \lambda = 12, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.11476.$

Figure C.6: Point estimators of the posterior density with a TV prior (4.11), with $\bar{\mathbf{u}}_{est}$ found using pCN samples.

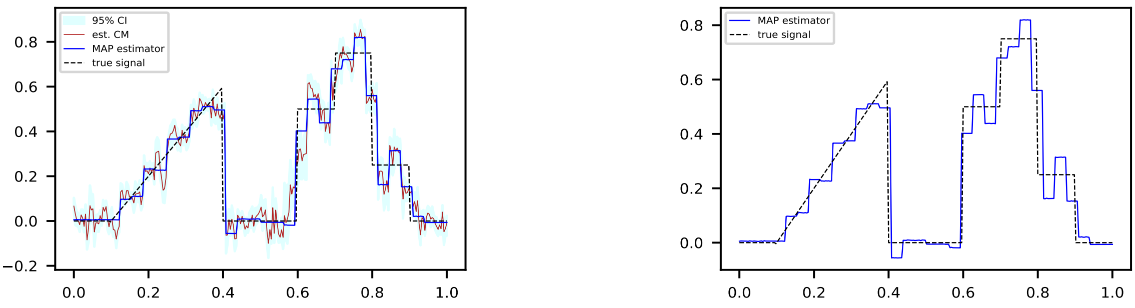
C.3 Point estimators from Besov (Haar) priors



(a) RW $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 1, q = 1, \kappa = 1.4, \beta = 3.43 \times 10^{-3}, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.13596$

(b) Besov (Haar) $\mathbf{u}_{MAP}, s = 1, q = 1, \kappa = 1.4, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.24134.$

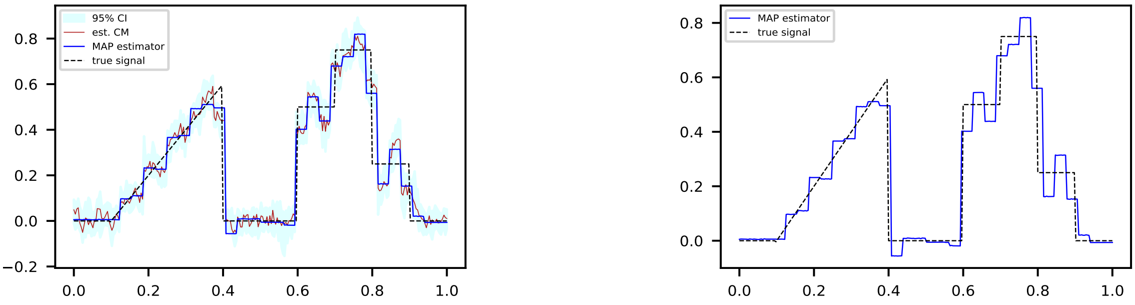
Figure C.7: Point estimators of the posterior density with a Besov (Haar) prior (4.12), with $\bar{\mathbf{u}}_{est}$ found using RW samples.



(a) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 1, q = 1, \kappa = 1.3, \beta = 3.0 \times 10^{-2}, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.13428$.

(b) Besov (Haar) $\mathbf{u}_{MAP}, s = 1, q = 1, \kappa = 1.3, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.24080$.

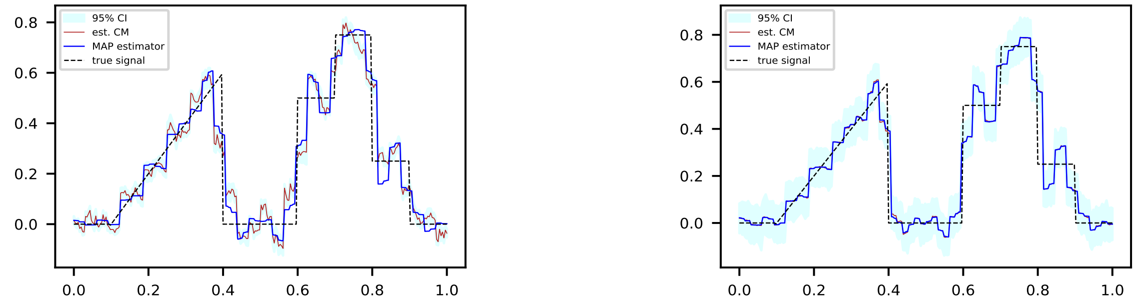
Figure C.8: Point estimators of the posterior density with a Besov (Haar) prior (4.12), with $\bar{\mathbf{u}}_{est}$ found using pCN samples.



(a) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 1, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.08087$.

(b) Besov (Haar) $\mathbf{u}_{MAP}, s = 1, q = 1, \kappa = 1.3, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.24080$.

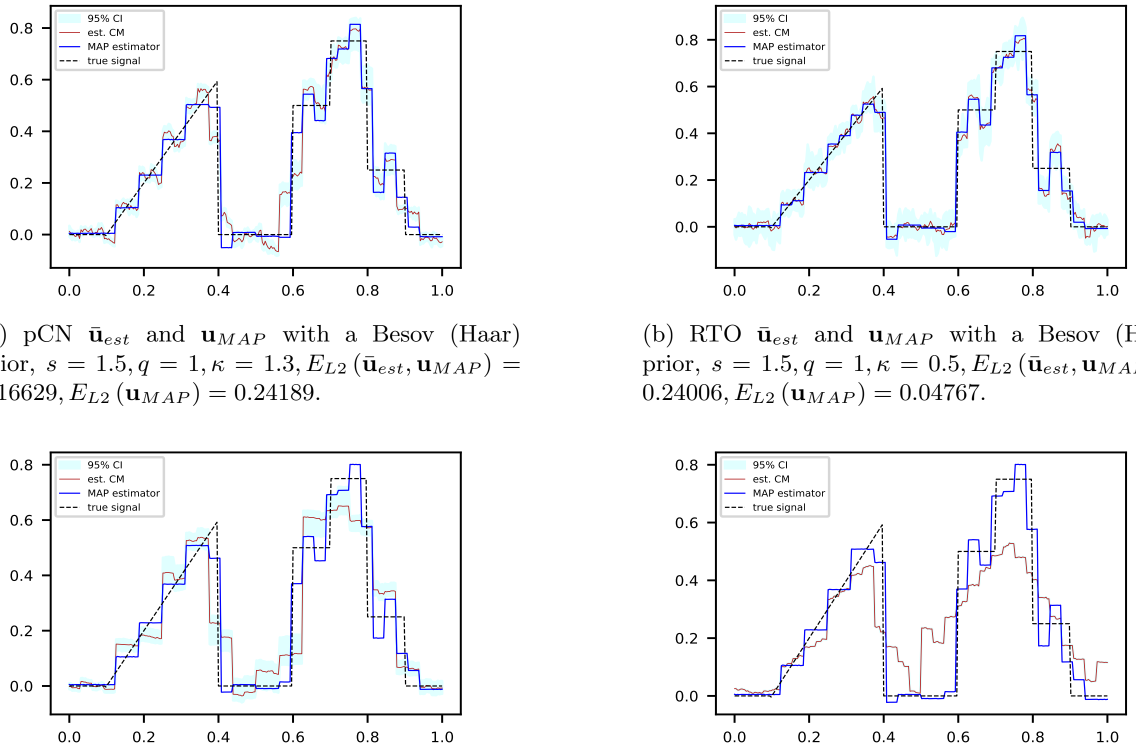
Figure C.9: Point estimators of the posterior density with a Besov (Haar) prior (4.12), with $\bar{\mathbf{u}}_{est}$ found using RTO samples.



(a) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 1, q = 2, \kappa = 0.15, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.11803, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.24708$.

(b) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 1, q = 2, \kappa = 0.05, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.23502, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.00918$.

Figure C.10: Point estimators of the posterior density with a Besov (Haar) prior (4.12), with $q = 2$.



(a) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 1.5, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.16629, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.24189$.

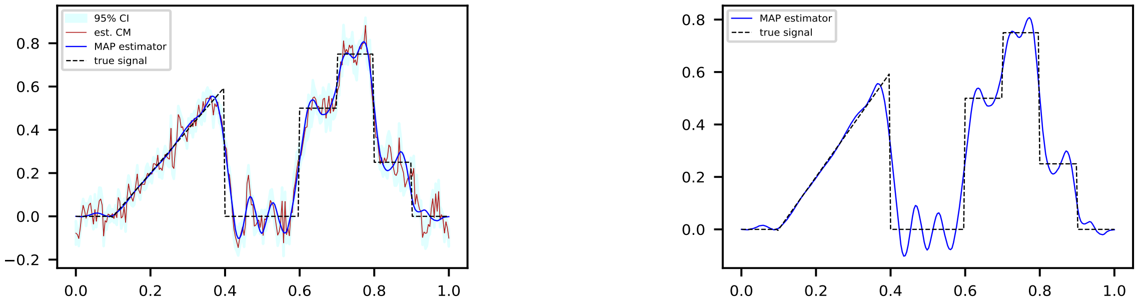
(b) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 1.5, q = 1, \kappa = 0.5, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.24006, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.04767$.

(c) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 2.0, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.27720, E_{L2}\left(\bar{\mathbf{u}}_{est}\right) = 0.24296$.

(d) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 2.0, q = 1, \kappa = 1.3, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.24296, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.38630$.

Figure C.11: Point estimators of the posterior density with a Besov (Haar) prior (4.12), with $q = 1$ and $s \in \{1.5, 2.0\}$.

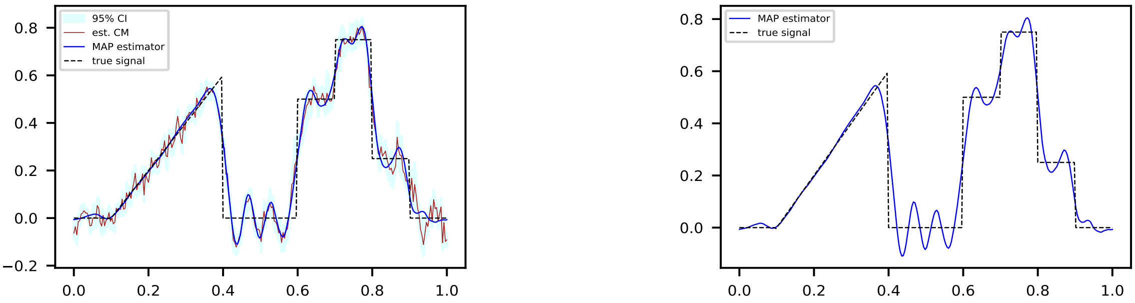
C.4 Point estimators from Besov (db8) priors



(a) RW $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (db8) prior, $s = 1, q = 1, \kappa = 1.2, \beta = 2.35 \times 10^{-3}, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.14748$.

(b) Besov (db8) $\mathbf{u}_{MAP}, s = 1, q = 1, \kappa = 1.2, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.18105$.

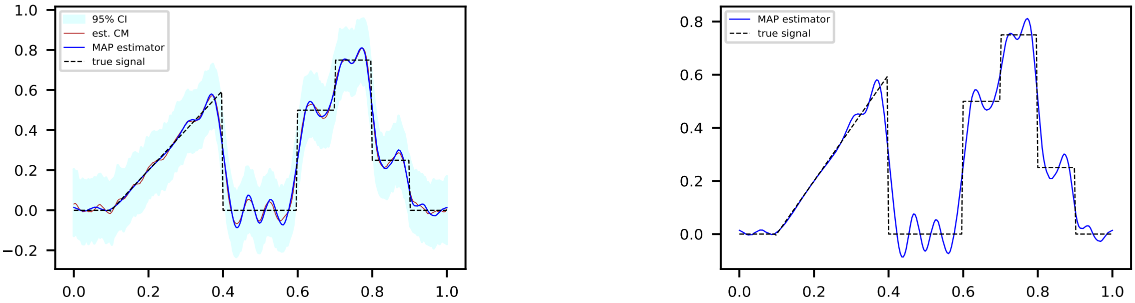
Figure C.12: Point estimators of the posterior density with a Besov (db8) prior (4.12), with $\bar{\mathbf{u}}_{est}$ found using RW samples.



(a) RW $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (db8) prior, $s = 1, q = 1, \kappa = 1.5, \beta = 4.30 \times 10^{-3}, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.10249$.

(b) Besov (db8) $\mathbf{u}_{MAP}, s = 1, q = 1, \kappa = 1.5, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.18429$.

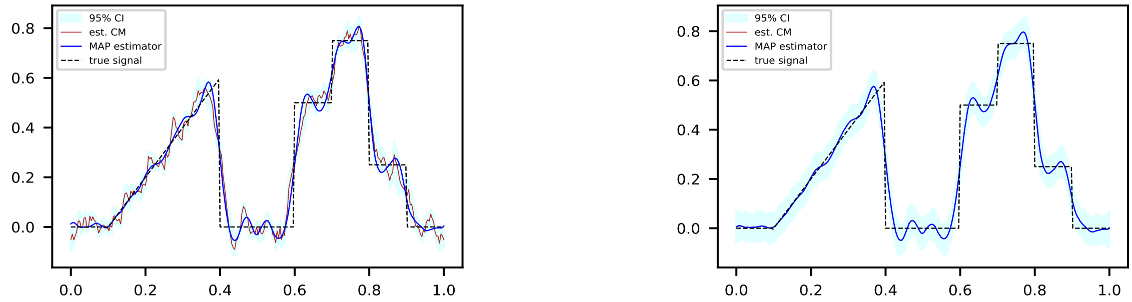
Figure C.13: Point estimators of the posterior density with a Besov (db8) prior (4.12), with $\bar{\mathbf{u}}_{est}$ found using pCN samples.



(a) RW $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (db8) prior, $s = 1, q = 1, \kappa = 0.6, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.03667$.

(b) Besov (db8) $\mathbf{u}_{MAP}, s = 1, q = 1, \kappa = 0.6, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.17596$.

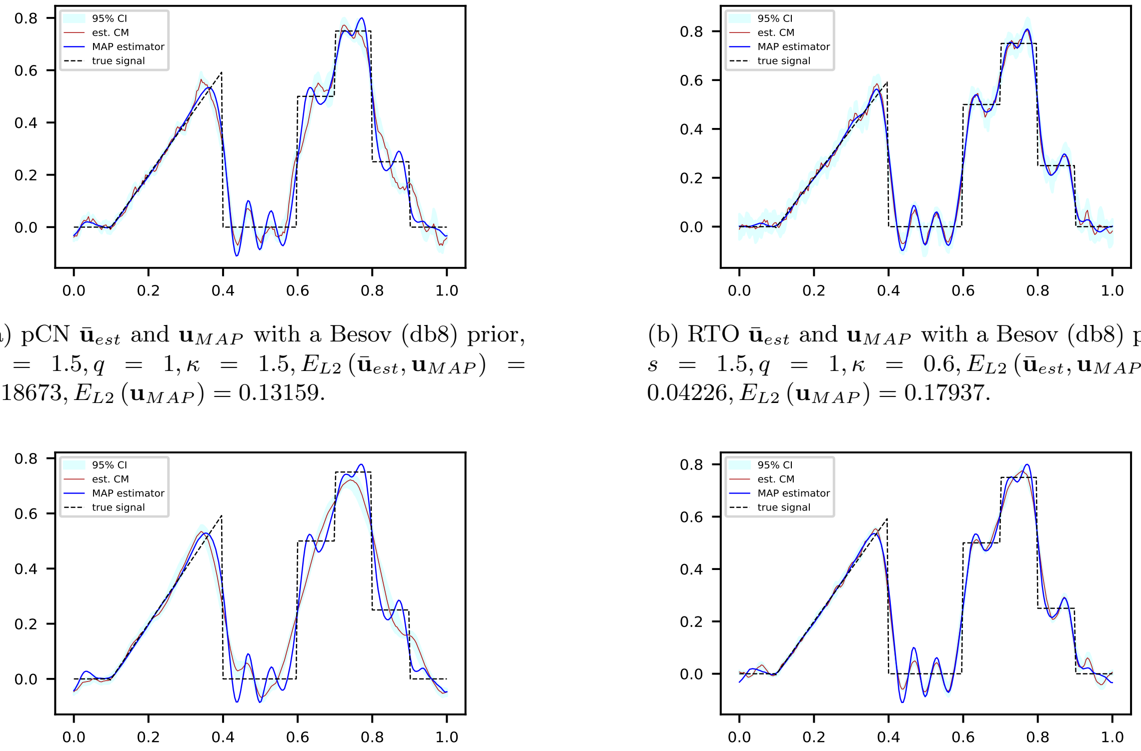
Figure C.14: Point estimators of the posterior density with a Besov (db8) prior (4.12), with $\bar{\mathbf{u}}_{est}$ found using RTO samples.



(a) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (db8) prior, $s = 1, q = 2, \kappa = 0.05, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.10333, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.16450$.

(b) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 1, q = 2, \kappa = 0.1, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.00276, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.16810$.

Figure C.15: Point estimators of the posterior density with a Besov (db8) prior (4.12), with $q = 2$.



(a) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (db8) prior, $s = 1.5, q = 1, \kappa = 1.5, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.18673, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.13159$.

(b) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (db8) prior, $s = 1.5, q = 1, \kappa = 0.6, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.04226, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.17937$.

(c) pCN $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (Haar) prior, $s = 2.0, q = 1, \kappa = 1.5, E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.15765, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.19059$.

(d) RTO $\bar{\mathbf{u}}_{est}$ and $\mathbf{u}_{MAP}$ with a Besov (db8) prior, $s = 2.0, q = 1, \kappa = 0., E_{L2}\left(\bar{\mathbf{u}}_{est}, \mathbf{u}_{MAP}\right) = 0.05855, E_{L2}\left(\mathbf{u}_{MAP}\right) = 0.18645$.

Figure C.16: Point estimators of the posterior density with a Besov (db8) prior (4.12), with $q = 1$ and $s \in \{1.5, 2.0\}$.