THE UNIVERSITY *of* EDINBURGH

Accelerating Bayesian computation in imaging

Luis A. Vargas-Mieles

Doctor of Philosophy
University of Edinburgh
2022

Declaration

I declare that this thesis was composed by myself and that the work contained therein is my own, except where explicitly stated otherwise in the text.

(*Luis A. Vargas-Mieles*)

To my wife Daniela and my mum Ruth, the most important pillars of my life.

Acknowledgements

Moving from Ecuador to pursue a PhD in a country thousands of miles away from my parents and friends is quite an experience! An experience that, for a developing country citizen, is almost impossible to contemplate. Therefore, I wanted to start by thanking my supervisors, Kostas Zygalakis and Marcelo Pereyra, for trusting in me and allowing me to make this possible. You have no idea how life-changing this experience has been for me and my family, so thank you for being so patient with me in this learning process.

Being a student with a family and dealing with a global pandemic that seriously afflicted my country and my hometown of Guayaquil during its early months made this path more difficult to travel. It was pretty challenging to carry out my obligations hundreds of miles away while dealing with so much bad news, in addition to the difficulties that come with a PhD.

When I felt entirely frustrated and without the strength to go on, my wife Daniela was always there to support me, divert my attention, and lift me out of those unpleasant circumstances. She was crucial to this process, and I can confidently say that without her, I would not have been able to complete my PhD. But because of her, I was able to persevere, and as a result, I am now completing this road. Thank you, Daniela, for your words, for drying my tears and for all the laughter you provided to keep me going. I love you. This thesis is thanks to you.

I also want to thank my son Matías for his patience. I sincerely apologise for the times I could not accept your requests to play or spend time with you. I promise that when you grow up, you will realise that all of my efforts will have been worth it.

Finally, this adventure would not have been possible without more than thirty years of love and support from my dad Luis, but above all, the help and patience of my mum Ruth. She played a significant role in this project as she has supported me with my university degree, my English courses and especially with my decision to travel to Scotland. I have no words to express the enormous gratitude I feel for her. I love you, mum. I hope this makes you proud of me. I know it will.

For all the people I mentioned above, here is my Thesis.

Lay Summary

Decision-making processes require robust tools in domains as essential as medicine and astronomy to support and deliver reliable judgments in critical situations. For instance, if an atypical object were to appear on the CT image of a patient, it would be beneficial if a doctor also had tools that provided a degree of assurance that the object was indeed inside the patient's body. This would further help the doctor make accurate diagnostic and treatment decisions.

In this context, Bayesian computation provides efficient tools to quantify the uncertainty in these situations. These methods are usually admissible from the computational point of view in small dimensional applications, i.e., it takes a few minutes to obtain results, for example, in applications where one is interested in the time evolution of a few quantities, such as forecasting stock prices or the spread rates of COVID-19. However, this is not the case for imaging applications, where one wants to analyse images with hundreds of thousands of pixels, i.e., dimensions.

For performing the Bayesian analysis required in large imaging applications, it is necessary to have methods that provide accurate results in an allowable amount of time. The development of efficient Bayesian approaches for extremely high dimensional applications, such as imaging, has been one of the main focuses of the Bayesian imaging community.

Despite the significant efforts of the scientific community in recent decades, the amount of data handled by new applications is expanding quickly, and the methods developed just a few years ago are starting to become obsolete with the sheer volume of information produced by next-generation applications. In February 2022, for instance, as part of the calibration and alignment procedure, the James Webb Space Telescope generated a picture mosaic of over 2 billion pixels.

In light of these current challenges, we present in this thesis three novel Bayesian methods, which significantly outperform existing state-of-the-art approaches in speed and/or accuracy, as demonstrated by the theory and numerical experiments developed in this work.

Abstract

The dimensionality and ill-posedness often encountered in imaging inverse problems are a challenge for Bayesian computational methods, particularly for state-of-the-art sampling alternatives based on the Euler-Maruyama discretisation of the Langevin diffusion process. In this thesis, we address this difficulty and propose alternatives to accelerate Bayesian computation in imaging inverse problems, focusing on its computational aspects.

We introduce, as our first contribution, a highly efficient proximal Markov chain Monte Carlo (MCMC) methodology, based on a state-of-the-art approximation known as the proximal stochastic orthogonal Runge-Kutta-Chebyshev (SK-ROCK) method. It has the advantage of cleverly combining multiple gradient evaluations to significantly speed up convergence, similar to accelerated gradient optimisation techniques. We rigorously demonstrate the acceleration of the Markov chains in the 2-Wasserstein distance for Gaussian models as a function of the condition number κ.

In our second contribution, we propose a more sophisticated MCMC sampler, based on the careful integration of two advanced proximal Langevin MCMC methods, SK-ROCK and split Gibbs sampling (SGS), each of which uses a unique approach to accelerate convergence. More precisely, we show how to integrate the proximal SK-ROCK sampler with the model augmentation and relaxation method used by SGS at the level of the Langevin diffusion process, to speed up Bayesian computation at the expense of asymptotic bias. This leads to a new, faster proximal SK-ROCK sampler that combines the accelerated quality of the original sampler with the computational advantages of augmentation and relaxation.

Additionally, we propose the augmented and relaxed model to be considered a generalisation of the target model rather than an approximation that situates relaxation in a bias-variance trade-off. As a result, we can carefully calibrate the amount of relaxation to boost both model accuracy (as determined by model evidence) and sampler convergence speed. To achieve this, we derive an empirical Bayesian method that automatically estimates the appropriate level of relaxation via maximum marginal likelihood estimation.

The proposed methodologies are demonstrated in several numerical experiments related to image deblurring, hyperspectral unmixing, tomographic reconstruction and inpainting. Comparisons with Euler-type proximal Monte Carlo approaches confirm that the Markov chains generated with our methods exhibit significantly faster convergence speeds, achieve larger effective sample sizes, and produce lower mean square estimation errors with the same computational budget.

Contents

Chapter 1

Introduction

The estimation of an unknown image from noisy and incomplete data has been widely studied in the research community during the last century. Problems in this area cover several topics, such as image denoising [115, 52], deblurring [83, 31], compressive sensing reconstruction [7, 17], super-resolution [90, 116], tomographic reconstruction [73, 130] and inpainting [63, 51], to name a few. A common issue in these problems is that estimating a solution directly from the data is difficult since many of the imaging problems are ill-posed or ill-conditioned. Currently, there are three main strategies to address this difficulty:

- The variational framework [30], whose goal is to minimise a data misfit functional together with an appropriate chosen regularisation term.
- The machine learning approach [6] which seeks to exploit deep learning techniques to solve inverse problems by learning a mapping function from a training data set.
- The Bayesian statistical framework [102], which attempts to recover the full probability distribution of the solution to calculate not only estimators but also to measure their uncertainty.

In this thesis, we focus on the Bayesian paradigm, which is especially well suited to address imaging problems where uncertainty is a significant factor. For instance, in medical imaging, it is necessary or desirable to quantify the uncertainty in the delivered solutions to inform decisions or conclusions, see, e.g., [15, 139]. The framework is also well suited to blind, semi-blind, and unsupervised problems involving partially unknown models (e.g., unspecified regularisation parameters or observation operators) [136]. Bayesian model selection techniques also allow the objective comparison of several potential models to analyse the observed imaging data, even in cases where there is no ground truth available [45, Section 4.1].

The Bayesian framework allows *a priori* information to be incorporated into the target probability distribution in the form of a prior distribution that also acts as a regulariser, playing a key role in the well-posedness of the problem. Several different distributions have been proposed as regularisers and have been studied in the literature in recent decades [11], including:

- Markov random fields [77, Section 3.3.4], which are able to model neighbour pixel dependence. Popular approaches are based on Total-variation (TV) [105, 29] and total generalized variation (TGV) [21], which are designed to denoise flat regions while preserving edges at the same time.
- Sparsity-promoting priors [27, 120], such as the Laplace distribution [93] which are suitable for sparse images i.e., images where some (or most) of their pixels are zero. Applications include sparsity w.r.t. a set of column vectors or *atoms* placed in a matrix which is normally referred to as a *dictionary* [38].
- Data-driven priors [137], which have become a popular approach, have shown in most cases to outperform the traditional functional priors, examples of these are generative priors based on adversarial network and variational autoencoder models [61, 72] and deep image priors [121].

- Gaussian mixture model priors [135, 138], which enjoy the flexibility of assuming that the image is generated by a mixture of tractable probability distributions and allows the computation of point estimates in a closed-form expression for Gaussian observation models. Examples include patch-based models [24, 131] which have been demonstrated to preserve local information.

In this thesis, we are interested in log-concave priors which have an underlined convex geometry that allows the application of efficient Bayesian methods based on optimisation and sampling [101, 98].

The Bayesian framework offers a lot of flexibility in terms of the different quantities of interest that can be used for image reconstruction. In particular:

- The first approach is to formulate a convex model in which the computation of the maximum a-posteriori (MAP) estimate can be performed efficiently by using convex optimization algorithms [101, 26]. However, it cannot tackle more complex Bayesian analyses that go beyond point estimates.

- A second approach is based on Monte Carlo methods and, in particular, Markov chain Monte Carlo (MCMC) [103, 5], its main objective is to approximate the computation of important integrals via the generation of an ergodic Markov chain with invariant distribution.

- A third approach to performing Bayesian inference consists of an approximation of the target distribution based on the application of deterministic techniques [113], such as variational Bayes [54, 16]. Although this approach may be computationally faster than the stochastic simulations performed by MCMC, variational Bayes does not provide convergence guarantees to the target distribution, while MCMC enjoys such guarantees of producing (asymptotically) exact samples from the target density.

This thesis will mainly focus on providing some advances in the computational aspects of MCMC methods to solve Bayesian imaging inverse problems and, in particular, gradient-based MCMC methods applied to log-concave models. Regarding this class of methods, two main challenges can be found in designing efficient gradient-based MCMC algorithms to perform robust Bayesian inference in imaging: the high dimensionality of the problem and the lack of differentiability of the target density. A first attempt to address these problems is the proposal of proximal MCMC methods [95], such as the so-called Moreau Yosida unadjusted Langevin algorithm (MYULA) [45] that addresses the issue of non-differentiability by borrowing ideas from the field of non-smooth convex optimization.

Although these algorithms enjoy good theoretical convergence guarantees, they can become computationally expensive for very ill-conditioned problems, since the corresponding step-size restriction results in very slow convergence to steady-state. In this thesis, we tackle this difficulty by first proposing an orthogonal Runge-Kutta-Chebyshev stochastic approximation of the Langevin diffusion process called SK-ROCK [2] that is significantly more computationally efficient than the conventional Euler-Maruyama approximation used by existing proximal MCMC methods. In particular, we present a new method that applies this approximation to the Moreau-Yosida regularised Langevin diffusion underpinning the MYULA [45], and show in a class of Gaussian models and numerical experiments that this leads to dramatic improvements in convergence speed and estimation accuracy.

A separate line of research for dealing with the step-size restriction proposes the use of an *augmented* model in conjunction with a Gibbs sampling scheme named the Split Gibbs Sampler (SGS) [125, 129], which also allows a larger step-size to be used by splitting the sampling process and relaxing the dependence of a large Lipschitz constant in poor conditioning problems. Regarding the latter, we formally identified a relationship between SGS and MYULA, which allows us to propose two novel MCMC methods, the first being an improved version of SGS and the second a clever fusion between SGS and SK-ROCK. Furthermore, we revisited the augmented model of [100, 125, 129] and, by adjusting the methodology in [123], we propose an empirical Bayesian method to estimate the hyperparameters of this augmented model, which allows us to propose an enhanced class of models for Bayesian inference in imaging inverse problems that can deliver more accurate estimates than the non-augmented model.

1.1 Structure of the thesis

The structure of this thesis is as follows: Chapter 2 introduces the class of imaging problems considered in this thesis, Chapter 3 provides a summary of some of the most important exact and inexact MCMC methods. In Chapter 4, we introduce the proximal SK-ROCK method and study it from both a theoretical and numerical point of view. In Chapter 5, we present two new MCMC methods based on the SGS algorithm and present the augmented model as an enhanced posterior distribution for Bayesian inference in imaging applications. Conclusions and perspectives for future work are reported in Chapter 6.

Chapter 2

Bayesian inference in imaging inverse problems

We seek to estimate an unknown image $x \in \mathbb{R}^d$ from observed data y, related to x by a forward model of the following form

$$y = \Phi x + \eta \tag{2.1}$$

where Φ is a linear operator that represents the physical properties of the image acquisition process. In real applications, we also need to consider the sources of error in data acquisition, this situation is represented by η, which denotes the observation noise. In this setting, Hadamard [65] postulated three conditions in which problems of the form (2.1) are called well-posed:

- The solution exists.
- The solution is unique.
- The solution depends continuously on the data, i.e., a small variation in y results in small errors in the solution.

Problems that do not meet one of these conditions are called ill-posed and one may find that inverse problems generally do not satisfy at least one of the above conditions. Furthermore, there are also inverse problems that admit a unique solution, but are highly sensitive and not stable w.r.t. small perturbations, these problems are called ill-conditioned.

2.1 Model problems

To better illustrate the issues of ill-posedness and ill-conditioning that we described above, we present below the particular inverse problems in imaging that we are going to deal with in this thesis.

2.1.1 Image Deblurring

The deblurring or deconvolution problem of recovering a signal $x(t)$ from observed data $y(t)$ follows the equation below

$$y(t) = \int_{-\infty}^{\infty} k(t,s)x(s)ds + \eta(t), \tag{2.2}$$

where $k(x, y)$ is assumed to be spatially invariant, i.e.,

$$k(x, y) = k(x - y),$$

and known as the convolution kernel. In the noiseless case, i.e.,

$$y_\star(t) = \int_{-\infty}^{\infty} k(t,s)x(s)ds,$$

one applies the Fourier transform, it yields

$$\hat{y}_\star(\xi) = \int_{-\infty}^{\infty} e^{-it\xi} y_\star(t) dt,$$

and by the convolution theorem, one has that $\hat{y}_\star(\xi) = \hat{k}(\xi)\hat{x}(\xi)$. Therefore, by the inverse Fourier transform

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-it\xi} \frac{\hat{y}_\star(\xi)}{\hat{k}(\xi)} d\xi.$$

In the noisy case, one can show that

$$\tilde{x}(t) = x(t) + \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-it\xi} \frac{\hat{\eta}(\xi)}{\hat{k}(\xi)} d\xi,$$

however, the inverse of $\hat{k}$ usually decreases exponentially, making $\tilde{x}(t)$ very sensitive to small changes produced by noise. In practice, a finite-dimensional representation of (2.2) is employed. After discretisation of the continuous model, the deconvolution model can be expressed as

$$y = \mathrm{A}x + \eta,$$

where $\mathrm{A} \in \mathbb{R}^{d \times d}$ is a blur operator and $\eta \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_d)$. Figure 2.1(a) presents the `cameraman` test image of size $d = 256 \times 256$ pixels, which we will denote x. Figure 2.1(b) shows the artificially blurry and noisy observation y, generated by applying the *box* blur or *box* linear filter A [88], which performs average smoothing on each of the image pixels x using a 5×5 uniform blur operator, given by the matrix $1/25 J_5$ (where J_n is an $n \times n$ matrix of ones) and then adding Gaussian noise with $\sigma = 0.58$, related to a signal-to-noise ratio of 40dB. Figure 2.1(c) shows the absolute value of the real part of the eigenvalues of A in the Fourier domain and, as can be seen, there is a number of eigenvalues that are close to 10^{-5}. Because the eigenvalues of A decrease exponentially (as we mentioned before in the continuous case), this yields highly noise-sensitive solutions, making the problem ill-conditioned.

2.1.2 Image Inpainting

This experiment consists of randomly selecting a percentage of the image pixels $x \in \mathbb{R}^d$ contaminated with Gaussian noise with an SNR level of 40dB, to form the observation vector $y \in \mathbb{R}^m$ (note that $m < d$). This can be represented by the linear model

$$y = \mathrm{A}x + \eta,$$

where $\mathrm{A} \in \mathbb{R}^{m \times d}$ is a binary matrix formed by a subset of rows of the d-dimensional identity matrix and $\eta \in \mathbb{R}^m$ is noise. Figure 2.2 presents the noisy and incomplete observation y after applying the rectangular operator A to the same `cameraman` test image from the previous example and adding Gaussian noise. In this case, 40% of the pixels were randomly deleted. We clearly have in this inverse problem an underdetermined system that can produce an infinity of solutions, therefore violating one of the Hadamard conditions.

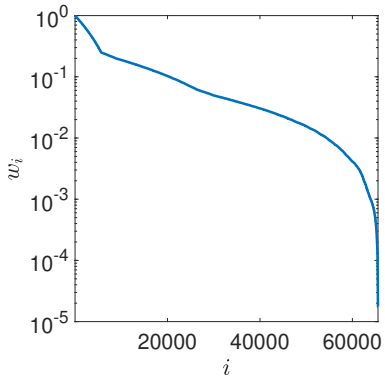
2.1.3 Magnetic Resonance Imaging (MRI)

This is a non-invasive and non-ionising medical imaging technique that, through a phenomenon called magnetic resonance, allows the sequential measurement of the Fourier coefficients of the image of interest. However, the process of acquiring an MRI can be long, uncomfortable and expensive for the patient. This is why incomplete Fourier data acquisition is commonly used as a way to speed up the imaging process.

In tomographic image reconstruction, we seek to recover an image $x \in \mathbb{R}^d$ from an observation $y \in \mathbb{C}^m$ related to x by a linear Fourier model

$$y = \mathrm{A}x + \eta,$$

(a) true image x (b) observation y

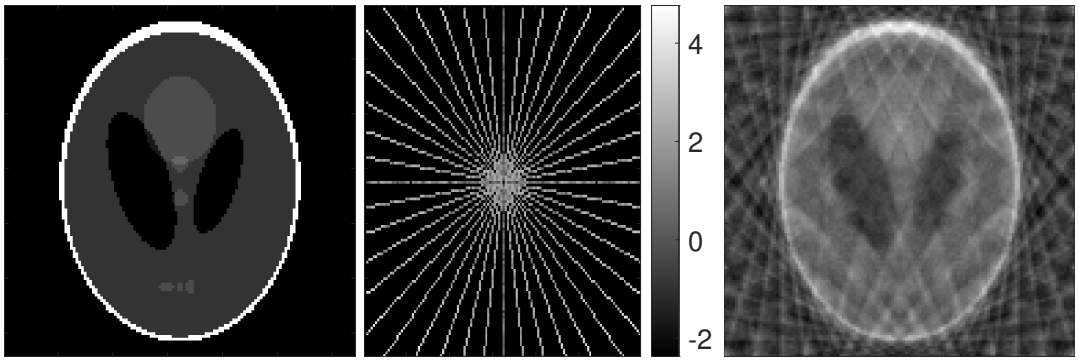


(c) eigenvalues of A

Figure 2.1: `Cameraman` deblurring experiment: (a) Original image of dimension 256×256 pixels; (b) real part of eigenvalues of A (in absolute value). (c) Blurred observation with SNR= 40.

Figure 2.2: `Cameraman` inpainting experiment: Noisy and incomplete observation y.

where A = HF, F is the discrete Fourier transform operator on $\mathbb{C}^d$, H $\in \mathbb{C}^{m\times d}$ is a (sparse) tomographic subsampling mask and $\eta \sim N(0, \sigma^2 \mathbb{I}_{2m})$. Typically $m \ll d$, i.e., a very small number of noisy Fourier measurements are available, which makes the estimation problem strongly ill-posed (due to the non-uniqueness of the solution), resulting in significant uncertainty about the true value of x. Figure 2.3 presents this experiment with the `Shepp-Logan phantom` test image of dimension 128×128, depicted in Figure 2.3(a), which we use to generate a noisy observation y by measuring 15% of the original Fourier coefficients, corrupted with additive Gaussian noise with $\sigma = 10^{-2}$. To improve visibility, Figure 2.3(b) shows the amplitude of the Fourier coefficients on a logarithmic scale, where unobserved coefficients are depicted in black. Figure 2.3(c) shows the so-called "back-projected image" which is the projection of the observation y onto the image domain.



(a) true image x (b) observation y (c) back-projected image $\tilde{y}$

Figure 2.3: `Tomography` experiment: (a) `Shepp-Logan phantom` image (128×128 pixels); (b) tomographic observation y (amplitude of Fourier coefficients in logarithmic scale). (c) Back-projected image $\hat{y}$.

2.1.4 Hyperspectral unmixing

The hyperspectral acquisition technique consists of capturing narrowband spectral images in different frequency bands. This dense set of images is then stored in a three-dimensional hyperspectral data cube for analysis. Despite the high number of spectral bands, the spacial resolution is usually low, causing in most cases an undesired mixed spectral effect [111]. This gives rise to the hyperspectral unmixing problem [14, 82], which aims to separate the mixed pixels into their corresponding spectra and fractional abundances or proportions.

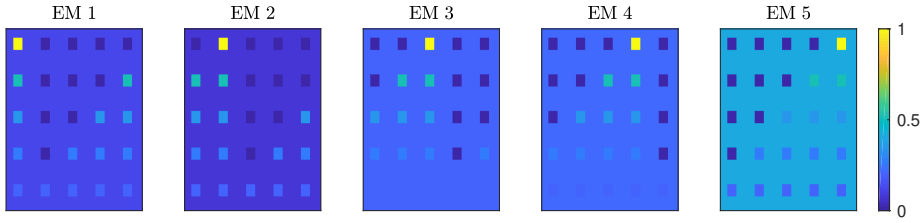
To obtain the latter, a linear model is assumed. In particular, given a hyperspectral image $y \in \mathbb{R}^{m\times d}$ with m spectral bands and d pixels, the unmixing problem assumes that the observed scene is composed of k materials or *endmembers*, each with a characteristic spectral response $a_j \in \mathbb{R}^m$ for $j \in \{1, \dots, k\}$, and seeks to determine the proportions or abundances $x_{j,i}$ of each material $j \in \{1, \dots, k\}$ in each image pixel $i \in \{1, \dots, d\}$. Here we consider the widely used linear mixing model

$$y = \mathrm{A}x + \eta,$$

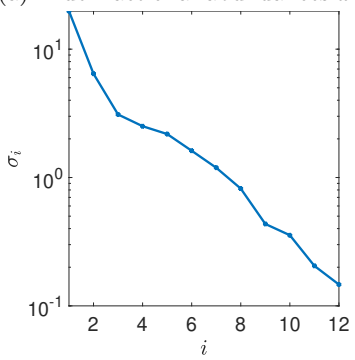
where $\mathrm{A} = \{a_1, \dots, a_k\} \in \mathbb{R}^{m\times k}$ is a spectral library gathering the spectral responses of the materials, $x \in \mathbb{R}^{k\times d}$ gathers the abundances, and $\eta \sim \mathbb{N}(0, \sigma^2 \mathbb{I}_{m\times d})$ is additive Gaussian noise.

In general, this is a challenging ill-posed inverse problem due to the high correlation of the spectral signatures of the materials in the spectral library A [74, 48]. To illustrate this situation, Figure 2.4 shows the *Simulated Data Cube 1* synthetic experiment from [75, Section 4], in which we consider a simplified version of the spectral signature library A $\in \mathbb{R}^{224\times 12}$ with 224 frequency bands, selecting 12 random materials out of 240 originally presented in the library[1]. The image x has dimension $75 \times 75 = 5625$ and out of the 12 materials, only five are present in the synthetic image. Figure 2.4(a) shows the true fractional abundance of these five materials and Figure 2.4(b) shows the rapid decay of the singular values of A.

[1] Available online from `http://speclab.cr.usgs.gov/spectral.lib06`.



(a) True fractional abundances x



(b) Singular values of A

Figure 2.4: Hyperspectral unmixing experiment: (a) True fractional abundances of endmembers 1 to 5 (from left to right), and (b) singular values of the spectral library A.

2.2 The need for regularisation

A common way to accurately estimate x from y and achieve a unique and stable solution to these problems is using a deterministic approach in the field of regularisation theory [11]. For instance, a basic regularisation scheme comes from Tikhonov regularisation [119, 118], in which one is interested in solve

$$\min_{x\in\mathbb{R}^d} \|\mathrm{A}x - y\|^2 + \beta\|x\|^2, \tag{2.3}$$

where a regularisation term $\|x\|^2$ is introduced, and $\beta > 0$ is commonly defined as the regularisation parameter, which balance the effect between the data fidelity term $\|\mathrm{A}x - y\|^2$ and the regularisation term $\|x\|^2$. It can be shown that the solution of (2.3) exists, is unique, stable [87] and given by

$$x^*_\beta = (\mathrm{A}^\mathsf{T}\mathrm{A} + \beta\mathbb{I}_d)^{-1}\mathrm{A}^\mathsf{T}y. \tag{2.4}$$

In Figure 2.5, we present the results after applying Tikhonov regularisation to the image deblurring experiment described in Section 2.1.1.

Regularisation methods have been widely used in recent decades and include approaches such as approximate analytic inversion [110], iterative methods with early stopping [23], discretisation as regularisation [78, 67] and variational methods [109], including Tikhonov regularisation. For a summary of regularisation methods, see e.g., [6, Section 2] and [77, Section 2].

2.3 The Bayesian approach

Another direction that renders the problem well-posed, and it is the approach we will utilise in this thesis, is to consider the Bayesian inversion theory [112], a statistical approach that seeks to recover the probability distribution of the unknown image x given the observed data y by modelling all variables of interest (x, y and noise) as random ones. In this setting, one first seeks to characterise the degree of knowledge of an image of interest x through a probability distribution that relates the available observation y with x via a likelihood distribution. In this thesis, we will assume this distribution has a probability density function (up to a normalisation

(a) $x^*_{0.1}$ (b) $x^*_{0.01}$ (c) $x^*_{0.001}$

Figure 2.5: Image deblurring experiment: results after applying the Tikhonov regularisation (2.4), using different values for the regularisation parameter β: (a) $\beta = 10^{-1}$, (b) $\beta = 10^{-2}$, (c) $\beta = 10^{-3}$.

constant) of the following form

$$p(y|x) \propto \mathrm{e}^{-f_y(x)}, \tag{2.5}$$

where $f_y(x) : \mathbb{R}^d \to \mathbb{R}$ is convex and Lipschitz continuously differentiable with constant L_f, that is,

$$\forall x_1, x_2 \in \mathbb{R}^d, \|\nabla f_y(x_1) - \nabla f_y(x_2)\| \leq L_f \|x_1 - x_2\|.$$

As we illustrated at the beginning of this section, the imaging experiments considered in this thesis has the linear form $y = \mathrm{A}x + \eta$, where $\eta \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_m)$. In this case, the likelihood distribution can be written as

$$p(y|x) \propto \exp\left(-\frac{\|y - \mathrm{A}x\|^2}{2\sigma^2}\right).$$

The Bayesian approach allows to regularise the estimation problem by incorporating the available knowledge of the unknown image x through the so-called prior distribution. In this thesis, we will assume that these distributions will have the following form

$$p(x|\beta) \propto \mathrm{e}^{-\beta^\intercal g(x)}, \tag{2.6}$$

where $\beta \in (0, +\infty)^p$ is a vector of hyperparameters that controls the amount of enforced regularity, $g : \mathbb{R}^d \to \mathbb{R}^p$ is a vector of statistics that includes all the prior information one wants to include, and each $(g_i)_{i \in \{1,\dots,p\}} : \mathbb{R}^d \to (-\infty, \infty]$ is proper, convex, lower semicontinuous, and potentially non-smooth.

Regarding the prior models that have been successfully used in inverse problems, we will now provide a brief explanation of the prior distributions that we will use for the problems previously illustrated in this section.

2.3.1 Types of prior

As we mentioned before, the priors that interest us in this thesis are those that enjoy log-concavity, which allows the application of efficient optimisation and sampling techniques. The ones we will use in this thesis are explained below.

ℓ^1 Prior

If the prior information one has from the image x seeking to estimate is that it contains small and well-localised objects (for example, a tumour in an MRI scan of a brain), then impulse prior models are the preferred approach. They are mainly applied to images with low average amplitude with few outliers. A typical impulse prior to consider is the ℓ^1 prior or Laplace

distribution [28, 117], which is defined as

$$p(x|\beta) \propto \exp(-\beta\|x\|_1)$$

Total variation Prior

There are some situations in which the image one wants to find contains discontinuities, i.e., it can have large jumps every now and then. The total variation density is useful in these scenarios [106]. Let $f : \Omega \to \mathbb{R}$ be a function in $L^1(\Omega)$, i.e., the space of integrable functions on $\Omega \subset \mathbb{R}^2$, the total variation of f is defined as

$$\mathrm{TV}(f) = \sup\left\{\int_\Omega f\nabla \cdot h\, dx \,:\, h \in C_c^1(\Omega, \mathbb{R}^2), \sqrt{h_1^2 + h_2^2} \leq 1\right\}.$$

When f is smooth, the total variation of x can be defined as

$$\mathrm{TV}(f) = \int_\Omega \|\nabla f\| dx.$$

For numerical implementations, several total-variation seminorms have been proposed [29, 32]. In particular, if we assume that the image x can be represented as a two-dimensional matrix, i.e., $x \in \mathbb{R}^{N\times N}$ where $N \times N = d$, the isotropic seminorm is given by

$$\mathrm{TV}_{\mathrm{iso}}(x) = \sum_{i,j} \sqrt{|x_{i+1,j} - x_{i,j}|^2 + |x_{i,j+1} - x_{i,j}|^2},$$

and the anisotropic version

$$\begin{aligned}\mathrm{TV}_{\mathrm{aniso}}(x) &= \sum_{i,j} \sqrt{|x_{i+1,j} - x_{i,j}|^2} + \sqrt{|x_{i,j+1} - x_{i,j}|^2} \\ &= \sum_{i,j} |x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|,\end{aligned}$$

for $i, j = 1, \ldots, N$. In this thesis we use the isotropic seminorm, therefore, the discrete total variation density is then given by

$$p(x|\beta) \propto \exp(-\beta\mathrm{TV}_{\mathrm{iso}}(x)).$$

From now on we refer to $\mathrm{TV}_{\mathrm{iso}}(x)$ as simply $\mathrm{TV}(x)$. These are the priors we are going to use in our experiments, i.e., in the imaging inverse problems previously illustrated.

2.3.2 Bayes' theorem and the posterior distribution

Having defined the prior and likelihood distributions we will work with, we can use Bayes' theorem, to derive the posterior distribution $p(x|y, \beta)$, given by

$$p(x|y, \beta) = \frac{p(y|x)p(x|\beta)}{\int_{\mathbb{R}^d} p(y|x)p(x|\beta)\mathrm{d}x}.$$

Our main focus is the computational aspects of calculating estimates from posterior distributions which, taking into account (2.5) and (2.6), will have the following form

$$p(x|y, \beta) \propto \exp[-f_y(x) - \beta^{\mathsf{T}} g(x)]. \tag{2.7}$$

2.3.3 The different approaches in the Bayesian paradigm

Under the Bayesian framework, the typical approach in the imaging community is to exploit the convexity properties of (2.7) and compute the so-called maximum a posteriori (MAP) point

estimate [96], which can be formulated as the following optimisation problem

$$x_{\text{MAP}} = \underset{x}{\operatorname{argmax}}\, p(x|y,\beta) = \underset{x}{\operatorname{argmin}}\, f_y(x) + \beta^{\mathsf{T}} g(x),$$

and can be computed with efficient convex optimisation algorithms that exploit the log-concavity of (2.7) and are able to deliver accurate solutions in reasonable computational times [30]. However, to perform more complex Bayesian analysis techniques such as Bayesian model selection, model calibration and hypothesis testing [102], this commonly requires the computation of high dimensional integrals of the form

$$\bar{h} = \int_{\mathbb{R}^d} h(x) p(x|y,\beta) \mathrm{d}x, \tag{2.8}$$

which are typically intractable.

Several approaches have been proposed to compute numerically (2.8), such as variational Bayesian inference [16], which approximates $p(x|y,\beta)$ by selecting from a family of tractable distributions the one that best fits the target distribution by the application of optimization techniques. Despite being computationally fast, variational inference has weaker convergence guarantees to the exact posterior distribution.

A technique that does possess more robust convergence properties is Monte Carlo integration and, in particular, Markov chain Monte Carlo (MCMC) methods [103, 22]. MCMC methods are used to generate correlated samples from target distributions that are not amenable to exact sampling. The idea is to construct a time-homogeneous Markov chain from an irreducible and aperiodic transition kernel K with invariant probability distribution μ that admits a density $\pi(x)$, and generates samples $x_1, x_2, \ldots$ using the kernel K (see [107] for details).

In this case, the corresponding Markov chain $x_1, x_2, \ldots$ is ergodic, and the following relationship holds

$$\mathbb{P}\left(\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} h(x_i) = \int_{\mathbb{R}^d} h(x)\pi(x)\mathrm{d}x \right) = 1.$$

This implies that we can use the samples $x_1, x_2, \ldots$ to approximate integrals (i.e., expectations w.r.t. the invariant distribution) in the same way as we use i.i.d. samples from classical Monte Carlo methods.

Chapter 3

Markov-chain Monte Carlo for Bayesian inference

As we mentioned in the previous chapter, Markov chain Monte Carlo (MCMC) methods deal with the problem of drawing extract samples from $p(x|y,\beta)$ by generating correlated samples that satisfy the ergodicity property so that they can be used to compute (2.8) with convergence guarantees similar to those of classical Monte Carlo methods.

For several years, the main focus on MCMC methods has been on producing samples via asymptotically exact MCMC methods [46, 55, 132, 12], that is, samples of the exact target distribution μ. Although this is an ideal direction to calculate estimators in a very precise way, it implies an intermediate Metropolis-Hastings (MH) correction step at each iteration that deteriorates the performance of these exact sampling methods when the problem is defined in very high dimensions, or the step-size of these methods must be adjusted to be very small to have a desirable acceptance rate [55].

The latter has increased the interest in research on inexact MCMC methods [44, 33, 134, 108, 91, 34]. In this case, due to the lack of the MH correction step, the Markov chain targets a different distribution that is normally close to the true target distribution, however, these methods have demonstrated to produce estimators in more efficient computational times, paying for a small amount of error or bias that can be controlled adapting the values of algorithmic parameters, such as the step-size. In this chapter, we briefly discuss both approaches, reviewing some of the most important exact and inexact MCMC methods.

3.1 Exact MCMC Methods

3.1.1 Metropolis-Hastings algorithm

This MCMC algorithm was presented in 1953 for the case of symmetrical proposal distributions [89] and then extended for a more general case in 1970 [68]. It can be seen as the general framework on which the MCMC methods we will describe in this section are based. The main idea is to generate samples from a proposal distribution $q(x,x')$ that differs from the true target density $\pi(x)$ but is easier to generate samples from. The application of an appropriately chosen MH step ensures that the target distribution $\pi(x)$ is invariant for the corresponding Markov chain, and so the samples are distributed according to $\pi(x)$ when the chain reaches stationarity[1][103, Section 6.6]. This method is called the MH algorithm and is illustrated in Algorithm 1.

3.1.2 Random-Walk Metropolis algorithm

One of the most common and easy-to-implement MH algorithms of the last decades is called the Random-walk Metropolis (RWM) algorithm [89, 68]. This method consists on construct a

[1]It is important to emphasize that the stationary distribution is also a limiting distribution in the sense that x_n is distributed according to $\pi(x)$ when $n \to \infty$, that is, x_n is *asymptotically* distributed according to $\pi(x)$.

Algorithm 1 MH Algorithm

Require: Initial sample x_0, no. samples N, target $\pi(x)$, proposal $q(x, x')$
  for $n = 0 : N - 1$ **do**
    Draw $x'_{\text{prop}} \sim q(x_n, x')$
    Calculate

$$\alpha(x_n, x'_{\text{prop}}) = \min\left(1, \frac{\pi(x'_{\text{prop}})q(x'_{\text{prop}}, x_n)}{\pi(x_n)q(x_n, x'_{\text{prop}})}\right)$$

    Draw $u \sim \mathcal{U}([0, 1])$
    if $u < \alpha(x_n, x'_{\text{prop}})$ **then**
      Accept: **Set** $x_{n+1} = x'_{\text{prop}}$
    else
      Reject: **Set** $x_{n+1} = x_n$
    end if
  end for

proposal distribution based on a random walk model $q(x, x') = h(x - x')$ where $h : \mathbb{R}^d \to \mathbb{R}_+$. This scheme is illustrated in Algorithm 2, in which $q(x, x') \propto \exp(-0.5\delta^{-2}(x - x')^{\mathsf{T}}(x - x'))$, this is a typical proposal choice since $q(x, x') = q(x', x)$, allowing a more simplified structure of the acceptance probability, i.e., the proposal distribution does not appear in $\alpha(x, x')$.

Algorithm 2 Random-Walk Metropolis Algorithm

Require: Initial sample x_0, no. samples N, target $\pi(x)$, step-size δ
  for $n = 0 : N - 1$ **do**
    Draw $\xi \sim \mathcal{N}(0, \delta^2\mathbb{I}_d)$
    Draw $x'_{\text{prop}} = x_n + \xi$
    Calculate

$$\alpha(x_n, x'_{\text{prop}}) = \min\left(1, \frac{\pi(x'_{\text{prop}})}{\pi(x_n)}\right)$$

    Draw $u \sim \mathcal{U}([0, 1])$
    if $u < \alpha(x_n, x'_{\text{prop}})$ **then**
      Accept: **Set** $x_{n+1} = x'_{\text{prop}}$
    else
      Reject: **Set** $x_{n+1} = x_n$
    end if
  end for

Notice that the acceptance ratio $\alpha(x, x')$ presents a simplified form compared to Algorithm 1, thanks to the symmetry of q. The step-size δ in the latter algorithm is also called the *scaling parameter*, as it is well known that, when it is sufficiently small, the algorithm will accept the majority of proposed samples, but the progress of the chain will be slow; on the other hand, when δ is very large, the algorithm will reject the majority of samples, also affecting the performance of the algorithm.

Despite its simplicity and advances in its theoretical analysis, RWM remains a computationally expensive algorithm in high dimensions, mainly due to the lack of information about the target distribution in the proposed samples, making its use impractical for imaging applications. This motivated the research community to study and propose more sophisticated sampling schemes that enhance the proposal distribution and thus improve the sampling process. The Metropolis-adjusted Langevin algorithm (MALA) and the Hamiltonian Monte Carlo (HMC) algorithm are two examples of more sophisticated schemes that scale better in higher dimensions, as we will see later.

3.1.3 Metropolis-Adjusted Langevin Algorithm

An obvious disadvantage of the RMW algorithm is that it does not contain any information about the target density π in the proposed samples x'_{prop}. One natural way to overcome the

latter is to add some information on the target distribution in the proposal in such a way that enforces to move to areas of large probability. This can be achieved by incorporating the gradient of $\log \pi$ to the proposal distribution, i.e.,

$$q(x, x') \propto \exp\left(-\frac{1}{4\delta}\|x' - x - \delta\nabla \log \pi(x)\|^2\right). \tag{3.1}$$

If we incorporate this proposal into Algorithm 1, the resulting sampling method is the so-called Metropolis adjusted Langevin Algorithm (MALA), illustrated in Algorithm 3.

Algorithm 3 Metropolis adjusted Langevin Algorithm

Require: Initial sample x_0, no. of samples N, step-size δ, functions $\pi(x)$ and $\nabla \log \pi(x)$.
  for $n = 0 : N - 1$ **do**
    Draw $\xi \sim \mathcal{N}(0, \mathbb{I}_d)$
    Set $x'_{\text{prop}} = x_n + \delta\nabla \log \pi(x_n) + \sqrt{2\delta}\xi$
    Calculate

$$\alpha(x_n, x'_{\text{prop}}) = \min\left(1, \frac{\pi(x'_{\text{prop}})q(x'_{\text{prop}}, x_n)}{\pi(x_n)q(x_n, x'_{\text{prop}})}\right)$$

    Draw $u \sim \mathcal{U}([0, 1])$
    if $u < \alpha(x_n, x'_{\text{prop}})$ **then**
      Accept: **Set** $x_{n+1} = x'_{\text{prop}}$
    else
      Reject: **Set** $x_{n+1} = x_n$
    end if
  end for

Note that $q(x, x')$ is not symmetric as in Algorithm 2 and therefore $q(x, x')$ needs to be taken into account in the MH step. In addition, the choice of the proposal distribution (3.1) comes from the context of stochastic differential equations (SDEs), which we will discuss in Section 3.2.1.

3.1.4 Hamiltonian Monte Carlo

The Hamiltonian Monte Carlo (HMC) method was first introduced more than seventy years ago in the area of lattice field theory [40]. Since then, several works have been presented that develop the theory, applications and variants of this MCMC method [76, 71, 19, 13, 18]. The idea behind this successful algorithm is to simulate the Hamiltonian dynamics

$$\frac{\mathrm{d}q_i}{\mathrm{d}t} = \frac{\partial H}{\partial p_i} \qquad \text{and} \qquad \frac{\mathrm{d}p_i}{\mathrm{d}t} = -\frac{\partial H}{\partial q_i}, \tag{3.2}$$

that is, the evolution of the position q and momentum p of a particle during some time t, with Hamiltonian function defined as

$$H(q, p) = U(q) + K(p), \tag{3.3}$$

where $U(x)$ is called the potential energy and will be defined as $-\log \pi(x)$, i.e., the log of the density of interest, and $K(x)$ is called the kinetic energy which is usually defined as

$$K(p) = p^\intercal M^{-1} p/2, \tag{3.4}$$

but other kinetic energy functions can be used [12, Section 4.2]. The idea behind HMC is that the so-called canonical density, defined as $\exp(-H(q, p))$ is preserved by the Hamiltonian dynamics due to the reversibility, conservation of energy and volume-preserving properties that this Hamiltonian dynamics enjoy (see [107, Theorem 7]). Moreover, from this canonical density, one can notice that the stochastic vectors q and p are independent, which allows defining a Markov chain in the following way

- Draw $p_n \sim \mathcal{N}(0, \mathrm{M})$.

- Find (q_{n+1}, p_{n+1}) by solving the Hamiltonian dynamics (3.2) with initial condition (q_n, p_n) and Hamiltonian function (3.3) up to some time T.

By using this recursion, the resulting Markov chain q_n, q_{n+1} (resp., x_n, x_{n+1}) has $\pi(q)$ (resp., $\pi(x)$) as its invariant density and is reversible w.r.t. π [107, Theorem 8].

However, in most applications it is impossible to exactly solve (3.2) therefore, a numerical scheme needs to be considered. A reversible algorithm that simulates this dynamics and preserves volume is known as the *leapfrog* or Störmer-Verlet method [66]. Given an step-size τ, a step performed by the leapfrog method from (q_n, p_n) to (q_{n+1}, p_{n+1}) is defined as

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{\tau}{2}\nabla \log \pi(q_n), \\ q_{n+1} &= q_n + \tau M^{-1} p_{n+1/2}, \\ p_{n+1} &= p_{n+1/2} - \frac{\tau}{2}\nabla \log \pi(q_{n+1}). \end{aligned}$$

Since it does not preserve energy, one needs to add an MH step in order to leave the target distribution invariant. With all these ingredients, we now have all of the necessary information to present the HMC method, which is illustrated in Algorithm 4.

Algorithm 4 Hamiltonian Monte Carlo Algorithm

Require: Initial value x_0, no. of samples N, no. of leapfrog iterations L, step-size δ, functions $\pi(x)$ and $\nabla \log \pi(x)$.
Set $q_0^0 = x_0$
for $n = 0 : N-1$ **do**
  Draw $p_n^0 \sim \mathcal{N}(0, \mathrm{M})$
  for $m = 0 : L-1$ **do**
    Set $p_n^{m+1/2} = p_n^m - 0.5\delta \nabla \log \pi(q_n^m)$
    Set $q_n^{m+1} = q_n^m + \delta M^{-1} p_n^{m+1/2}$
    Set $p_n^{m+1} = p_n^{m+1/2} - 0.5\delta \nabla \log \pi(q_n^{m+1})$
  end for
  Calculate

$$\alpha(x, y) = \min\left\{1, \frac{\exp(-H(q_n^L, p_n^L))}{\exp(-H(q_n^0, p_n^0))}\right\}$$

  Draw $u \sim \mathcal{U}([0,1])$
  if $u < \alpha(x, y)$ **then**
    Accept: **Set** $x_{n+1} = q_n^L$, $q_{n+1}^0 = q_n^L$
  else
    Reject: **Set** $x_{n+1} = q_n^0$, $q_{n+1}^0 = q_n^0$
  end if
end for

3.2 Inexact MCMC Methods

The exact MCMC techniques mentioned above include an MH correction step at every iteration, which can slow down the progress of the chain, particularly in high-dimensional applications like imaging. This last aspect motivated the study of inexact MCMC methods that, despite not targeting the true distribution and having a slightly higher asymptotic bias, have faster convergence, lower initialization bias and lower estimation variance. Compared to exact MCMC methods, this enables the generation of higher quality samples in a shorter computational time. We will discuss in this section the inexact MCMC methods used in this thesis.

3.2.1 Unadjusted Langevin Algorithm

Recall from Section 3.1.3 that the proposal distribution (3.1) can be explained from the theory of stochastic differential equations (SDEs). In this context, the (overdamped) Langevin diffusion,

characterised by the following SDE

$$\mathrm{d}X_t = \nabla \log \pi(X_t)\mathrm{d}t + \sqrt{2}\mathrm{d}W_t\,, \tag{3.5}$$

where W_t is a d-dimensional Brownian motion, has a unique strong solution and admits $\pi(x)$ as its unique invariant density, under mild assumptions on $\log \pi(x)$ [104, Theorem 2.1]. However, it is usually not possible to solve (3.5) exactly, so a numerical approximation needs to be taken into consideration. The simplest possible way to do so is to consider the Euler-Maruyama (EM) scheme given by

$$X_{n+1} = X_n + \delta \nabla \log \pi(x) + \sqrt{2\delta} Z_{n+1}, \tag{3.6}$$

where $\delta > 0$ is a given step-size and $(Z_{n+1})_{n\geq 0}$ is an i.i.d. sequence of d-dimensional standard Gaussian random vectors. This numerical scheme targets an approximated distribution $\pi_\delta(x)$ that is close to $\pi(x)$ [104, Section 1.4.1]. If we add a MH correction to target π, we have Algorithm 3. Without any MH correction step, the resulting scheme is called unadjusted Langevin algorithm (ULA) [44, 43] or Langevin Monte Carlo (LMC) [33, 42], illustrated in Algorithm 5.

Algorithm 5 unadjusted Langevin Algorithm

Require: Initial value x_1, no. of samples N, step-size δ, function $\nabla \log \pi(x)$.
  for $n = 2 : N$ **do**
    Draw $\xi \sim \mathcal{N}(0, \delta \mathbb{I}_d)$
    Set $x_k = x_{k-1} + \delta \nabla \log \pi(x_{k-1}) + \sqrt{2\delta}\xi$
  end for

Under some regularity assumptions, namely L-Lipschitz continuity of $\nabla \log \pi$ and $\delta < 1/L$, the Markov chain $(X_n)_{n\geq 0}$ is ergodic with stationary distribution $\pi_\delta(x)$ close to π [44]. Additionally, when π is log-concave, ULA inherits the favourable properties of (3.5) and converges to $\pi_\delta(x)$ geometrically fast with good convergence rates, offering an efficient Bayesian computation methodology for high dimensional problems [44].

3.3 Performance & accuracy of MCMC methods

After defining some of the most popular exact & inexact MCMC methods in the literature, we will briefly describe some approaches that we use in this thesis to evaluate the performance and accuracy of the sampling schemes we will propose.

3.3.1 Asymptotic bias and variance

Recall that the goal of MCMC methods is to accurately approximate expectations (i.e., integrals) of the form

$$\mathbb{E}_\pi(\phi(x)) = \int_{\mathbb{R}^d} \phi(x)\pi(x)\mathrm{d}x. \tag{3.7}$$

Assuming that we have access to an ergodic Markov chain $x_1, x_2, \ldots$ with invariant density π (in the case of exact MCMC methods) or π_δ (in the case of inexact MCMC methods with stepsize δ), we can approximate the latter expectation by calculating the following sum

$$\hat{\phi}_n = \frac{1}{n}\sum_{i=1}^{n} \phi(x_i).$$

One way to assess the accuracy of the estimator $\hat{\phi}_n$ is to compute its asymptotic mean squared error (MSE). The latter can be expressed as

$$\mathrm{MSE}_{\hat{\phi}_{n\to\infty}} = \left(\mathrm{Bias}_{n\to\infty}(\hat{\phi}_n)\right)^2 + \mathrm{Var}_{n\to\infty}(\hat{\phi}_n),$$

that is, the combination of the **asymptotic bias**, given by

$$\mathrm{Bias}_{n\to\infty}(\hat{\phi}_n) = \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} \phi(x_i) - \int_{\mathbb{R}^d} \phi(x)\pi(x)\mathrm{d}x,$$

and the **asymptotic variance**, given by

$$\mathrm{Var}_{n\to\infty}(\hat{\phi}_n) = \lim_{n\to\infty} n\mathrm{Var}(\hat{\phi}_n).$$

Regarding the exact MCMC methods mentioned above, since their limiting invariant density is the true target distribution π, their asymptotic bias vanishes and thus their asymptotic MSE will be equal to their asymptotic variance, but they normally require a large number of iterations to obtain a stable estimator.

In the case of inexact MCMC methods, such as the unadjusted Langevin algorithm, they have an asymptotic bias associated with targeting a different density $\pi_\delta(x)$ instead of π, which can be reduced by decreasing δ, and vanishes as $\delta \to 0$. However, decreasing δ deteriorates the convergence properties of the chain and amplifies the associated non-asymptotic bias and variance. Therefore, to apply ULA to large problems in a computationally efficient way, it is necessary to use values of δ that are close to the stability limit $1/L$, to obtain a good trade-off between bias and variance.

The investigation of efficient MCMC algorithmic parameters to have a good balance between asymptotic bias and variance, and the development of new MCMC methods and techniques that aim to reduce asymptotic bias/variance is an active area of research. See, e.g., [41, 10, 124, 84].

3.3.2 Kullback–Leibler divergence

Another way to measure the accuracy of MCMC methods is to compute the Kullback–Leibler (KL) divergence [79] between the target distribution with density π and the approximated distribution generated by n samples of an MCMC method with stepsize δ, whose density we will call π_δ^n. This is defined as

$$D_{\mathrm{KL}}(\pi \parallel \pi_\delta^n) = \int_{-\infty}^{+\infty} \pi(x) \log\left(\frac{\pi(x)}{\pi_\delta^n(x)}\right) \mathrm{d}x.$$

Since the distribution π_δ^n is computed from n samples, the latter expression is usually computed numerically. For instance, one can use built-in functions that compute the relative entropy between the target distribution and a numerical fitting of a distribution to the grouped data, i.e., an estimate of the density function from the available Markov chain samples. See, e.g., [58, Section 24.5].

3.3.3 Autocorrelation function

Since MCMC methods produce correlated samples, it is desirable to generate samples with the lowest possible correlation between them, which can be seen as an indicator of how much information the samples carry from the target distribution. A high correlation indicates that the target distribution has not been explored enough and thus a large number of samples would be needed, which can be computationally expensive.

One way to measure the latter is the computation of the autocorrelation function (ACF) at lag τ [62, Section 3]. It measures the correlation between the univariate samples x_i and $x_{i+\tau}$ from N samples generated by an MCMC method, and is defined as

$$r_\tau = \frac{\sum_{i=1}^{N-\tau}(X_i - \bar{X})(X_{i+\tau} - \bar{X})}{\sum_{i=1}^{N}(X_i - \bar{X})},$$

where $\bar{X}$ is the sample mean. We would then like r_τ to decay to zero as quickly as possible as τ increases. This is commonly presented in an ACF plot that illustrates this decay for different

values of τ. Since this is a univariate estimator, in the multivariate case one can study the evolution of a particular component of the generated multidimensional samples.

3.3.4 Effective sample size (ESS)

A heuristic that also allows comparing the efficiency between MCMC methods is the effective sample size (ESS) [8, Section 2.3], defined by

$$\text{ESS} = \frac{N}{1 + 2\sum_{\tau=1}^{\infty} r_\tau},$$

where N is the total sample size and r_τ is the autocorrelation at lag τ. In practice, the sum in the denominator is truncated at lag k when r_k is less than some threshold value close to 0 (e.g., 0.05) [22, Section 15.5]. In a nutshell, the ESS is the number of independent samples with the same estimation power as the N autocorrelated samples. This means that the closer the ESS is to N, the better the quality of the MCMC samples.

3.4 Proximal MCMC methods

We are now ready to consider the class of models given by (2.7), which are not smooth. Unfortunately, ULA, MALA and HMC cannot be directly applied to such models, as they require Lipschitz differentiability of $\log p(x|y,\beta)$. Proximal MCMC methods address this difficulty by carefully constructing the following smooth approximation

$$p^\lambda(x|y,\beta) = \frac{p(y|x)p^\lambda(x|\beta)}{\int_{\mathbb{R}^d} p(y|x)p^\lambda(x|\beta)\mathrm{d}x} \propto \exp[-f_y(x) - \beta^\mathsf{T} g^\lambda(x)], \tag{3.8}$$

where

$$g^\lambda(x) = [g_1^\lambda(x), \ldots, g_p^\lambda(x)],$$

that is, each non-smooth term $g_i(x)$, $i \in \{1,\ldots,p\}$ is replaced by its Moreau-Yosida envelope[2]

$$g_i^\lambda(x) = \min_{u\in\mathbb{R}^d} \left\{ g_i(u) + \frac{1}{2\lambda}\|x-u\|^2 \right\}, \tag{3.9}$$

where $g_i^\lambda(x) \to g_i(x)$ as $\lambda \to 0$. This leads to the differentiable log-posterior with gradient given by

$$\begin{aligned}\nabla \log p^\lambda(x|y,\beta) &= -\nabla f_y(x) - \nabla(\beta^\mathsf{T} g^\lambda(x)), \\ &= -\nabla f_y(x) - \frac{1}{\lambda}\sum_{i=1}^{p}\left(x - \text{prox}_{\beta_i g_i}^\lambda(x)\right),\end{aligned}$$

and Lipschitz constant $L = L_f + p/\lambda$ where

$$\text{prox}_{g_i}^\lambda(x) = \underset{u\in\mathbb{R}^d}{\text{argmin}} \left\{ g_i(u) + \frac{1}{2\lambda}\|x-u\|^2 \right\}.$$

As an illustration, Figure 3.1 shows the Moreau-Yosida envelope of the Laplace distribution $0.5\exp(-|x|)$ and the uniform distribution $\mathcal{U}_{[-1,1]}$ using different values for λ. As can be seen, the lower the value of λ, the better the approximation of the true distribution.

We have now a differentiable log-posterior distribution $p^\lambda(x|y,\beta)$ that satisfies all the regularity conditions required by ULA, MALA and HCM, and can be made arbitrarily close to the original model $p(x|y,\beta)$ by tuning a regularisation parameter $\lambda > 0$. It will define the state-of-the-art sampling schemes that we will take as comparative MCMC methods that we

[2] If the calculation of the proximal operator of the sum of some elements of g is possible, it is not necessary to replace each of these elements of the vector g with its corresponding Moreau-Yosida envelope. In addition, if there is some $g_k(x)$, $k \in \{1,\ldots,p\}$ that is Lipschitz differentiable, its gradient can be computed directly.

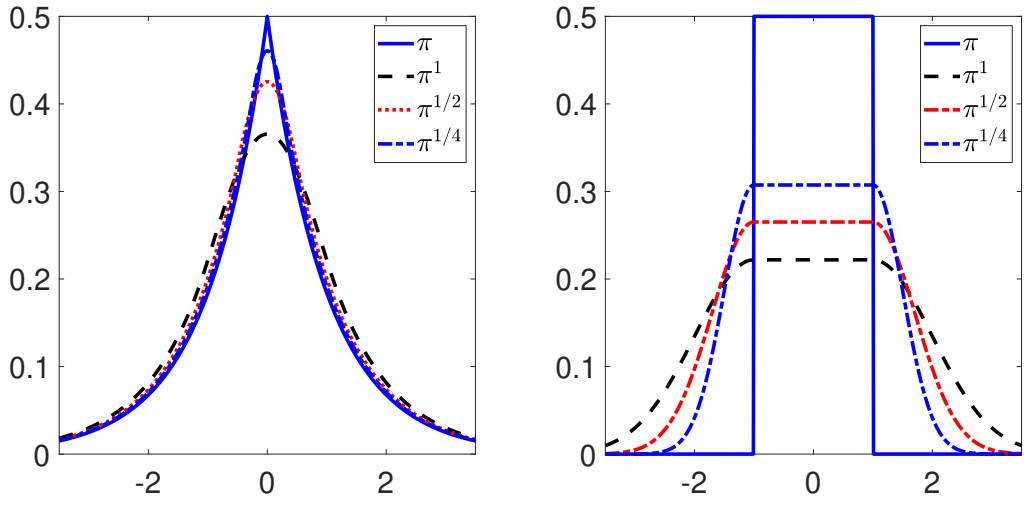


Figure 3.1: Plots for (a) Laplace and (b) uniform distribution in $[-1, 1]$, and their Moreau-Yosida approximations.

will present below, and will allow us to build and propose our novel and enhanced MCMC methods in the following chapters.

3.4.1 Moreau-Yosida unadjusted Langevin algorithm

Given the smooth approximation $p^\lambda(x|y,\beta)$, we define the auxiliary Langevin SDE

$$\mathrm{dX}_t = \nabla \log p^\lambda(\mathrm{X}_t|y,\beta)\mathrm{d}t + \sqrt{2}\mathrm{d}W_t\,, \tag{3.10}$$

and derive the MYULA Markov chain [45] by discretising this SDE by the EM method

$$X_{n+1} = X_n - \delta\nabla f_y(X_n) - \frac{\delta}{\lambda}\sum_{i=1}^{p}\left(X_n - \mathrm{prox}^\lambda_{\beta_i g_i}(X_n)\right) + \sqrt{2\delta}Z_{n+1},\,. \tag{3.11}$$

The main benefit of the MYULA is that now since $p^\lambda(x|y,\beta)$ is smooth and preserves log-concavity, the results from [44, 43] apply, hence providing an efficient method for application in imaging problems. In addition, the asymptotic bias can be removed, if necessary, by complementing MYULA with an MH step [95], which is useful for benchmarking purposes.

As mentioned previously, despite being relatively recent, proximal MCMC methods have already been successfully applied to many large-scale inference problems related to imaging sciences [25, 60, 126], and machine learning [128, 80].

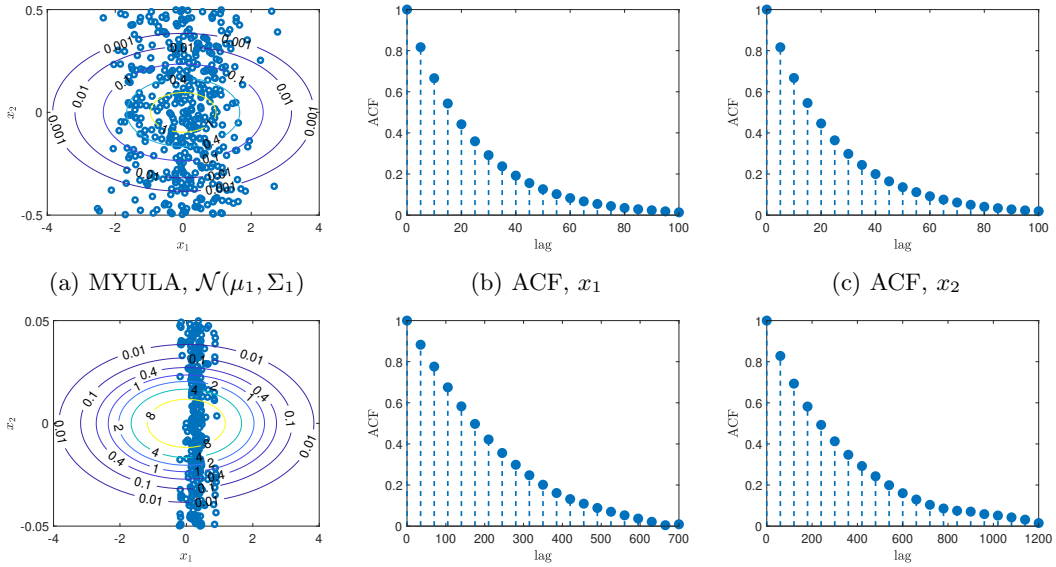
3.4.2 Limitations of MYULA and proximal MALA methods

One of the main limitations of ULA, MALA and their proximal variants is that they are all derived from the EM approximation (3.6) of the Langevin SDE. This approximation is mainly used because it is computationally efficient in high dimensions, it is easy to implement, and it can be rigorously theoretically analysed. However, the EM approximation is not particularly suitable for problems that are ill-conditioned or ill-posed as its performance is very sensitive to the anisotropy of the target density, which is a common feature of imaging problems. More precisely, in order to be useful for Bayesian computation, the EM approximation of the Langevin SDE (3.10) has to be numerically stable.

For MYULA, this requires using a step-size $\delta < 2/L$ with $L = L_f + 1/\lambda$, where we recall that L_f is the Lipschitz constant of ∇f_y and that λ controls the quality of the approximation

$p^\lambda(x|y,\beta)$ of $p(x|y,\beta)$. This restriction essentially guarantees that the chain moves slowly enough to follow changes in $\nabla \log p^\lambda(x|y,\beta)$ in a numerically stable manner, particularly along directions of fast change. However, this is problematic when $p^\lambda(x|y,\beta)$ has some directions or regions of the parameter space that change relatively very slowly, as the chain will struggle to properly explore the solution space and will require a very large number of iterations to converge. In imaging models, this typically arises when the likelihood $p(y|x)$ has identifiability issues (e.g, if it involves an observation operator A for which $\mathrm{A}^\mathsf{T}\mathrm{A}$ is badly conditioned or rank deficient), or if we seek to use a small value of λ to bring $p^\lambda(x|y,\beta)$ close to $p(x|y,\beta)$.

To highlight this issue, we report below two simple illustrative experiments where MYULA is applied to a two-dimensional Gaussian distribution. In this case, there is no non-smooth term g and the step-size restriction is dictated by the Lipschitz constant of f, but the same phenomenon arises in more general models. In the first experiment we consider $\mu_1 = (0,0)$ and $\Sigma_1 = \mathrm{diag}(1, 10^{-2})$ (i.e., $L_f = 10^2$); whereas in the second experiment we use $\mu_2 = (0,0)$ and $\Sigma_2 = \mathrm{diag}(1, 10^{-4})$ (i.e., $L_f = 10^4$). The results are presented in Figure 3.2. Notice that in the first case MYULA explores the distribution very well, showing a good rate of decay in the autocorrelation functions of both components. However, in the second case, MYULA exhibits poor convergence properties as it struggles to explore the first component.



(a) MYULA, $\mathcal{N}(\mu_1, \Sigma_1)$ (b) ACF, x_1 (c) ACF, x_2

(d) MYULA, $\mathcal{N}(\mu_2, \Sigma_2)$ (e) ACF, x_1 (f) ACF, x_2

Figure 3.2: Two-dimensional Gaussian distribution: (a) 10^3 samples generated by MYULA using the target distributions $\mathcal{N}(\mu_1, \Sigma_1)$ with $\delta = 2/(L+\ell) = 1.98\times10^{-2}$ where $L = 1/\sigma_{11}^2 = 100$ and $\ell = 1/\sigma_{22}^2 = 1$; and (d) 5×10^3 samples generated by MYULA using the target distributions $\mathcal{N}(\mu_2, \Sigma_2)$ with $\delta = 2/(L+\ell) = 1.99\times10^{-4}$ where $L = 1/\sigma_{11}^2 = 10^4$ and $\ell = 1/\sigma_{22}^2 = 1$. Autocorrelation functions of the (b)-(e) first and (c)-(f) second component (i.e., x_1 and x_2) of the samples generated by the ULA algorithm, having $\mathcal{N}(\mu_1, \Sigma_1)$ and $\mathcal{N}(\mu_2, \Sigma_2)$ as target distributions, respectively.

This limitation of the EM approximation could be partially mitigated by preconditioning the gradient $\nabla \log \pi_\lambda$ by considering a Langevin SDE on an appropriate Riemannian manifold, as recommended in [59], and in a spirit akin to natural gradient descent and Newton optimisation methods. The preconditioning procedure proposed in [59] is very effective but too expensive for imaging models because it requires evaluating quantities related to second and third-order derivatives of $\log \pi_\lambda$ and performing expensive matrix operations. Conversely, simple procedures such as preconditioning with a pseudo-inverse of the Hessian matrix of the log-likelihood function are computationally efficient but do not typically lead to significant improvements in performance because they do not take into account the geometry of the log-prior. The development of computationally efficient yet effective preconditioning strategies for imaging models is an active research topic, see, e.g., [85, 86].

Chapter 4

Proximal SK-ROCK MCMC method

The results presented in this chapter appeared in the SIAM Journal on Imaging Sciences [97] and are joint work with Marcelo Pereyra and Konstantinos C. Zygalakis.

4.1 Introduction

This chapter seeks to exploit recent developments in the numerical analysis of SDEs to significantly improve the computational efficiency of proximal MCMC methodology. More precisely, we propose to use a state-of-the-art orthogonal Runge-Kutta-Chebyshev stochastic approximation of the Langevin diffusion process [2] that is significantly more computationally efficient than the conventional Euler-Maruyama approximation used by existing proximal MCMC methods. In particular, we present a new proximal MCMC method that applies this approximation to the Moreau-Yosida regularised Langevin diffusion underpinning the MYULA algorithm, and show both theoretically and empirically that this leads to dramatic improvements in convergence speed and estimation accuracy.

This highly advanced Runge-Kutta stochastic integration scheme extends the deterministic Chebyshev method [1] to SDEs, in order to approximate (3.8), instead of the basic EM discretisation scheme that underpins MYULA. Its implementation is straightforward as it only requires knowledge of the gradient operator $\nabla \log p^{\lambda}(x|y,\beta)$ given by (3.4), which is also used in MYULA.

Furthermore, this sophisticated method performs $s \in \mathbb{N}^*$ evaluations of $\nabla \log p^{\lambda}(x|y,\beta)$ at carefully chosen extrapolated points determined by Chebyshev polynomials, unlike MYULA, which uses a single evaluation of $\nabla \log p^{\lambda}(x|y,\beta)$ per iteration. In this regard, the resulting computational benefit of this scheme is similar to the one of accelerated optimization methods when compared to gradient descent. In fact, the deterministic Runge-Kutta-Chebyshev method was recently shown to have similar theoretical convergence properties to Nesterov's accelerated optimisation algorithms in the case of strongly convex functions [49].

4.2 The Algorithm

The proposed proximal SK-ROCK method is presented in Algorithm 6 below, where T_s denotes the Chebyshev polynomial of order s of the first kind, defined recursively by $T_{k+1} = 2xT_k(x) - T_{k-1}(x)$ with $T_0(x) = 1$ and $T_1(x) = x$. The two main parameters of the algorithm are the number of stages $s \in \mathbb{N}^*$ and the step-size $\delta \in (0, \delta_s^{max}]$. Notice that the range of admissible values for δ is controlled by s: for any $s \in \mathbb{N}^*$, the maximum allowed step-size is given by $\delta_s^{max} = l_s/(L_f + 1/\lambda)$ with $l_s = [(s-0.5)^2(2-4/3\eta) - 1.5]$ and $\eta = 0.05$ [2]. Violating this upper bound leads to a potentially explosive Markov chain. Also note that in the case of $s = 1$ the method reduces to MYULA.

Algorithm 6 SK-ROCK

Require: : $X_0 \in \mathbb{R}^d$, $\lambda > 0$, $n \in \mathbb{N}$, $s \in \{3, \dots, 15\}$, $\eta = 0.05$.
Compute $l_s = (s - 0.5)^2(2 - 4/3\eta) - 1.5$
Compute

$$\omega_0 = 1 + \frac{\eta}{s^2}, \;\; \omega_1 = \frac{T_s(\omega_0)}{T_s'(\omega_0)}, \;\; \mu_1 = \frac{\omega_1}{\omega_0}, \;\; \nu_1 = s\omega_1/2, \;\; k_1 = s\omega_1/\omega_0$$

Choose $\delta \in (0, \delta_s^{max}]$, where $\delta_s^{max} = l_s/(L_f + 1/\lambda)$
for $i = 0 : n - 1$ **do**
  Set $\tilde{\mathrm{X}}_0 = X_i$
  Sample $\xi_{i+1} \sim \mathcal{N}(0, 2\delta\mathbb{I}_d)$
  Compute$\tilde{\mathrm{X}}_1 = \tilde{\mathrm{X}}_0 + \mu_1\delta\nabla \log p^\lambda(\tilde{\mathrm{X}}_0 + \nu_1\xi_{i+1}|y, \beta) + k_1\xi_{i+1}$
  for $j = 2 : s$ **do**
    Compute $\mu_j = 2\omega_1 T_{j-1}(\omega_0)/T_j(\omega_0)$, $\nu_j = 2\omega_0 T_{j-1}(\omega_0)/T_j(\omega_0)$, $k_j = 1 - \nu_j$,
    Compute $\tilde{\mathrm{X}}_j = \mu_j\delta\nabla \log p^\lambda(\tilde{\mathrm{X}}_{j-1}|, y, \beta) + \nu_j\tilde{\mathrm{X}}_{j-1} + k_j\tilde{\mathrm{X}}_{j-2}$
  end for
  Set $X_{i+1} = \tilde{\mathrm{X}}_s$
end for
Output: Samples $X_1, \dots, X_n$.

The values of δ and s are subject to standard bias-variance trade-offs. On the one hand, to optimise the mixing properties of the algorithm one would like to choose δ as large as possible. MYULA, based on the EM method, requires setting $\delta < \delta_1^{max} = 1/(L_f + 1/\lambda)$ for stability, but in SK-ROCK one can in principle take δ arbitrarily large by increasing the value of s. However, this would also increase the asymptotic bias and the computational cost per iteration. In our numerical experiments we found that a good trade-off in terms of bias, variance, and computational cost per iteration was achieved by setting $3 < s < 15$ and using a value of δ that is close to the maximum allowed step-size δ_s^{max}. As a general rule for imaging problems, we recommend using $s = 15$ in problems that are strongly log-concave, and $s = 10$ otherwise.

To illustrate the benefits of using the proximal SK-ROCK method instead of MYULA, we repeat the two Gaussian experiments reported in Figure 3.2 with Algorithm 6. The results are shown in Figure 4.1, and where we have set the number of s optimally by using (4.8). Observe that because the SK-ROCK method is allowed to use a larger step-size δ in a stable manner, it produces, for the same computational cost (i.e., number of gradient evaluations), samples that are significantly less correlated than MYULA with respect to the slow component. We also observe in Figure 4.1 that this allows SK-ROCK to explore the target distribution more accurately.

4.3 Mean-square stability analysis

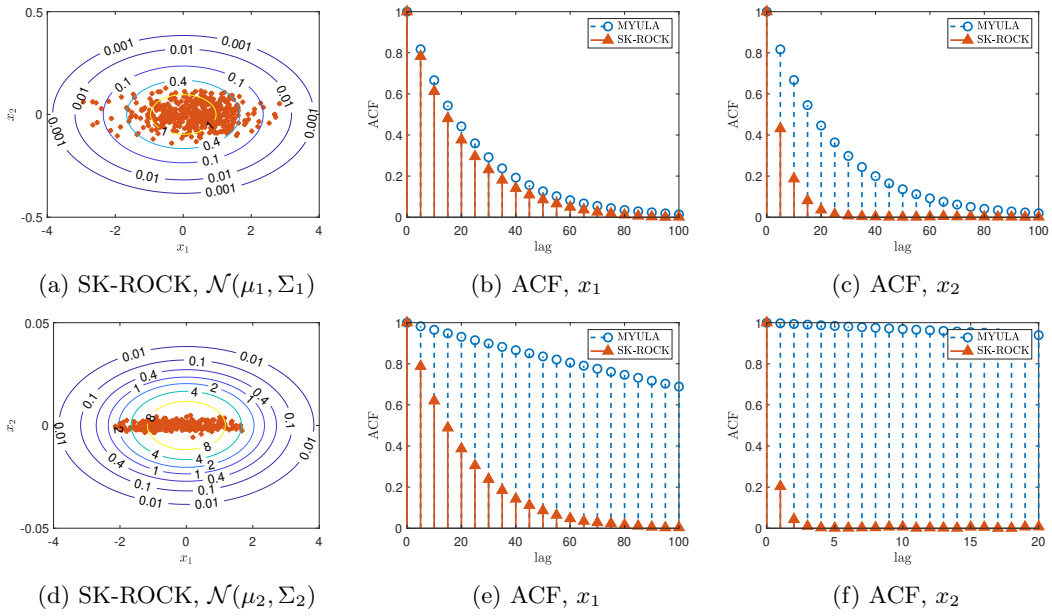
We now discuss the mean-square stability properties of SK-ROCK and the EM method. In particular, we consider the following test equation that is widely used in the numerical analysis literature [69, 70] to benchmark SDE solvers

$$dX(t) = \gamma X(t)dt + \mu X(t)dW(t), \;\; X(0) = 1, \tag{4.1}$$

where γ, $\mu \in \mathbb{R}$, which has the solution $X(t) = \exp[(\gamma - 1/2\mu^2)t + \mu W(t)]$. It is easy to show using Ito calculus that when $2\gamma + \mu^2 < 0$

$$\lim_{t\to\infty} \mathbb{E}(|X(t)|^2) = 0.$$

We want to understand for what range of the step-size δ would a numerical discretisation X_n of (4.1) behave in a similar manner as $n \to \infty$, *i.e.* $\mathbb{E}(|X_n|^2) \to 0$. In the case of EM one has



(a) SK-ROCK, $\mathcal{N}(\mu_1, \Sigma_1)$ (b) ACF, x_1 (c) ACF, x_2

(d) SK-ROCK, $\mathcal{N}(\mu_2, \Sigma_2)$ (e) ACF, x_1 (f) ACF, x_2

Figure 4.1: Two-dimensional Gaussian distribution: (a) $10^3/s$ samples generated by the SK-ROCK algorithm ($s = 2$) using the target distribution $\mathcal{N}(\mu_1, \Sigma_1)$ with $\delta = 4.82 \times 10^{-2}$ and (d) $5 \times 10^3/s$ samples ($s = 16$) using the target distribution $\mathcal{N}(\mu_2, \Sigma_2)$ with $\delta = 4.84 \times 10^{-2}$. Autocorrelation functions of the (b)-(e) first and (c)-(f) second component (i.e., x_1 and x_2) of the samples generated by the SK-ROCK algorithm, having $\mathcal{N}(\mu_1, \Sigma_1)$ and $\mathcal{N}(\mu_2, \Sigma_2)$ as target distributions, respectively.

that

$$X_{n+1} = X_n + \delta\gamma X_n + \sqrt{\delta}\mu X_n Z_{n+1},$$

and hence

$$\mathbb{E}(|X_{n+1}|^2) = R(p,q)\mathbb{E}(|X_n|^2), \quad R(p,q) = (1+p)^2 + q^2, \quad p = \delta\gamma, q = \sqrt{\delta}\mu.$$

In order to have $\mathbb{E}(|X_n|^2) \to 0$ one needs that $R(p,q) < 1$. We visualise the values of admissible p, q for the EM method in Figure 4.2(a), where we can see that there is only a very small portion of the true mean-square stability domain ($2p + q^2 < 0$) covered by it (anything on the left-hand side of the dotted line in Figure 4.2(a)-(b) belongs to the true stability domain). This implies that when one or both of the parameters γ, μ are large, a very small δ needs to be chosen to be stable (for example when $\mu = 0$ one recovers the stability condition $\delta < -2\gamma^{-1}$ for the Langevin SDE). In the case of SK-ROCK, one has that

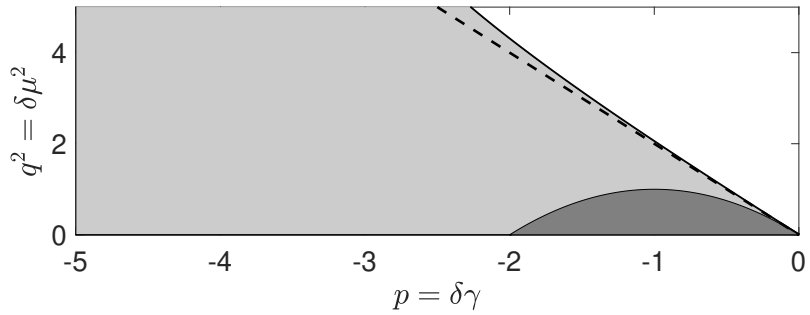
$$R(p,q) = R_1(p)^2 + R_2(p)^2 q^2,$$

where R_1 and R_2 are given by (4.4).

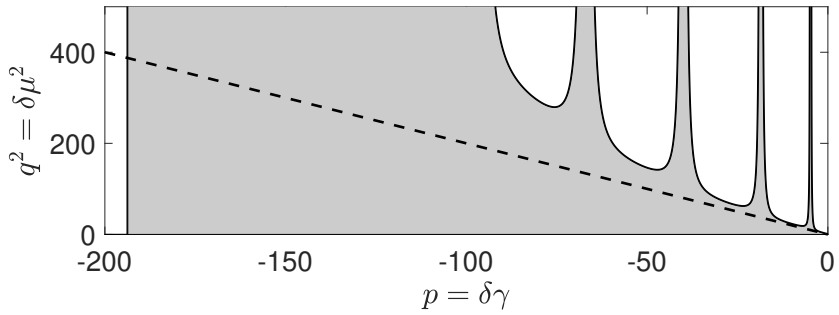
Similarly to the case of the EM method, we now plot the mean-square stability domain of SK-ROCK in Figure 4.2(b). As we can see, a significantly larger portion of the true mean-square stability domain is now covered when compared to the EM method. One can show, using the properties of Chebyshev polynomials [2], that for SK-ROCK the coverage of the mean-square stability domain increases quadratically in s; i.e., that if $(p,q) \in \{2p + q^2 < 0 \ \cap \ p < C(\eta)s^2\}$ then $R(p,q) < 1$ for the SK-ROCK method.

In contrast, if for comparison one would consider s-steps of the EM method, the corresponding coverage of the mean-square stability domain would be linear in s. This means that for the same number of gradient evaluations s, one can choose a much larger step-size δ for SK-ROCK and still integrate equation (4.1) in a stable manner. The spikes observed in 4.2(b) at specific values of p correspond to roots of the polynomial $R_2(p)$ defined in (4.4); these are determined by the values of s and η, and by the roots of the Chebyshev polynomial of the second kind

U_{s-1}.



(a) EM (dark grey) and SK-ROCK (light grey) mean-square stability regions.



(b) SK-ROCK mean-square stability region ($s = 10$, $\eta = 0.05$)

Figure 4.2: Mean-square stability domains for (a) EM and and (b) SK-ROCK (with $s = 10$) in the $p - q^2$ plane. The dashed line represents the upper boundary of the true mean-square stability domain.

4.4 Computational Complexity

To the best of our knowledge, it is not possible to establish general complexity results for Runge-Kutta-Chebyshev methods by using existing analysis techniques, and we are currently investigating new bespoke techniques to study SK-ROCK. This is an important difference w.r.t. the EM scheme used in MYULA, for which there are detailed non-asymptotic convergence results available that can be used to characterise its computational complexity [44]. Nevertheless, it is possible to get an intuition for the computational complexity of SK-ROCK by theoretically analysing its convergence properties for a d-dimensional Gaussian target distribution with density $\pi(x) \propto \exp\left(-0.5x^\top \Sigma^{-1} x\right)$, and $\Sigma = \mathrm{diag}(\sigma_1^2, ..., \sigma_d^2)$. More precisely, we study the convergence of SK-ROCK in the 2-Wasserstein distance, as a function of the number of gradient evaluations and the condition number $\kappa = \sigma_{\max}^2/\sigma_{\min}^2$, and compare it with MYULA. This is achieved by analysing in full generality the numerical solution of the Langevin SDE associated with π, given by

$$dX_t = -\Sigma^{-1} X_t dt + \sqrt{2} dW_t \,, \tag{4.2}$$

by a one step numerical integrator, which yields (in general) a recurrence of the form

$$X_{n+1}^i = R_1(z_i) X_n^i + \sqrt{2\delta} R_2(z_i) \xi_{n+1}^i, \quad \xi_{n+1}^i \sim N(0,1), \tag{4.3}$$

where $z_i = -\delta/\sigma_i^2$ and $X_0 = (x_0^1, ..., x_0^d)^T$ is a deterministic initial condition. For the EM scheme used in MYULA we have $R_1(z) = 1 + z$ and $R_2(z) = 1$, and for the SK-ROCK we have that [2]

$$R_1(z) = \frac{T_s(\omega_0 + \omega_1 z)}{T_s(\omega_0)}, \;\; R_2(z) = \frac{U_{s-1}(\omega_0 + \omega_1 z)}{U_{s-1}(\omega_0)} \left(1 + \frac{\omega_1}{2} z\right) , \tag{4.4}$$

where T_s, U_s are Chebyshev polynomials of first and second kind respectively and

$$\omega_0 = 1 + \frac{\eta}{s^2}, \quad \omega_1 = \frac{T_s(\omega_0)}{T_s'(\omega_0)}.$$

By using the fact that Gaussian distributions are closed under linear transformations, and assuming that the initial condition X_0 is deterministic, we derive the distribution of X_n for any $\delta > 0$ and obtain the following general result that holds for the EM (MYULA) method and for SK-ROCK.

Proposition 1. *Let $\pi(x) \propto \exp\left(-0.5x^T\Sigma^{-1}x\right)$ with $\Sigma = diag(\sigma_1^2, ..., \sigma_d^2)$, and let Q_n be the probability measure associated with n iterations of the generic Markov kernel (4.3). Then the 2-Wasserstein distance between π and Q_n is given by*

$$W_2(\pi; Q_n)^2 = \sum_{i=1}^{d} \left(D_n(z_i, x_0^i) + B_n(z_i, \sigma_i)\right) \tag{4.5}$$

where

$$D_n(x,u) = (R_1(x))^{2n}u^2, \quad B_n(x,u) = \left[u - \sqrt{2\delta}R_2(x)\left(\frac{1-(R_1(x))^{2n}}{1-(R_1(x))^2}\right)^{1/2}\right]^2.$$

In addition, the following bound holds

$$W_2(\pi; Q_{n+1})^2 \leq W_2(\pi;\tilde{\pi})^2 + CW_2(\tilde{\pi}, Q_n)^2 \tag{4.6}$$

where

$$\tilde{\pi} = \mathcal{N}\left(0, 2\delta(R_2(z))\left[\frac{1}{1-R_1^2(z)}\right]\right),$$

is the numerical invariant measure and

$$C = \max_{1\leq i\leq d} R_1(z_i)^2. \tag{4.7}$$

The bound (4.6) can now be used to compare the EM and the SK-ROCK method in terms of how many gradient evaluations are required to achieve $W_2(\pi; Q_n) < \varepsilon$ for some desired accuracy level $\varepsilon > 0$. We see that the W_2^2 distance between π and Q_n involves two terms. The first term $W_2(\pi;\tilde{\pi})^2$ relates directly to the asymptotic bias of the method (recall that without a Metropolis correction step, any generic approximation of (4.2) will have some asymptotic bias because it will not exactly converge to π). The second term $CW_2(\tilde{\pi}, Q_n)^2$ related to the convergence of the chain to the stationary distribution $\tilde{\pi}$, with the C controlling the convergence rate. In imaging problems, the computational complexity is usually largely dominated by the second term in (4.6) because of the dimensionality involved.

For the case of the EM (MYULA) method it is known [42] that, with a suitable choice of δ, the number of gradient evaluations that one needs to take in order to achieve $W_2(\pi; Q_n) < \varepsilon$ is of order $\mathcal{O}(\kappa)$, where we recall that $\kappa = \sigma_{\max}^2/\sigma_{\min}^2$ is the condition number of Σ. For SK-ROCK, the number of gradient evaluations depends on the choice of s and δ. Our focus is on problems where κ is large, where the optimal performance is achieved by minimising C by setting the number of internal stages s of each step to be

$$s = \left\lceil \sqrt{\frac{\eta}{2}(\kappa - 1)} \right\rceil, \tag{4.8}$$

with $\eta = 0.05$, and

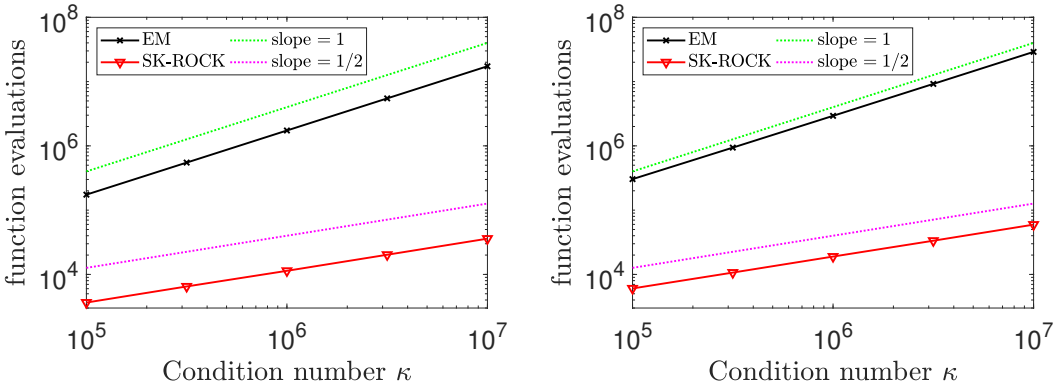
$$\delta = \frac{\omega_0 - 1}{\ell_s \omega_1}, \quad \ell_s = \frac{1}{\sigma_{\max}^2}, \tag{4.9}$$

so that $C \approx (\sqrt{\kappa}-1)^2/(\sqrt{\kappa}+1)^2$ (see [49] and Appendix B for details). In that case, and under the assumption that $W_2(\pi, \tilde{\pi}) \ll \epsilon$ so that $W_2(\pi; Q_n)$ is dominated by the term $CW_2(\tilde{\pi}, Q_n)^2$

related to convergence to $\tilde{\pi}$, we observe that the number of gradient evaluations required to achieve $W_2(\pi; Q_n) < \varepsilon$ is of the order of $\mathcal{O}(\sqrt{\kappa})$ instead of $\mathcal{O}(\kappa)$, similarly to the behaviour of accelerated algorithms in optimization [92, 64]. These convergence results are illustrated in Figure 4.3, where we plot the number of gradient evaluations required to achieve $W_2(\pi; Q_n) < \varepsilon$ as a function of the conditioning number κ for the EM method and for SK-ROCK, where π is a 100-dimensional Gaussian distribution with mean zero and covariance $\Sigma = \text{diag}(\sigma_1^2, ..., \sigma_d^2)$, with decreasing diagonal elements uniformly spread between $\sigma_1 = 1$ and $\sigma_d = 1/\kappa$.

One can also simplify the non-asymptotic W_2^2 results of Appendix A to obtain non-asymptotic results for the estimation bias of the EM and SK-ROCK methods for the mean of Gaussian target densities (this is a weaker analysis than convergence in W_2^2). As in the case of the W_2^2 analysis, the number of gradient evaluations to attain a prescribed non-asymptotic bias for the mean is of order $\mathcal{O}(\sqrt{\kappa})$ for SK-ROCK, whereas it is of order $\mathcal{O}(\kappa)$ for the EM method. Both methods are asymptotically unbiased for the mean for Gaussian models.

We emphasise at this point that there are situations where one would not observe any acceleration by using SK-ROCK, namely situations in which a very accurate solution is required and the bound (4.6) is dominated by the asymptotic bias term $W_2(\pi, \tilde{\pi})$. In that case, instead of using MYULA or SK-ROCK with a very small δ, we would recommend using the P-MALA method described in [95], which combines an EM approximation with an MH correction.



(a) $\varepsilon^2 = 10^{-1}$ (b) $\varepsilon^2 = 10^{-3}$

Figure 4.3: Wasserstein distance bounds, Gaussian analysis: Minimum number of gradient evaluations of the EM and SK-ROCK methods in order to have $W_2(P; Q_n)^2 < \varepsilon^2 W_2(P; Q_0)^2$, given different condition numbers κ.

4.5 Numerical experiments

In this section, we demonstrate the proposed SK-ROCK proximal MCMC methodology with a range of numerical experiments related to image deblurring, tomographic reconstruction, and hyperspectral unmixing, previously explained in Sections 2.1.1, 2.1.3 and 2.1.4, respectively. We have selected these experiments to represent a wide variety of configurations in terms of ill-posedness and ill-conditioning, strict and strong log-concavity, and dimensionality of y and x. Following our previous recommendation, in the experiments related to image deblurring and hyperspectral unmixing the model is strongly log-concave so we use $s = 15$, whereas for the tomography experiment we use $s = 10$. We report comparisons with the MYULA method [45] to highlight the benefits of using the SK-ROCK discretization as opposed to the conventional EM discretization used in Langevin and Hamiltonian algorithms [98].

To make the comparisons fair, in all experiments we use the same number of gradient (and proximal operator) evaluations for MYULA and SK-ROCK and compare their computational efficiency in several ways (the efficiency of an MCMC method is not an absolute quantity as it depends on the estimator considered). Because our aim is to illustrate the performance of SK-ROCK in Bayesian imaging problems, here we use the MYULA and SK-ROCK samples to

compute the following quantities:

1. The minimum mean square error solution given by the posterior mean $\mathrm{E}(x|y)$, which is a classic image point estimator.

2. The marginal posterior variances or standard deviations for the image pixels, which provide an indication of the performance of the methods in uncertainty quantification tasks.

3. The effective sample size (ESS)[1] of the fastest mixing component of the chain, calculated after burn-in (i.e., when the chain reaches steady-state).

4. The ESS of the slowest mixing component of the chain, also calculated after burn-in (these fast and slow components correspond to the one-dimensional subspaces where the Markov chains achieve their highest and lowest convergence rates respectively, and that we have identified via an estimate of the first and last eigenvectors of the samples posterior covariance).

We choose to report ESS values because these are intuitive quantities that are directly related to the variance of the Monte Carlo estimators, and hence provide an indication of the accuracy of the methods, up to estimation bias[2].

In addition to reporting estimates, we use autocorrelation plots to visually compare the convergence properties of both methods (again, we report the autocorrelation function for the fastest and the slowest components of the Markov chains). We also show the evolution of the estimation MSE across iterations and display the estimates of the marginal (pixelwise) standard deviations. The latter is useful for illustrating the differences in the performance of the methods, as second-order moments are more difficult to estimate by Monte Carlo integration than the posterior mean.

Notice that because the methods are compared with the same computational budget they do not produce the same number of samples, since their complexity per iteration is different. More precisely, if the MYULA chain has n-samples, then the SK-ROCK chain has only n/s samples, which is considerably lower. However, experiments show that SK-ROCK usually delivers higher ESS values because of its superior convergence properties. Similarly, to make the comparison of autocorrelation plots fair with regards to computational complexity, in all autocorrelation plots we apply a 1-in-s thinning to the MYULA chain to artificially boost its autocorrelation function decay rate by a factor of s.

4.5.1 One dimensional distributions

We start our numerical experiments by studying two simple one-dimensional distributions, namely the Laplace distribution and the uniform distribution in $[-1, 1]$, for which we can also perform computations exactly. Since both distributions are not Lipschitz differentiable we employ the corresponding Moreau-Yosida approximation using $\lambda = 10^{-5}$ to bring π_λ very close to π and deliver a good approximation. This implies that the largest step-size δ that can be used for MYULA is 2×10^{-5}, which is dramatically small. We set $\delta = 10^{-5}$ for MYULA and run the corresponding chain for $n = 15 \times 10^6$ iterations to create a situation where MYULA struggles to deliver a good approximation and that highlights the superior performance of SK-ROCK.

For SK-ROCK we use $s = 15$ and set δ as it is explained in Algorithm 6. Notice that we choose the (regularised) Laplace and the uniform distributions to illustrate the performance of the methods in two different scenarios: the regularised Laplace distribution is strongly log-concave near the mode and only strictly log-concave in the tails, which is problematic for the Langevin diffusion because the gradient remains constant as $|x|$ grows, whereas the regularised uniform distribution is flat over $[-1, 1]$ and hence has most of its mass in regions where the gradient is zero, and then strongly log-concave in the tails.

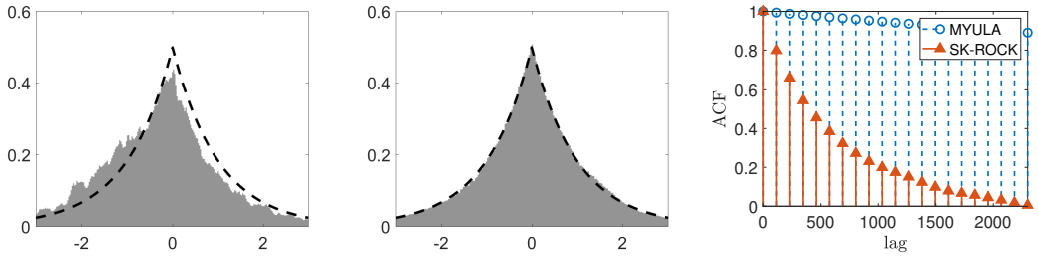
[1] Recall that $\mathrm{ESS} = n\{1 + 2\sum_k \rho(k)\}^{-1}$, where n is the total number of samples and $\sum_k \rho(k)$ is the sum of the K monotone sample auto-correlations which we estimated with the initial monotone sequence estimator [56].

[2] Note that the computation of ESS values is well-posed because $p(x|y)$ is log-concave. If $p(x|y)$ were heavy-tailed or multi-modal then we would need to consider robust efficiency indicators [122].

Table 4.1: Values of the step-size δ, effective sample sizes (ESS) and KL-divergence of the EM and SK-ROCK algorithms for the one-dimensional Laplace distribution.

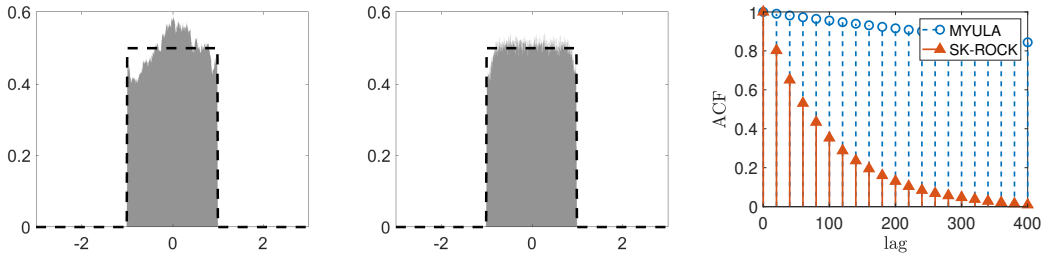
Stages s	**Method**	**Step-size** δ	**ESS**	**KL-Divergence**	**Speed-up**
-	MYULA	1.0×10^{-5}	3.6×10^{1}	4.8×10^{-2}	-
$s = 10$	SK-ROCK	1.7×10^{-3}	6.0×10^{2}	1.4×10^{-2}	16.67
$s = 15$	SK-ROCK	4.0×10^{-3}	9.5×10^{2}	1.0×10^{-2}	26.39

Figures 4.4 and 4.5 display the histogram approximations of the distributions obtained with the two methods, as well as the autocorrelation functions of the generated Markov chains. Observe that in both cases SK-ROCK significantly outperforms MYULA, which struggles to deliver a good approximation due to the step-size limitation and the limited number of iterations (this phenomenon is particularly clearly captured by the difference in decay speed in the autocorrelation plots). These results are quantitatively summarised in Tables 4.1 and 4.2 respectively, where we highlight that SK-ROCK delivers an ESS that is over 25 times larger than MYULA, while also achieving higher accuracy as measured by the Kullback-Leibler (KL) divergence between the empirical distribution and π_λ. For completeness, we also report the results using SK-ROCK with $s = 10$.



(a) MYULA (b) SK-ROCK, $s = 15$ (c) ACF

Figure 4.4: One-dimensional Laplace distribution: Histograms computed with (a) 15×10^6 samples generated by MYULA and (b) $15 \times 10^6/s$ samples generated by SK-ROCK from the approximated Laplace distribution, using an approximation parameter $\lambda = 10^{-5}$ and $s = 15$ for the SK-ROCK method. (c) Autocorrelation functions of the samples.



(a) MYULA (b) SK-ROCK, $s = 15$ (c) ACF

Figure 4.5: One-dimensional uniform distribution: Histograms computed with (a) 15×10^6 samples generated by MYULA and (b) $15 \times 10^6/s$ samples generated by SK-ROCK from the approximated uniform distribution, using an approximation parameter $\lambda = 10^{-5}$ and $s = 15$ for the SK-ROCK method. (c) Autocorrelation functions of the samples.

It is worth emphasising at this point that we could improve the ESS performance of both methods by increasing the value of λ, at the expense of some additional bias. In the case of the uniform distribution, this would lead to a considerable number of samples outside the true support $[-1, 1]$.

Table 4.2: Values of the step-size δ, effective sample sizes (ESS) and KL-divergence of the EM and SK-ROCK algorithms for the one-dimensional uniform distribution.

Stages s	**Method**	**Step-size** δ	**ESS**	**KL-Divergence**	**Speed-up**
-	MYULA	1.0×10^{-5}	1.7×10^{2}	1.3×10^{-2}	-
$s = 10$	SK-ROCK	1.7×10^{-3}	3.4×10^{3}	3.2×10^{-2}	20
$s = 15$	SK-ROCK	4.0×10^{-3}	4.9×10^{3}	3.9×10^{-2}	28.82

4.5.2 Image deblurring with total-variation prior

We now consider a non-blind image deblurring problem, previously described in Section 2.1.1, where we seek to recover a high-resolution image $x \in \mathbb{R}^d$ from a blurred and noisy observation $y = \mathrm{A}x + \xi$, where A is a known blur operator and $\xi \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_d)$. Figure 2.1 presents an experiment with the `cameraman` test image of size $d = 256 \times 256$ pixels, depicted in Figure2.1(a). Figure 2.1(b) shows an artificially blurred and noisy observation y, generated by using a 5×5 uniform blur and $\sigma = 0.58$, related to a blurred signal-to-noise ratio of 40dB. This problem is ill-conditioned i.e., A is nearly singular, thus yielding highly noise-sensitive solutions. To make the estimation problem well-posed, we use a total-variation norm prior that promotes solutions with spatial regularity. The resulting posterior distribution is given by

$$p(x|y) \propto \exp\left(-\|y - \mathrm{A}x\|^2/2\sigma^2 - \beta \mathrm{TV}(x)\right), \tag{4.10}$$

where $\mathrm{TV}(x)$ represents the total-variation pseudo-norm, previously described in Section 2.3.1, and $\sigma, \beta \in \mathbb{R}^+$ are model hyper-parameters that we assume fixed (in this experiment we use $\beta = 0.047$, determined using the method of [123]).

We use MYULA and SK-ROCK to draw Monte Carlo samples from (4.10) using $\lambda = L_f^{-1} = 0.21$. To make the comparison fair, we generate 10^3 samples using MYULA and $10^3/s$ samples using SK-ROCK for $s = 15$. We then use the generated samples to compute two quantities:

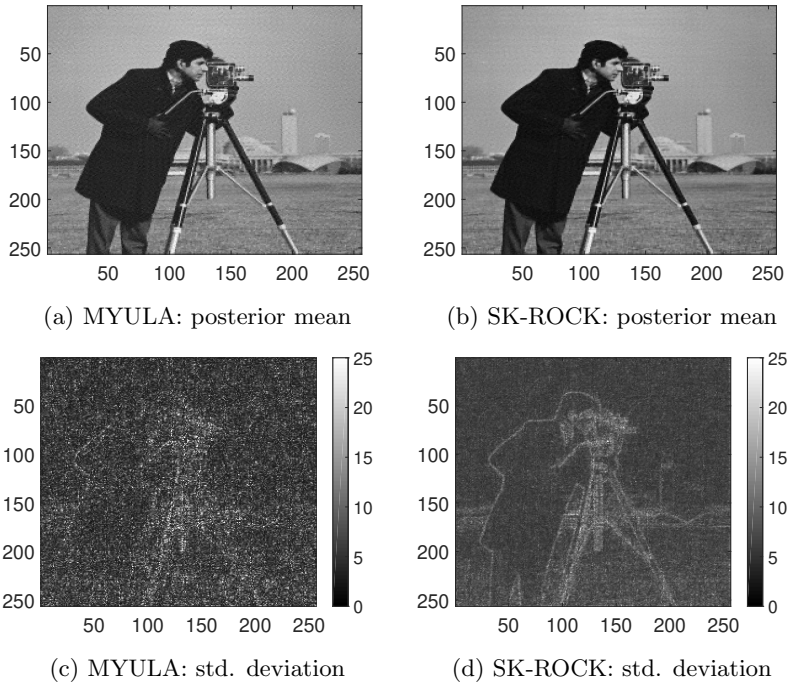
1. The minimum mean square error (MMSE) estimator of $x|y$, given by the posterior mean.
2. The pixel-wise (marginal) posterior standard deviation, which provides an indication of the level of confidence in each pixel value, as measured by the model.

The posterior standard deviation is useful to highlight features in the image that are difficult to accurately determine; in image deblurring problems these are the exact locations of edges and contours in the image. Notice that computing standard deviations require computing second-order statistical moments, which is more difficult than estimating the posterior mean, and hence requires a larger number of effective samples to produce stable estimates.

Figure 4.6 shows the MMSE estimator and the pixel-wise posterior standard deviation. Observe in these figures that while the estimates of the posterior mean obtained with MYULA and SK-ROCK are visually similar, the estimates of the pixel-wise standard deviations obtained with SK-ROCK are noticeably more accurate and in agreement with the results obtained by sampling the true posterior with an asymptotically unbiased Metropolised algorithm, see [95, Section 4.1]. In particular, the standard deviations estimated with SK-ROCK accurately capture the uncertainty in the location of the contours in the image, whereas MYULA produces very noisy results as it struggles to estimate second-order moments because of the step-size limitation and limited computation budget (with a sufficiently large number of iterations, MYULA would produce similar results to SK-ROCK).

Moreover, to rigorously analyse the convergence properties of the two methods and compute autocorrelation functions, we generated 10^7 samples with MYULA and $10^7/s$ samples using SK-ROCK ($s = 15$). We then used these samples to determine the fastest and slowest components of each chain[3] and measured their autocorrelation functions. We also computed trace plots for

[3]The chain's slowest (fastest) component was identified by computing the approximated singular value decomposition of the chain's covariance matrix and choosing on the samples the component with the largest (smallest) singular value.



(a) MYULA: posterior mean

(b) SK-ROCK: posterior mean

(c) MYULA: std. deviation

(d) SK-ROCK: std. deviation

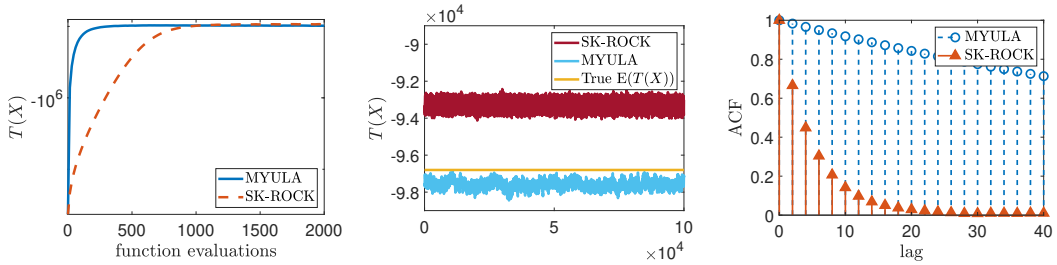
Figure 4.6: `Cameraman` deblurring experiment: (a) Mean of 10^3 samples generated by MYULA and (b) mean of $10^3/s$ samples generated by SK-ROCK (with $s = 15$). (c) Standard deviation of the samples generated by MYULA and (d) SK-ROCK (with $s = 15$).

the chains by using $T(x) = \log \pi_\lambda(x|y)$ as a scalar statistic, which is particularly interesting because it determines the typical set of $x|y$ [94]. These trace plots clearly illustrate how the methods behave during their transient regime, and then how they behave once the chain has converged to equilibrium (i.e., reached steady state).

Figure 4.7(a) shows the convergence of the Markov chains to the typical set $\{x : T(x) \approx \mathrm{E}[T(x)|y]\}$. Moreover, Figure 4.7(b) shows the last 10^5 samples of the chains (again with a 1-in-s thinning for MYULA). Additionally, we have included the summary statistic $\mathbb{E}(T(X))$ calculated by a very long run of the P-MALA algorithm [95], which targets (4.10) exactly, in order to study the bias of the methods[4]. We can see that, for this experiment, the bias of SK-ROCK is slightly increased in comparison to MYULA, however, it has significantly better mixing properties that result in better exploration of the typical set. Lastly, the superior convergence properties of SK-ROCK are also clearly illustrated by the autocorrelation plots of Figure 4.7(c), which show the autocorrelation functions for the slowest components of the chains, and where again we observe a dramatic improvement in decay rate (we have again used a 1-in-s thinning for MYULA for a fair comparison). Table 4.3 reports the associated ESS values for this experiment, where we note that SK-ROCK with $s = 15$ outperforms MYULA by a factor of 21.77 in terms of computational efficiency for the slowest component.

We conclude this experiment by comparing the two methods in terms of estimation of the MSE against the true image. Figure 4.8 shows the evolution of the estimation error for the MMSE solution, as estimated by MYULA and SK-ROCK, and as a function of the number of gradient and proximal operator evaluations. Again, observe that the acceleration properties of SK-ROCK lead to a dramatic improvement in convergence speed, and consequently to a significantly more accurate computation of the MMSE estimator for a given computational budget.

[4]The statistics $T(x) = \log p(x|y)$ is very useful for analysing the bias of high-dimensional log-concave distributions because these concentrate sharply on the typical set $T(x) \approx E(T(X))$ [94].



(a) $\log \pi^{\lambda}(x)$ (b) $\log \pi^{\lambda}(x)$ (c) ACF slow component

Figure 4.7: `Cameraman` deblurring experiment: (a) Convergence to the typical set of the posterior distribution (4.10) for the first 2×10^3 MYULA samples and the first $2 \times 10^3/s$ SK-ROCK ($s = 15$) samples. (b) Last 10^5 values of $\log \pi(x)$. (c) Autocorrelation function for the slowest component.

Table 4.3: `Cameraman` experiment: Summary of the results after generating 10^7 samples with MYULA and $10^7/s$ samples with SK-ROCK with $s = 15$. Computing time 35 hours per method.

Method	**Step-size** δ	**ESS slow com.**	**ESS fast com.**	**Speed-up slow com.**	**Speed-up fast com.**
MYULA	0.106	2.88×10^3	1.00×10^6	-	-
SK-ROCK ($s = 10$)	14.65	4.00×10^4	2.63×10^4	13.89	2.63×10^{-2}
SK-ROCK ($s = 15$)	34.30	6.27×10^4	6.92×10^4	21.77	6.92×10^{-2}

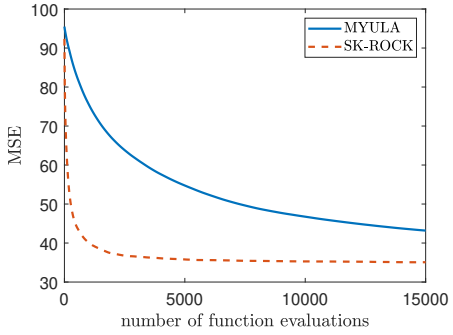


Figure 4.8: `Cameraman` experiment: Mean squared error (MSE) between the mean of the algorithms and the true image, using 15×10^3 samples from MYULA and $15 \times 10^3/s$ samples from SK-ROCK ($s = 15$), after burn-in.

4.5.3 Hyperspectral Unmixing

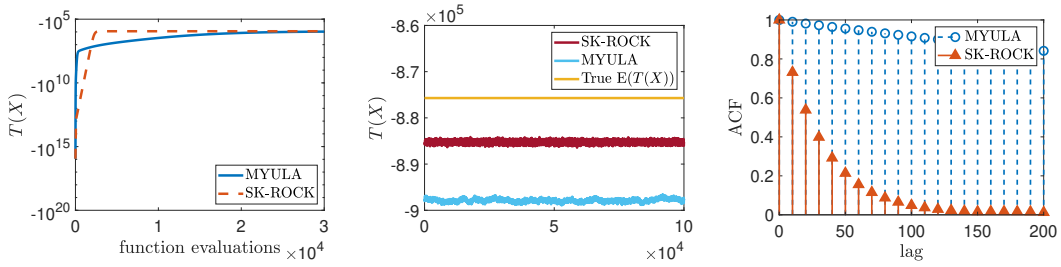
We now present an application to hyperspectral unmixing, previously described in Section 2.1.4. Here we consider the widely used linear mixing model $y = \mathrm{A}x + w$, where $\mathrm{A} = \{a_1, \ldots, a_k\} \in \mathbb{R}^{m \times k}$ is a spectral library gathering the spectral responses of the materials, $x \in \mathbb{R}^{k \times d}$ gathers the abundance maps, and $w \sim \mathbb{N}(0, \sigma^2 \mathbb{I}_{m \times d})$ is additive Gaussian noise. Moreover, following [75], we expect x to be sparse since most image pixels contain only a subset of the materials. Also, we expect materials to exhibit some degree of spatial coherence and regularity. In order to promote solutions with these characteristics, we use the ℓ_1-TV prior proposed in [75] for this type of problem

$$p(x) \propto \exp\{-\beta_1 \|x\|_1 - \beta_2 \mathrm{TV}(x)\} \mathbf{1}_{\mathbb{R}^n_+}(x),$$

where $\beta_1 > 0$ and $\beta_2 > 0$ are hyper-parameters that we assume fixed (in this experiment we use $\beta_1 = 25$ and $\beta_2 = 185$, determined using the method of [123]). The resulting posterior distribution is given by

$$p(x|y) \propto \exp\left[-\|y - \mathrm{A}x\|^2 / 2\sigma^2 - \beta_1 \|x\|_1 - \beta_2 \mathrm{TV}(x)\right] \mathbf{1}_{\mathbb{R}^n_+}(x). \tag{4.11}$$

Figure 2.4 presents an experiment with a synthetic dataset described in [75, Section IV-A] of size $n = 75 \times 75 = 5625$, with 5 materials, and noise amplitude $\sigma = 8.4 \times 10^{-4}$ related to a signal-to-noise-ratio of 40dB. Figure 4.10(a) presents the evolution of the estimation MSE between the true abundance maps and the posterior mean as estimated by MYULA and SK-ROCK (with $s = 15$), and as a function of the number of gradient and proximal operator evaluations (using $\lambda = 7.08 \times 10^{-7}$ which is in the order of L_f^{-1}, as it is recommended in [45, Section 3.3]). As in previous experiments, observe that the posterior means estimated with SK-ROCK converge dramatically faster than the ones calculated with MYULA, clearly exhibiting the benefits of the proposed methodology.



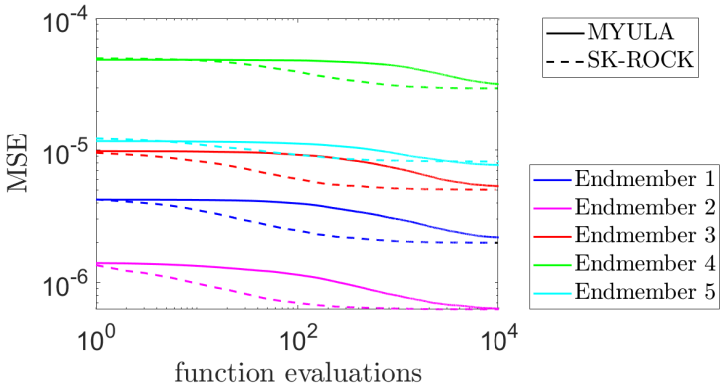
(a) $\log \pi^\lambda(x)$ (b) $\log \pi^\lambda(x)$ (c) ACF slow component

Figure 4.9: `Hyperspectral` experiment: (a) Convergence to the typical set of the posterior distribution (4.11) for the first 3×10^4 MYULA samples and the first $3 \times 10^4/s$ SK-ROCK ($s = 15$) samples. (b) Last 10^5 values of $\log \pi(x)$. (c) Autocorrelation function for the slowest component.

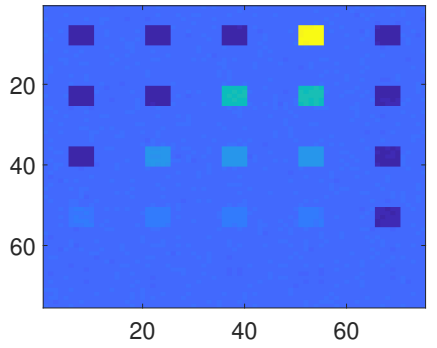
Moreover, for illustration, Figures 4.10(b)-(e) respectively show the estimated abundance maps for the fourth endmember for MYULA (5×10^5 samples) and SK-ROCK ($5 \times 10^5/s$ samples, $s = 15$), as well as the pixel-wise (marginal) standard deviations for the abundances of this material. Again, as in previous experiments, we notice that the estimates obtained with SK-ROCK are noticeably more precise than the ones of MYULA, which would require a larger number of iterations to accurately estimate these second-order statistical moments.

To further compare the convergence properties of the two methods we repeated the experiment and generated 5×10^6 samples with MYULA and $5 \times 10^6/s$ samples with SK-ROCK for $s = 15$ to make the comparisons fair. Figure 4.9(a) presents trace plots for the two chains during their transient regimes using $T(x) = \log p(x|y)$ as summary statistic, as a function of the number of gradient and proximal operator evaluations; observe that SK-ROCK attains the typical set of $x|y$ significantly faster than MYULA, similarly to the previous experiments. Figure 4.9(b) presents similar trace plots for the two chains after burn-in.

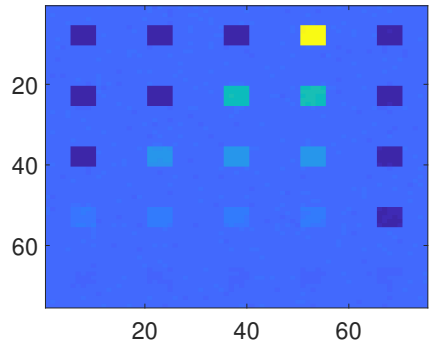
Additionally, as we did in the *cameraman* experiment, we have included the summary



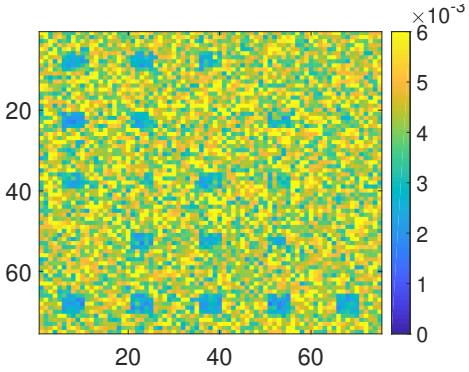
(a) MSE



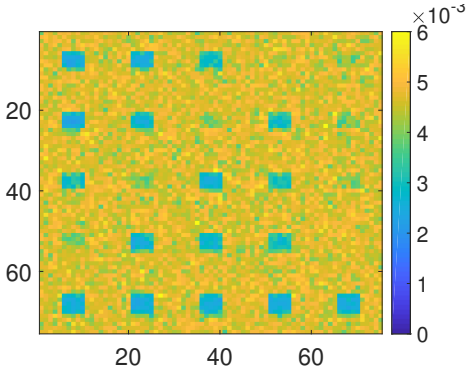
(b) MYULA: posterior mean



(c) SK-ROCK: posterior mean



(d) MYULA: standard deviation



(e) SK-ROCK: standard deviation

Figure 4.10: `Hyperspectral` experiment: (a) Mean squared error (MSE) between the mean of the algorithms and the true image (fractional abundances of endmembers 1 to 5) measured using 10^4 samples from MYULA (solid line) and $10^4/s$ samples from SK-ROCK (dash-dot line, $s = 15$), in logarithmic scale. (b) Posterior mean as estimated with 10^5 samples generated with MYULA and (c) $10^5/s$ samples generated by SK-ROCK (with $s = 15$). (d) Standard deviation of the samples generated by MYULA and (e) SK-ROCK (with $s = 15$).

Table 4.4: `Hyperspectral` experiment: Summary of the results after generating 5×10^6 samples with MYULA and $5 \times 10^6/s$ samples with SK-ROCK. Computing time 88 hours per method.

Method	**Step-size** δ	**ESS slow com.**	**ESS fast com.**	**Speed-up slow com.**	**Speed-up fast com.**
MYULA	1.79×10^{-9}	1.50×10^2	0.63×10^4	-	-
SK-ROCK ($s = 10$)	3.11×10^{-7}	2.90×10^3	1.70×10^4	19.33	2.69
SK-ROCK ($s = 15$)	7.28×10^{-7}	5.69×10^3	3.63×10^4	37.93	5.76

statistic $\mathbb{E}(T(X))$ calculated by a very long run of the P-MALA algorithm, which targets (4.11) exactly, in order to study the bias of the methods. As can be seen clearly, SK-ROCK presents a lower bias than MYULA and also exhibits better mixing properties. The good convergence properties of SK-ROCK can be clearly observed in the autocorrelation plots of Figure 4.9(c), which correspond to the slowest components of the chains as determined by their covariance structure, and where we have again applied the 1-in-15 thinning to the MYULA chain for fairness of comparison. Table 4.4 reports the ESS values for this experiment. In particular, observe that SK-ROCK outperforms MYULA by a factor of 37.9 in terms of ESS for the slowest component of the chain, and by a factor of 5.76 for the fastest component.

4.5.4 Tomographic image reconstruction

We conclude this section with a tomographic image reconstruction experiment, previously described in Section 2.1.3. We have selected this problem to illustrate the proposed methodology in a setting where the posterior distribution is strictly log-concave. The lack of strong log-concavity has a clear negative impact on the convergence properties of the continuous-time Langevin SDE (3.5) [44], and also impacts the convergence properties of the MYULA and SK-ROCK approximations.

In tomographic image reconstruction we seek to recover an image $x \in \mathbb{R}^d$ from an observation $y \in \mathbb{C}^p$ related to x by a linear Fourier model $y = \mathrm{A}\mathrm{F}x + \xi$, where F is the discrete Fourier transform operator on $\mathbb{C}^d$, $\mathrm{A} \in \mathbb{C}^{p \times d}$ is a (sparse) tomographic subsampling mask and $\xi \sim N(0, \sigma^2 \mathbb{I}_{2p})$. Typically $d \gg p$, making the estimation problem strongly ill-posed. We address this difficulty by using a total-variation prior to regularise the estimation problem and promote solutions with certain spatial regularity properties. From Bayes' theorem, the posterior $p(x|y)$ is given by:

$$p(x|y) \propto \exp\left[-\|y - \mathrm{A}\mathrm{F}x\|^2/2\sigma^2 - \beta \mathrm{TV}(x)\right], \tag{4.12}$$

with hyper-parameters $\sigma, \beta \in \mathbb{R}^+$ assumed fixed (in this experiment we use $\beta = 10^2$).

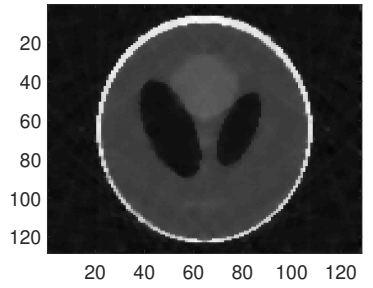
Table 4.5: `Tomography` experiment: Summary of the results after generating 5×10^6 samples with MYULA and $5 \times 10^6/s$ samples with SK-ROCK. Computing time 20 hours per method.

Method	**Step-size** δ	**ESS slow com.**	**ESS fast com.**	**Speed-up slow com.**	**Speed-up fast com.**
MYULA	1.67×10^{-5}	1.31×10^4	1.64×10^5	-	-
SK-ROCK ($s = 5$)	5.02×10^{-4}	5.31×10^4	2.56×10^5	4.05	1.56
SK-ROCK ($s = 10$)	2.30×10^{-3}	2.65×10^5	1.33×10^5	20.23	0.81

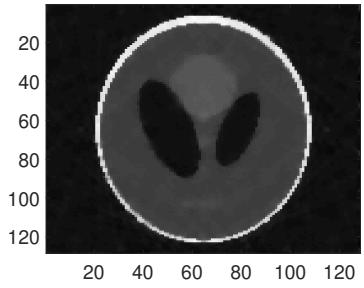
Figure 2.3 presents an experiment with the `Shepp-Logan` phantom test image of size $d = 128 \times 128$ pixels, which we use to generate a noisy observation y by measuring 15% of the original Fourier coefficients, corrupted with additive Gaussian noise with $\sigma = 10^{-2}$ (to improve visibility, Figure 2.3(b) shows the amplitude of the Fourier coefficients in logarithmic scale, unobserved coefficients are depicted in black).

Following on from this, we use MYULA and SK-ROCK with $s = 10$ to generate 10^4 and 10^3 samples respectively from $p(x|y)$ with $\lambda = 0.2 \times 10^{-4}$ which is in the order of L_f^{-1}, as it

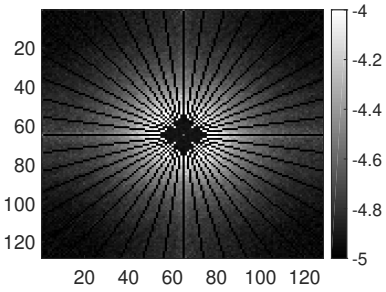
is recommended in [45, Section 3.3]. We then use these samples to compute two quantities: 1) the MMSE estimators - displayed in Figures 4.11(a)-(b); and 2) the (marginal) standard deviations of the amplitude of the Fourier coefficients of $x|y$, depicted in Figures 4.11(c)-(d) in logarithmic scale. Observe that, in this experiment, both methods deliver good and similar results with the number of samples available, with MYULA producing slightly less accurate standard deviation estimates. More interestingly, notice from Figures 4.11(c)-(d) that in this tomographic experiment the uncertainty is concentrated in the unobserved medium frequencies, whereas in the deblurring experiment uncertainty was predominant in the high frequencies.



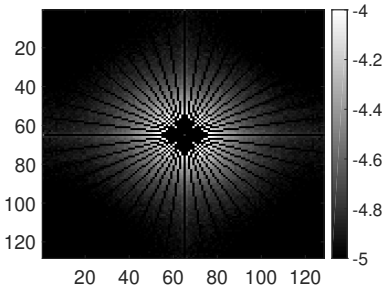
(a) MYULA: posterior mean



(b) SK-ROCK: posterior mean

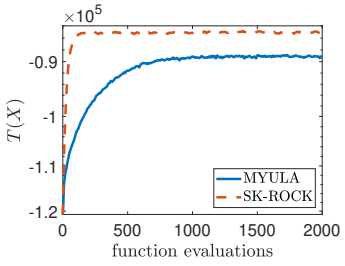


(c) MYULA: std. deviation

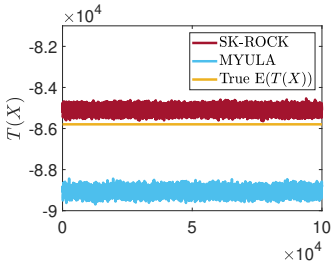


(d) SK-ROCK: std. deviation

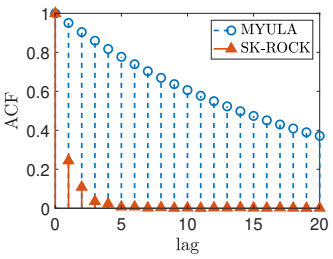
Figure 4.11: `Tomography` experiment: Posterior mean of $x|y$ as estimated with (a) MYULA (10^4 samples) and (b) SK-ROCK (10^3 samples, $s = 10$). Standard deviations of the amplitude of the Fourier coefficients of $x|y$ as estimated with (c) MYULA (10^4 samples) and (d) SK-ROCK (10^3 samples, $s = 10$).



(a) $\log \pi^{\lambda}(x)$



(b) $\log \pi^{\lambda}(x)$



(c) ACF slow component

Figure 4.12: `Tomography` experiment: (a) Convergence to the typical set of the posterior distribution (4.12) for the first 3×10^4 MYULA samples and the first $3 \times 10^4/s$ SK-ROCK ($s = 10$). (b) Last 10^5 values of $\log \pi(x)$ from MYULA and SK-ROCK ($s = 10$) chains. (c) Autocorrelation function for the slowest component

Moreover, to analyse the convergence properties of the two methods we compute autocorrelation functions by generating 5×10^6 samples with MYULA and $5 \times 10^6/s$ samples using SK-ROCK with $s = 10$. We use said samples to determine the fastest and slowest components of each chain and measure their autocorrelation functions. Table 4.5 reports the associated

ESS, which shows that the SK-ROCK outperform MYULA by a factor of 20.23 in terms of ESS for the slowest component of the chain. These superior convergence properties can be clearly observed in Figure 4.12(c), which presents the autocorrelation plots for the slowest components of the chains. For completeness, Table 4.5 also reports the values obtained with SK-ROCK with $s = 5$.

Finally, as in previous experiments, Figure 4.12(a) presents trace plots for the two chains during their burn-in stages; we can see that SK-ROCK reaches the typical set of $x|y$ significantly faster than MYULA. Figure 4.12(b) shows the $\log \pi(x)$ trace of both methods after burn-in, and similarly to the *cameraman* and *hyperspectral* experiments we have also included the entropy $\mathbb{E}(T(X))$ of the distribution calculated by a very long run of the P-MALA algorithm, which targets (4.12) exactly. As can be seen, SK-ROCK presents a lower bias than MYULA.

Chapter 5

When SK-ROCK meets the split Gibbs sampler

The results presented in this Chapter are currently under review [99] and are joint work with Marcelo Pereyra and Konstantinos C. Zygalakis.

5.1 Introduction

This chapter explores two recent and different approaches in order to accelerate the convergence of proximal MCMC algorithms: the proximal stochastic Runge-Kutta-Chebyshev method (SK-ROCK) [2, 97] introduced in the previous chapter, which carefully combines s gradient evaluations to achieve an s^2-increase in the step-size, and the Split Gibbs Sampler (SGS) [125, 129], which is based on an augmentation and relaxation scheme that can significantly improve convergence speed at the expense of some estimation bias.

Specifically, we investigate two natural questions: first, how SGS and SK-ROCK compare methodologically and empirically, and second, whether the two methods can be combined in order to yield even more efficient MCMC methods. We address these questions in the following way:

1. Rather than viewing the model augmentation and relaxation strategy of [100, 125, 129] as an approximation, we propose to regard the augmented model as a generalisation of the original model. We show empirically that there is a range of relaxation values for which the accuracy of the model improves. In this regime, relaxation leads to better convergence properties and better accuracy. Beyond this regime, the accuracy of the relaxed model deteriorates rapidly.

2. Given the critical role of the amount of relaxation, we build on [123] to propose an empirical Bayesian method to automatically estimate the value of the relaxation parameter by maximum marginal likelihood estimation.

3. We formally identify a relationship between SGS and MYULA by re-expressing SGS as a discrete-time approximation of a Langevin stochastic differential equation (SDE) closely related to MYULA.

4. Having connected SGS and MYULA at the level of the SDE, we propose two novel MCMC methods for Bayesian imaging: 1) an integration of SGS and MYULA that improves on both SGS and MYULA; and 2) an integration of SGS and SK-ROCK that outperform SK-ROCK, the previously fastest method in the literature.

5.2 The split Gibbs sampler

A separate line of research seeks to address the limitation of MYULA by introducing an auxiliary variable $z \in \mathbb{R}^d$ to operate on an augmented state-space (x, z) and relaxing the original model

(2.7) by defining the following augmented posterior

$$p(x, z|y, \beta, \rho^2) \propto \exp\left[-f_y(x) - \beta^{\mathsf{T}} g(z) - \frac{1}{2\rho^2}\|x - z\|^2\right], \quad \rho^2 > 0, \tag{5.1}$$

where ρ^2 controls the correlation between the variable of interest x and the auxiliary variable z, and f_y, g are the same as in Chapter 2. This approach was first introduced several decades ago as a way to calculate maximum likelihood estimates from incomplete data [37], and as an efficient method for sampling from posterior distributions [114] (see [47] for a review of these techniques). In the current literature, this model was revisited by [100] in the context of consensus Monte Carlo in distributed settings and applied to imagining inverse problems in [125], where its similarities to the algorithmic structure of the Alternating Direction Method of Multipliers (ADMM) optimization algorithm [20] were also discussed.

If we now consider the marginal posterior distribution

$$p(x|y, \beta, \rho^2) = \int_{\mathbb{R}^d} p(x, z|y, \beta, \rho^2)\, \mathrm{d}z, \tag{5.2}$$

it is possible to show that it converges in total variation to the original posterior $p(x|y, \beta)$ as $\rho^2 \to 0$. From a computational point of view, as in the case of MYULA, because g is not differentiable one needs to approximate $p(x, z|y, \beta, \rho^2)$ by

$$p^\lambda(x, z|y, \beta, \rho^2) \propto \exp\left[-f_y(x) - \beta^{\mathsf{T}} g^\lambda(z) - \frac{1}{2\rho^2}\|x - z\|^2\right], \quad \rho^2 > 0. \tag{5.3}$$

To sample (5.3), [125, 129] proposed a splitting strategy based on the Gibbs sampling, applied to the following conditional distributions

$$p(x|y, z, \rho^2) \propto \exp\left[-f_y(x) - \frac{1}{2\rho^2}\|x - z\|^2\right], \tag{5.4}$$

$$p^\lambda(z|x, \beta, \rho^2) \propto \exp\left[-\beta^{\mathsf{T}} g^\lambda(z) - \frac{1}{2\rho^2}\|x - z\|^2\right]. \tag{5.5}$$

This method is known as the split Gibbs sampler (SGS). See Algorithm 7.

Algorithm 7 SGS

Require: : $X_0, Z_0 \in \mathbb{R}^d$, $\lambda, \rho^2 > 0$, $n \in \mathbb{N}$.
  for $i = 0 : n - 1$ **do**
    Sample $X_{i+1} \sim p\left(x|y, Z_i, \rho^2\right)$ according to (5.4),
    Compute $Z_{i+1} = Z_i - \delta \sum_{k=1}^{p} [Z_i - \mathrm{prox}^{\lambda}_{\beta_k g_k}(Z_i)]/\lambda - \delta(Z_i - X_{i+1})/\rho^2 + \sqrt{2\delta}\zeta_{i+1}$; where $\zeta_{i+1} \sim \mathcal{N}(0, \mathbb{I}_d)$,
  end for
  Output: Samples $X_1, \ldots, X_n$.

In the case where the likelihood is Gaussian one can exactly sample from (5.4) [57] (for a review and comparison of existing Gaussian sampling approaches, see [127]). A main benefit of this splitting approach is that the step-size one needs to set for the proximal MCMC method used for sampling (5.5) will be independent of the Lipschitz constant associated with the likelihood distribution, and will only depend on the parameters λ and ρ^2. This can lead to faster sampling algorithms compared to MYULA for suitably chosen values of the parameter ρ^2 [125] but for a different posterior distribution.

5.3 Enhancing Bayesian imaging models by smoothing

As discussed previously, the augmented model (5.1) was originally proposed as a relaxation of (2.7) that allows for a faster exploration of the target distribution, at the expense of some

additional bias when compared to the original model. One then might think that $\rho^2 = 0$ represents the best model for inference (at the expense of higher computing cost). However, we have found empirically that this is not the case.

As an illustration, Figure 5.1(a) shows the estimation mean-squared error (MSE) for a Bayesian image deblurring problem (the details of this experiment will be explained in Section 5.5.1). The error is computed w.r.t. the posterior mean, as estimated by an adaptation of the SK-ROCK method to target (5.3) (see Section 5.4.2 for details), using a value of $\beta = 4.4 \times 10^{-2}$ estimated by [123, Algorithm 1], and by using different values for ρ^2. Recalling that increasing ρ^2 improves convergence speed, one can clearly identify a regime of small values of ρ^2 for which convergence speed improves without deterioration in estimation accuracy (in fact, there is a mild improvement). Beyond this range, the estimation MSE deteriorates dramatically. This suggests the need for a method to automatically set the value of ρ^2.

We propose an empirical Bayesian method to estimate optimal values for β and ρ^2 directly from y by maximum marginal likelihood estimation (MMLE)

$$(\beta_*, \rho_*^2) = \underset{\beta \in \mathrm{B}, \rho^2 \in \mathrm{R}}{\operatorname{argmax}} \; p(y|\beta, \rho^2), \tag{5.6}$$

where $\mathrm{B} \subset (0, +\infty)^p$ and $\mathrm{R} \subset (0, +\infty)$ are compact convex sets. To solve (5.6) we modify the stochastic approximation proximal gradient (SAPG) algorithm of [123]. Notice that by maximising the model evidence, (5.6) seeks to select the best model to perform inference within the class of posterior distributions parametrised by $\beta \in \mathrm{B}, \rho^2 \in \mathrm{R}$ [133].

5.3.1 Computing the optimal values for β and ρ^2

We adopt the approach of [123] to solve (5.6) and estimate optimal values for β and ρ^2 in (5.1). The method [123] was proposed for models of the form (2.7), so we will now adapt it to the augmented model (5.1).

We are interested in estimating the parameters $\beta \in \mathrm{B}$, $\rho^2 \in \mathrm{R}$ by MMLE (5.6), where $p(y|\beta, \rho^2)$ is defined for (5.1) as

$$p(y|\beta, \rho^2) = \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} p(y|x) p(x, z|\beta, \rho^2) \mathrm{d}x \mathrm{d}z = \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} p(y|x) p(x|z, \rho^2) p(z|\beta) \mathrm{d}x \mathrm{d}z,$$

with

$$p(y|x) = \frac{\exp(-f_y(x))}{\int_{\mathbb{R}^d} \exp(-f_y(x)) \mathrm{d}x}, \quad p(z|\beta) = \frac{\exp(-\beta^\mathsf{T} g(z))}{\int_{\mathbb{R}^d} \exp(-\beta^\mathsf{T} g(z)) \mathrm{d}z} \tag{5.7}$$

and

$$p(x|z, \rho^2) = \frac{\exp(-\|x - z\|^2 / 2\rho^2)}{\int_{\mathbb{R}^d} \exp(-\|x - z\|^2 / 2\rho^2) \mathrm{d}x} = \frac{\exp(-\|x - z\|^2 / 2\rho^2)}{(2\pi\rho^2)^{d/2}}. \tag{5.8}$$

If we had access to the gradients $\nabla_{\rho^2} \log p(y|\beta, \rho^2)$ and $\nabla_\beta \log p(y|\beta, \rho^2)$, then we could construct an iterative algorithm that converges to the solution of (5.6) by using the projected gradient algorithm [81]

$$\begin{aligned} \rho_{n+1}^2 &= \Pi_{\mathrm{R}} \left[\rho_n^2 + \gamma_n \nabla_{\rho^2} \log p(y|\beta_n, \rho_n^2) \right] \\ \beta_{n+1} &= \Pi_{\mathrm{B}} \left[\beta_n + \gamma_n \nabla_\beta \log p(y|\beta_n, \rho_n^2) \right], \end{aligned}$$

where Π_{R} and Π_{B} are the projection onto R and B respectively, and $(\gamma_n)_{n \in \mathbb{N}}$ is a sequence of non-increasing step-sizes. However, due to the complexity of the model, $\nabla_{\rho^2} \log p(y|\beta, \rho^2)$ and $\nabla_\beta \log p(y|\beta, \rho^2)$ are intractable.

As shown in [123], one can construct carefully designed stochastic estimates of these gradients that satisfy the conditions for the solution to converge to (5.6). To build these stochastic estimators, we are going to express the gradients as expectations by applying Fisher's identity [39, Proposition D.4] which we can then approximate using MCMC. We will see that in fact, one MCMC sample will suffice to obtain an estimate of the gradient accurate enough to converge

asymptotically to (5.6). More precisely, we have that

$$\nabla_{\rho^2} \log p(y|\beta,\rho^2) = \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} p(x,z|y,\beta,\rho^2)\nabla_{\rho^2}\log p(x,z,y|\beta,\rho^2)\mathrm{d}x\mathrm{d}z,$$

and

$$\nabla_{\beta} \log p(y|\beta,\rho^2) = \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} p(x,z|y,\beta,\rho^2)\nabla_{\beta}\log p(x,z,y|\beta,\rho^2)\mathrm{d}x\mathrm{d}z.$$

As $p(x,z,y|\beta,\rho^2) = p(y|x)p(x,z|\beta,\rho^2) = p(y|x)p(x|z,\rho^2)p(z|\beta)$, we have

$$\nabla_{\rho^2} \log p(y|\beta,\rho^2) = \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} p(x,z|y,\beta,\rho^2)\nabla_{\rho^2}\log p(x|z,\rho^2)\mathrm{d}x\mathrm{d}z,$$

and

$$\nabla_{\beta} \log p(y|\beta,\rho^2) = \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} p(x,z|y,\beta,\rho^2)\nabla_{\beta}\log p(z|\beta)\mathrm{d}x\mathrm{d}z.$$

Replacing (5.8) in $p(x|z,\rho^2)$ we obtain

$$\nabla_{\rho^2} \log p(y|\beta,\rho^2) = A_{\beta,\rho^2}(y) - \frac{d}{2\rho^2},$$

where

$$A_{\beta,\rho^2}(y) = \mathbb{E}_{x,z|y,\beta,\rho^2}\left[\frac{\|x-z\|^2}{2(\rho^2)^2}\right] = \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} p(x,z|y,\beta,\rho^2)\frac{\|x-z\|^2}{2(\rho^2)^2}\mathrm{d}x\mathrm{d}z,$$

and similarly, replacing (5.7) in $p(z|\beta)$ gives

$$\nabla_{\beta} \log p(y|\beta,\rho^2) = -B_{\beta,\rho^2}(y) - C_{\beta,\rho^2}(y),$$

where

$$\begin{aligned}
B_{\beta,\rho^2}(y) &= \mathbb{E}_{x,z|y,\beta,\rho^2}[g(z)] = \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} p(x,z|y,\beta,\rho^2)g(z)\mathrm{d}x\mathrm{d}z,\\
C_{\beta,\rho^2}(y) &= \mathbb{E}_{x,z|y,\beta,\rho^2}\left[\nabla_{\beta}\log\left(\int_{\mathbb{R}^d}\exp(-\beta^{\mathsf{T}}g(z))\mathrm{d}z\right)\right]\\
&= \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} p(x,z|y,\beta,\rho^2)\nabla_{\beta}\log\left[\int_{\mathbb{R}^d}\exp(-\beta^{\mathsf{T}}g(z))\mathrm{d}z\right]\mathrm{d}x\mathrm{d}z.
\end{aligned}$$

Because of the complexity of the model, $A_{\beta,\rho^2}(y)$ and $B_{\beta,\rho^2}(y)$ are not available analytically and need to be approximated by MCMC computation (e.g., by using the methods we develop in Section 5.4). With respect to $C_{\beta,\rho^2}(y)$, and more precisely, the integral between brackets, we can follow a similar procedure as in [123, Section 3.2.1]. In particular, if we consider the case where each $g_i(z)$ is α_i positively homogeneous[1], which is the case for many regularisers such as ℓ_1, ℓ_2 or TV, we have that

$$\frac{\partial \log p(y|\beta,\rho^2)}{\partial \beta^{(i)}} = \frac{d}{\alpha_i\beta^{(i)}} - \mathbb{E}_{x,z|y,\beta^{(i)},\rho^2}\left[g_i(z)\right].$$

(See [123] for more details and [123, Section 3.2] for the case of inhomogeneous regularisers).

Following on from this, and by using Monte Carlo approximations of $A_{\beta,\rho^2}(y)$ and $B_{\beta,\rho^2}(y)$, we construct an SAPG algorithm [53, 123] to solve (5.6) and produce optimal estimates of β and ρ^2. This method is presented in Algorithm 8. We refer the reader to [36, 35] for details about the convergence properties of this kind of SAPG algorithm. To illustrate Algorithm 8 in action, Figure 5.1 shows the value of ρ^2 estimated by the algorithm for the image deblurring problem. Observe that the MMLE estimate is close to the value that produces the best estimation MSE in this case. This is in agreement with the results reported in [123] for other problems.

[1] $g(x)$ is α positively homogeneous if, for any $x \in \mathbb{R}^d$ and $t > 0$, $g(tx) = t^{\alpha}g(x)$.

Algorithm 8 SAPG algorithm for the augmented model (5.1)

Input: $X_0^0, Z_0^0 \in \mathbb{R}^d$, $\beta_0, \rho_0^2, \gamma_0, \gamma_0' \in \mathbb{R}$, $\lambda > 0$, $m, n \in \mathbb{N}$.
for $i = 0 : m-1$ **do**
  if $i > 0$ **then**
    Set $X_i^0 = X_{i-1}^{n-1}$,
  end if
  for $j = 0 : n-1$ **do**
    Sample $X_{i+1}^{(j+1)}$, $Z_{i+1}^{(j+1)}$ according to Algorithm 9,
  end for
  for $j = 1 : p$ **do**
    Set $\beta_{i+1}^{(j)} = \Pi_{\mathrm{B}}\left[\beta_i^{(j)} + \frac{\gamma_{i+1}}{n}\sum_{k=1}^n \left\{\frac{d}{\alpha_j \beta_i^{(j)}} - g_j(Z_{i+i}^k)\right\}\right]$,
  end for
  Set $\rho_{i+1}^2 = \Pi_{\mathrm{R}}\left[\rho_i^2 + \frac{\gamma_{i+1}'}{n}\sum_{k=1}^n \left\{\|X_{i+1}^k - Z_{i+1}^k\|^2/2(\rho_i^2)^2 - d/(2\rho_i^2)\right\}\right]$,
end for
Output: $\overline{\beta}_m^{(j)} = \sum_{k=0}^{m-1}\omega_k \beta_k^{(j)} / \sum_{k=0}^{m-1}\omega_k$ for $j \in \{1, \ldots, p\}$, $\overline{\rho}_m^2 = \sum_{k=0}^{m-1}\omega_k \rho_k^2 / \sum_{k=0}^{m-1}\omega_k$.

5.4 Reinterpretation of SGS as noisy MYULA & new MCMC methods

In this section, we proceed to show that the SGS algorithm 7 can be viewed as a noisy version of MYULA. This link will be crucial in allowing us to write it as a noisy discretisation of an SDE, which will help us to propose more efficient MCMC methods for sampling (5.1).

First, note that the marginal of z computed from (5.3) can be written as follows

$$p^\lambda(z|y,\beta,\rho^2) = \int_{\mathbb{R}^d} p^\lambda(x,z|y,\beta,\rho^2)\,\mathrm{d}x \propto p(y|z,\rho^2)p^\lambda(z|\beta),$$

where

$$p(y|z,\rho^2) \propto \int_{\mathbb{R}^d} \exp\left[-f_y(x) - \frac{1}{2\rho^2}\|x-z\|^2\right]\mathrm{d}x^2, \quad p^\lambda(z|\beta) \propto \exp[-\beta^{\mathsf{T}} g^\lambda(z)].$$

Applying the MYULA to $p^\lambda(z|y,\beta,\rho^2)$, we have that

$$Z_{n+1} = Z_n + \delta\nabla_z \log p^\lambda(Z_n|\beta) + \delta\nabla_z \log p(y|Z_n,\rho^2) + \sqrt{2\delta}\zeta_{n+1}, \tag{5.9}$$

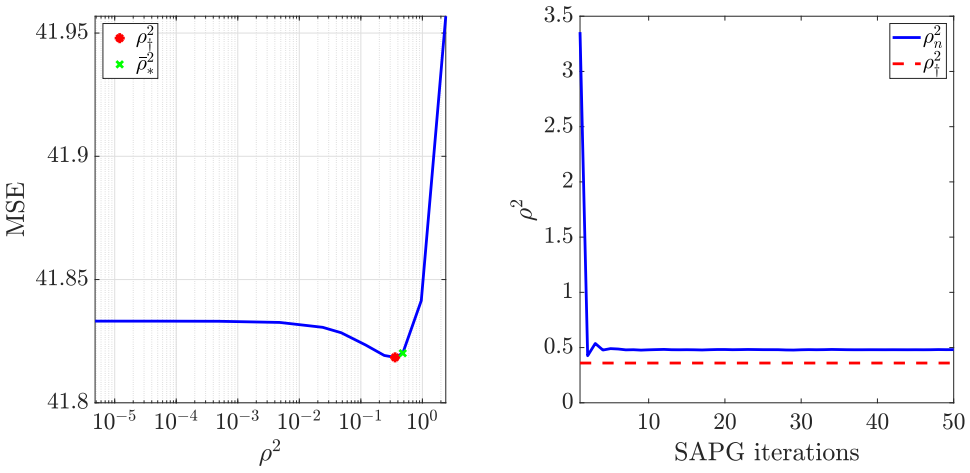
where $(\zeta_{n+1})_{n\geq 0}$ is an i.i.d. sequence of d-dimensional standard Gaussian random vectors. Due to the complexity of the model, it is difficult to compute $\nabla_z \log p(y|Z_n,\rho^2)$, however, we can express it as an expectation by using Fisher's identity [39, Proposition D.4] as follows

$$\begin{aligned}\nabla_z \log p(y|z,\rho^2) &= \int_{\mathbb{R}^d} p(x|y,z,\rho^2)\nabla_z \log p(x,y|z,\rho^2)\mathrm{d}x \\ &= \mathbb{E}_{x|y,z,\rho^2}\left[\nabla_z \log p(x,y|z,\rho^2)\right].\end{aligned}$$

As $p(x,y|z,\rho^2) = p(y|x)p(x|z,\rho^2)$, we have

$$\begin{aligned}\nabla_z \log p(x,y|z,\rho^2) &= \mathbb{E}_{x|y,z,\rho^2}\left[\nabla_z \log p(x|z,\rho^2)\right] \\ &= \frac{1}{\rho^2}\mathbb{E}_{x|y,z,\rho^2}\left(x-z\right).\end{aligned}$$

[2] In the case where $f_y(x)$ is quadratic, $p(y|z,\rho^2)$ is Gaussian with eigenvalues in its covariance matrix shifted by ρ^2, when compared with the covariance of $f_y(x)$.



(a) (b)

Figure 5.1: Image deblurring experiment: (a) MSE between the true image and the posterior mean estimated using Algorithm 10, for some values of ρ^2. In red, the optimal value of ρ^2 that minimises the MSE, and in green, the value of ρ^2 found by Algorithm 8. (b) Iterations of SAPG algorithm to estimate ρ^2.

Using this expression in (5.9) we obtain

$$Z_{n+1} = Z_n - \delta \nabla_z p^\lambda(Z_n|\beta) - \frac{\delta}{\rho^2}\mathbb{E}_{x|y,z,\rho^2}\left(Z_n - x\right) + \sqrt{2\delta}\zeta_{n+1}, \tag{5.10}$$

We are now ready to explicitly establish the connection to SGS. SGS stems from dealing with the presence of the expectation in this algorithm by replacing it with a Monte Carlo empirical average, i.e.,

$$\mathbb{E}_{x|y,z,\rho^2}\left(Z_n - x\right) \approx Z_n - \frac{1}{N}\sum_{i=1}^{N} X^{(i)}, \text{ where } X^{(i)} \sim p(x|y, Z_n; \rho). \tag{5.11}$$

More precisely, to recover SGS we take $N = 1$ and substitute in (5.10) to obtain

$$Z_{n+1} = Z_n - \delta \nabla_z p^\lambda(Z_n|\beta) - \frac{\delta}{\rho^2}(Z_n - X^{(1)}) + \sqrt{2\delta}\zeta_{n+1}. \tag{5.12}$$

Since $X^{(1)}$ is an exact sample from $p(x|y, Z_n, \rho^2)$, (5.12) corresponds to the fourth line of Algorithm 7.

This establishes that SGS is equivalent to a noisy version of MYULA that relies on one sample from $p(x|y, z, \rho^2)$ to compute a stochastic estimate of the gradient $\nabla_z \log p(y|z, \rho^2)$ via (5.11). Using multiple samples from $p(x|y, Z_n; \rho)$ would improve the estimation of the expectation (5.11) and hence the behaviour of the algorithm. Alternatively, in the experiments considered in this thesis $p(x|y, Z_n; \rho)$ is Gaussian, and hence this expectation can be calculated exactly. This is exploited in the MCMC methods proposed below.

5.4.1 Latent space MYULA

We established above that SGS is equivalent to MYULA targeting the marginal of z with an inexact (i.e., stochastic) estimate of the gradient. Replacing this stochastic estimate with its

exact value in Algorithm 7 produces the following recursion

$$\begin{aligned} X_{\text{grad}}^{i+1} &= \mathbb{E}_{x|y,Z_i,\rho^2}[x], \\ Z_{i+1} &= Z_i - \frac{\delta}{\lambda}\sum_{k=1}^{p}[Z_i - \text{prox}_{\beta_k g_k}^{\lambda}(Z_i)] - \delta(Z_i - X_{\text{grad}}^{i+1})/\rho^2 + \sqrt{2\delta}\zeta_{i+1}, \end{aligned} \tag{5.13}$$

where $\zeta_{i+1} \sim \mathcal{N}(0, \mathbb{I}_d)$.

We now discuss how to use samples $\{Z_i\}_{i\geq 1}^m$ to compute expectations w.r.t. the marginal of interest $x|y,\beta,\rho^2$. More precisely, consider the computation of an expectation $\mathbb{E}_{x|y,\beta,\rho^2}[h(x)]$ for some function h w.r.t. the posterior distribution $p^\lambda(x|y,\beta,\rho^2)$ defined in (5.2) by using (5.13). Formally,

$$\mathbb{E}_{x|y,\beta,\rho^2}[h(x)] = \int_{\mathbb{R}^d} h(x) \int_{\mathbb{R}^d} p(x,z|y,\beta,\rho^2)\,\mathrm{d}z\,\mathrm{d}x.$$

Using the fact that $p(x,z|y,\beta,\rho^2) = p(x|y,z,\rho^2)p(z|\beta)$ we have that

$$\begin{aligned} \mathbb{E}_{x|y,\beta,\rho^2}[h(x)] &= \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} h(x)p(x|y,z,\rho^2)p(z|\beta)\,\mathrm{d}z\,\mathrm{d}x \\ &= \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} h(x)p(x|y,z,\rho^2)\,\mathrm{d}x\,p(z|\beta)\,\mathrm{d}z \\ &= \mathbb{E}_{z|\beta}\left[\mathbb{E}_{x|y,z,\rho^2}\left(h(x)\right)\right]. \end{aligned} \tag{5.14}$$

In cases where $\mathbb{E}_{x|y,\beta,\rho^2}[h(x)]$ is available analytically, we suggest using a Rao-Blackwellised estimator of the form [103]

$$\mathbb{E}_{x|y,\beta,\rho^2}[h(x)] \approx \frac{1}{m}\sum_{i=1}^{m}\mathbb{E}_{x|y,Z_i,\rho^2}\left[h(x)\right].$$

The computation of $\mathbb{E}_{x|y,Z_i,\rho^2}\left[h(x)\right]$ can be done as a postprocessing step, or alternatively within the iterations of the sampler. If $\mathbb{E}_{x|y,\beta,\rho^2}[h(x)]$ is not available analytically, we would draw samples from the conditional $x|y,Z_i,\beta,\rho^2$ and apply a standard Monte Carlo estimator.

We are now ready to present our first new MCMC method, summarised in Algorithm 9 below. We henceforth refer to this method as *latent space MYULA* (ls-MYULA), since it corresponds to MYULA applied to the marginal of the latent variable z.

Algorithm 9 ls-MYULA

1: Input: $X_0, Z_0 \in \mathbb{R}^d$, $\lambda, \rho > 0$, $m \in \mathbb{N}$.
2: **for** $i = 0 : m-1$ **do**
3: Compute $X_{\text{grad}}^{i+1} = \mathbb{E}_{x|y,Z_i,\rho^2}[x]$,
4: Compute $Z_{i+1} = Z_i - \delta\sum_{k=1}^{p}[Z_i - \text{prox}_{\beta_k g_k}^{\lambda}(Z_i)]/\lambda - \delta(Z_i - X_{\text{grad}}^{i+1})/\rho^2 + \sqrt{2\delta}\zeta_{i+1}$; where $\zeta_{i+1} \sim \mathcal{N}(0,\mathbb{I}_d)$,
5: Compute $\hat{h}_{i+1} = \mathbb{E}_{x|y,Z_{i+1},\rho^2}[h(x)]$,
6: **end for**
7: Output: an estimator of $\mathbb{E}_{x|y,\beta,\rho^2}[h(x)]$ given by $\{\sum_{k=1}^{m}\hat{h}_k\}/m$.

Remark 1. *The underlying assumption in Algorithm 9 is that one can explicitly calculate* $\mathbb{E}_{x|y,Z_i,\rho^2}[x]$ *which, for example, is the case when the expectation represents the first moment of a Gaussian distribution, which corresponds to the likelihood models we consider in our experiments. In cases where* $\mathbb{E}_{x|y,Z_i,\rho^2}[x]$ *is intractable, we recommend to replace the expectation by its corresponding MCMC estimation, i.e.,*

$$\mathbb{E}_{x|y,Z_i,\rho^2}[x] \approx \frac{1}{M}\sum_{i=1}^{M} X_i, \text{ where } X_i \sim p(x|y,Z_i,\rho^2).$$

5.4.2 Latent space SK-ROCK

In the same way that an exact MYULA discretization is more beneficial than the stochastic MYULA discretization used in SGS, we can further improve results by using an exact SK-ROCK discretization which, as we described in Chapter 4, has many important advantages compared to MYULA. In particular, we present this method in Algorithm 10, and we will refer to it as *latent space SK-ROCK* (ls-SK-ROCK). The main difference between this algorithm and Algorithm 6 is that the conditional expectation $\mathbb{E}_{x|y,\tilde{\mathrm{Z}}_j,\rho^2}[x]$ is computed on each internal stage s.

Algorithm 10 ls-SK-ROCK

Input: $X_0, Z_0 \in \mathbb{R}^d$, $\lambda, \rho > 0$, $m, s \in \mathbb{N}$, $\eta = 0.05$.
Compute $l_s = (s-0.5)^2(2-4/3\eta) - 1.5$,
Compute $\omega_0 = 1 + \eta/s^2$, $\omega_1 = T_s(\omega_0)/T_s'(\omega_0)$,
Compute $\mu_1 = \omega_1/\omega_0$, $\nu_1 = s\omega_1/2$, $k_1 = s\omega_1/\omega_0$,
Choose $\delta \in (0, \delta_s^{\max}]$, where $\delta_s^{\max} = l_s/(1/(\rho^2 + L_f^{-1}) + 1/\lambda)$,
for $i = 0 : m-1$ **do**
    Set $\tilde{\mathrm{X}}_{\mathrm{grad}}^0 = X_{\mathrm{grad}}^i$, $\tilde{\mathrm{Z}}_0 = Z_i$,
    Sample $\xi_{i+1} \sim \mathcal{N}(0, 2\delta\mathbb{I}_d)$,
    Compute $\tilde{\mathrm{X}}_{\mathrm{grad}}^1 = \mathbb{E}_{x|y,\tilde{\mathrm{Z}}_0+\nu_1\xi_{i+1},\rho^2}[x]$,
    Compute $\Lambda(\tilde{\mathrm{Z}}_0) = \sum_{k=1}^p [\tilde{\mathrm{Z}}_0+\nu_1\xi_{i+1} - \mathrm{prox}_{\beta_k g_k}^{\lambda}(\tilde{\mathrm{Z}}_0+\nu_1\xi_{i+1})]/\lambda + (\tilde{\mathrm{Z}}_0+\nu_1\xi_{i+1} - \tilde{\mathrm{X}}_{\mathrm{grad}}^1)/\rho^2$,
    Compute $\tilde{\mathrm{Z}}_1 = \tilde{\mathrm{Z}}_0 - \mu_1\delta\Lambda(\tilde{\mathrm{Z}}_0) + k_1^2\xi_{i+1}$,
    for $j = 2 : s$ **do**
        Compute $\mu_j = 2\omega_1 T_{j-1}(\omega_0)/T_j(\omega_0)$, $\nu_j = 2\omega_0 T_{j-1}(\omega_0)/T_j(\omega_0)$, $k_j = 1 - \nu_j$,
        Compute $\tilde{\mathrm{X}}_{\mathrm{grad}}^j = \mathbb{E}_{x|y,\tilde{\mathrm{Z}}_{j-1},\rho^2}[x]$,
        Compute $\Lambda(\tilde{\mathrm{Z}}_{j-1}) = \sum_{k=1}^p [\tilde{\mathrm{Z}}_{j-1} - \mathrm{prox}_{\beta_k g_k}^{\lambda}(\tilde{\mathrm{Z}}_{j-1})]/\lambda + (\tilde{\mathrm{Z}}_{j-1} - \tilde{\mathrm{X}}_{\mathrm{grad}}^j)/\rho^2$,
        Compute $\tilde{\mathrm{Z}}_j = -\mu_j\delta\Lambda(\tilde{\mathrm{Z}}_{j-1}) + \nu_j\tilde{\mathrm{Z}}_{j-1} + k_j\tilde{\mathrm{Z}}_{j-2}$,
    end for
    Set $X_{\mathrm{grad}}^{i+1} = \tilde{\mathrm{X}}_{\mathrm{grad}}^s$, $Z_{i+1} = \tilde{\mathrm{Z}}_s$, $\hat{h}_{i+1} = \mathbb{E}_{x|y,Z_{i+1},\rho^2}[h(x)]$,
end for
Output: an estimator of $\mathbb{E}_{x|y,\beta,\rho^2}[h(x)]$ given by $\{\sum_{k=1}^m \hat{h}_k\}/m$.

5.4.3 Implementation guidelines

Setting λ

As the priors of the experiments performed in this work are non-differentiable, we will use the Moreau-Yosida envelope defined in (3.9) with $\lambda \in [L_f^{-1}, 10L_f^{-1}]$. We chose $\lambda = L_f^{-1}$ in our numerical experiments, however, we have found numerically that values of $\lambda = 5L_f^{-1}$ or $\lambda = 10L_f^{-1}$ lead to faster convergence at the cost of additional bias.

Setting γ_i, γ_i' and n

With respect to Algorithm 8, it is suggested in [123] to set $\gamma_i = C_0 i^{-p}$ and $\gamma_i' = C_0' i^{-p}$ where $p \in [0.6, 0.9]$ (in the experiments performed in this thesis, we have set $p = 0.8$), C_0 and C_0' starting with $(\beta_0 d)^{-1}$ and $(\rho_0^2 d)^{-1}$ respectively, and then readjusting it if necessary. With respect to n, we have followed the recommendation in [123] and used a single sample (i.e., $n = 1$) on each iteration (we did not observe significant difference for larger values of n).

Setting ω_n

Following [123], we recommend setting ω_n as follows

$$\omega_n = \begin{cases} 0 & \text{if } n < N_0, \\ 1 & \text{if } N_0 \leq n \leq N_1, \\ \gamma_n & \text{otherwise,} \end{cases}$$

where N_0 is the number of initial iterations to be discarded (when $n < N_0$, the values of ρ_n^2 and β_n are still bouncing and not stabilized), $n \in [N_0, N_1]$ corresponds to the averaging estimation phase, in which the values of ρ_n^2 and β_n have stabilized and start converging, and $n > N_1$ is known as the refinement phase where we use decreasing weights to enhance the accuracy of the estimator (see [123, Section 3.3.1] for details).

Setting a stopping criterion

It is recommended to supervise the evolution of $|\overline{\beta}_{m+1} - \overline{\beta}_m|/\overline{\beta}_m$ and $|\overline{\rho}^2_{m+1} - \overline{\rho}^2_m|/\overline{\rho}^2_m$ in the execution of Algorithm 8 until they reach a tolerance level ψ to stop the algorithm execution. In our imaging experiments, we set $\psi = 10^{-4}$ but we have observed that $\psi = 10^{-3}$ is often enough to reach an acceptable estimate of the hyper-parameters in small computational times.

Other implementation considerations

In the implementation of the SAPG method, it is important to update the step-size of the MCMC method to sample X_i and Z_i within each iteration of the SAPG scheme, as the maximum step-size depends on the value of ρ_i^2.

Regarding the Lipschitz constant L one needs to compute the step-size for MYULA and SK-ROCK algorithms, the model (3.8) has $L = \lambda^{-1} + L_f$. With respect to the augmented model (5.2), the Lipschitz constant is $L_a = \lambda^{-1} + (\rho^2 + L_f^{-1})^{-1}$ which we use to implement SGS, ls-MYULA and ls-SK-ROCK. Therefore, we set the step-size of the MCMC methods to $1/L$ for MYULA, to $1/L_a$ for SGS and ls-MYULA,, and to $\delta_s^{\max}$ for SK-ROCK and ls-SK-ROCK, where $\delta_s^{\max}$ can be found in Algorithms 6 and 10, respectively.

5.5 Numerical experiments

We now illustrate the improvement that can be obtained by sampling the augmented model (5.1) using Algorithms 9 and 10 together with an optimal estimate of ρ^2 using Algorithm 8. To evaluate the performance of the methods in a variety of situations, we perform two imaging experiments related to *image deblurring* (whose model is strongly log-concave) and *image inpainting* (whose model is weakly log-concave), previously explained in Sections 2.1.1 and 2.1.2, respectively, using the *cameraman* test image. We implement these algorithms as described in the implementation guidelines (see Section 5.4.3).

For a fair comparison the results we show have been plotted as a function of the number of gradient evaluations, i.e., the number of times $\nabla \log p^\lambda(x|, y, \beta)$ and $\nabla_z \log p^\lambda(z|y, \beta, \rho^2)$ are computed in our algorithms[3]. The plots we show include the evolution of the MCMC samples in burn-in stage using the scalar statistic $\log p(X_n|y, \beta)$ for MYULA and SK-ROCK, and $\log p(X_n^{\text{grad}}|y, \beta)$ for SGS, ls-MYULA and ls-SK-ROCK. We have also plotted the progression of the mean-squared error (MSE) between the posterior mean and the true image, when all the algorithms have reached their steady state (i.e., after burn-in period), including the MAP estimate defined in (2.3.3) and computed using a highly efficient optimization algorithm called SALSA [4, 3] for the *image deblurring* and *image inpainting* experiments.

We also provide pixel-wise standard deviation plots as a way of quantifying the uncertainty in the delivered solution. We have also computed standard deviation plots performing down-sampling by averaging the samples by a factor of $2 \times j$ where $j = \{1, 2, 4\}$, which allows us

[3]Each iteration of the MYULA, SGS and ls-MYULA requires one gradient evaluation, whereas SK-ROCK and ls-SK-ROCK requires s evaluations on each iteration.

to observe the uncertainty in image structures at different scales. Finally, we also show auto-correlation plots of the slowest component of the samples produced by each of the methods[4], applying a 1-in-s thinning to the MYULA, SGS and ls-MYULA chains to equal the number of gradient evaluations between the mentioned methods (one gradient evaluation per iteration) and SK-ROCK/ls-SK-ROCK methods (s gradient evaluations per iteration). With this quantity, we have also compute effective sample sizes (ESS)[5] of the five algorithms discussed in this chapter, where the sum is truncated at lag k when the lag-k autocorrelation reaches a value less than 0.05.

For completeness, in Table 5.7 we have also provided computing times of all the experiments. These results have been obtained on an Intel core i5-8350U@1.70GHz workstation running MATLAB R2018a.

5.5.1 Image deblurring

To examine the performance of the MCMC methods in different scenarios, we consider the deblurring problem explained in Section 2.1.1, but using two additional test images: *boat* and *man*. In this case, the target posterior distributions are, as follows

$$p(x|y,\beta) \propto \exp\left[-\|y-Hx\|^2/2\sigma^2 - \beta \mathrm{TV}(x)\right] \tag{5.15}$$

$$p(x,z|y,\beta,\rho^2) \propto \exp\left[-\|y-Hx\|^2/2\sigma^2 - \beta \mathrm{TV}(z) - \|x-z\|^2/2\rho^2\right], \tag{5.16}$$

Figures 5.2(a),(b) show the additional test images consider in this experiment, and Figures 5.2(c)-(d) show the corresponding observations y for each image. Recall that Figure 2.1(a) shows the *cameraman* test image and Figure 2.1(b) its corresponding blurry and noisy observation y.

(a) *boat*, true image x (b) *man*, true image x

(c) *boat*, observation y (d) *man*, observation y

Figure 5.2: Image deblurring experiments: Test images x and their corresponding noisy and blurred observations y.

We begin estimating optimal values for β and ρ^2 for the given models implementing Algorithm 8 setting $\gamma_i = \gamma_i' = 10 \times i^{-0.8}/d$, $\beta_0 = 0.04$, $\rho_0^2 = L_f^{-1} = \sigma^2$ and $X_0 = Z_0 = H^\intercal y$.

[4]The chain's slowest component was identified by computing the approximated eigenvalue decomposition of the poster covariance matrix and projecting the samples onto the leading eigenvector.

[5]Recall that $\mathrm{ESS} = n\{1 + 2\sum_k \rho(k)\}^{-1}$, where n is the total number of samples and $\sum_k \rho(k)$ is the sum of the K monotone sample auto-correlations which we estimated with the initial monotone sequence estimator.

The corresponding results for the parameters estimation are given in Table 5.1, together with the Lipschitz constants L and L_a required to sample (5.15) and (5.16) respectively. We then generate 5×10^6 samples using MYULA and $5 \times 10^6/s$ samples using SK-ROCK (with $s = 15$) from (5.15), and 5×10^6 samples using SGS and ls-MYULA and $5 \times 10^6/s$ samples using ls-SK-ROCK (with $s = 15$) from (5.16). The results of these three experiments are plotted in Figure 5.3. In particular, we note from the evolution of the MSE (when the chains have reached the typical set of the target distributions) that ls-MYULA and ls-SK-ROCK outperform SGS, as we are using an exact MYULA implementation rather than a noisy one, as shown in Section 5.4. We have also reported the step-size used by each method in Table 5.2.

Table 5.1: Values for β and ρ^2 estimated using Algorithm 8 for (5.15) and (5.16) in the image deblurring experiments, together with the corresponding Lipschitz constants L and L_a.

Experiment	β	ρ^2	σ^2	$L = 1/\lambda + 1/\sigma^2$	$L_a = 1/\lambda + (\sigma^2 + \rho^2)^{-1}$
cameraman	0.044	0.480	0.335	5.959	4.205
boat	0.048	0.244	0.170	11.749	8.290
man	0.048	0.265	0.185	10.804	7.624

Table 5.2: Image deblurring experiment: Summary of the values for the step-size δ for each of the MCMC methods applied to the three imaging experiments: cameraman, boat and man.

MCMC method	**Cameraman**	**Boat**	**Man**
MYULA	0.167	0.085	0.092
SK-ROCK ($s = 15$)	67.959	34.468	37.483
SGS	0.237	0.120	0.131
ls-MYULA	0.237	0.120	0.131
ls-SK-ROCK ($s = 15$)	96.294	48.858	53.120

We also plot the autocorrelation function of the slowest component from the chains of the MCMC algorithms, this is shown in Figures 5.3(c),(f),(i) and, as can be seen, ls-SK-ROCK presents the fastest decay. In addition, Table 5.3 shows the effective sample sizes (ESS) associated with these autocorrelation plots, and one can notice that ls-SK-ROCK reaches the largest ESS. In addition, we also illustrated in Figure 5.4 the minimum mean-square estimator (MMSE) of all the MCMC methods for all three deblurring experiments.

Table 5.3: Image deblurring experiments: Effective sample sizes of the slowest component, after generating 15×10^3 samples using the five algorithms discussed in this work, and the speed increase (i.e., sp. inc.) achieved by the algorithms w.r.t. MYULA.

	MYULA	**SK-ROCK**		**SGS**		**ls-MYULA**		**ls-SK-ROCK**	
	ESS	ESS	Sp. inc.	ESS	Sp. inc.	ESS	Sp. inc.	ESS	Sp. inc.
Cam.	46	1175	25.54	49	1.07	54	1.17	1604	34.87
Man	23	804	34.96	38	1.65	47	2.04	1107	48.13
Boat	34	669	19.68	42	1.24	45	1.32	944	27.76

Finally, Figure 5.5 shows the marginal posterior standard deviation of the cameraman deblurring experiment at different scales. One can notice that edges show higher uncertainty, which is expected due to the nature of the forward operator. As can be seen, ls-MYULA and, in particular, ls-SK-ROCK outperform SGS in terms of delivering accurate (i.e., less noisy) estimates, showing the benefit of using these algorithms to sample in a more efficient way the augmented posterior distribution.

5.5.2 Image inpainting

To conclude, we perform the image inpainting experiment described in Section 2.1.2. To test the MCMC methods in different regimes, we will use the *cameraman* test image shown in Figure 2.1(a), and the *boat* and *man* test images illustrated in Figure 5.2(a)-(b) (the observation images y are illustrated in Figures 2.2 and 5.6). For this experiment, we consider the following models

$$p(x|y,\beta) \propto \exp\left[-\|y-Ax\|^2/2\sigma^2 - \beta \mathrm{TV}(x)\right] \tag{5.17}$$

$$p(x,z|y,\beta,\rho^2) \propto \exp\left[-\|y-Ax\|^2/2\sigma^2 - \beta \mathrm{TV}(z) - \|x-z\|^2/2\rho^2\right], \tag{5.18}$$

where $f_y(x) = \|y-Ax\|^2/2\sigma^2$, $A \in \mathbb{R}^{m\times d}$ is a rectangular matrix obtained by taking a random subset of rows from the identity matrix in dimension d, and $g(x) = \mathrm{TV}(x)$.

We first proceed to estimate optimal hyperparameters β and ρ^2 for (5.17) and (5.18) using Algorithm 8 setting $\gamma_i = \gamma_i' = 10 \times i^{-0.8}/d$, $\beta_0 = 0.5$, $\rho_0^2 = L_f^{-1}/2 = \sigma^2/2$ and $X_0 = Z_0 = A^\intercal y$. The estimated parameter values can be seen in Table 5.4, together with the Lipschitz constants L and L_a required to sample (5.17) and (5.18) respectively.

Table 5.4: Values for β and ρ^2 estimated using Algorithm 8 for (5.17) and (5.18) in the image inpainting experiments

Experiment	β	ρ^2	σ^2	$L = 1/\lambda + 1/\sigma^2$	$L_a = 1/\lambda + (\sigma^2+\rho^2)^{-1}$
cameraman	0.058	0.65	0.388	5.146	3.530
boat	0.057	0.35	0.208	9.588	6.571
man	0.05	0.37	0.219	9.105	6.243

Having obtained our estimates from the SAGP algorithm for the values of the hyperparameters, we proceed to generate 5×10^6 MYULA samples and $5\times10^6/s$ SK-ROCK samples (with $s = 15$) from (5.17) and 5×10^6 SGS and ls-MYULA samples, and $5\times10^6/s$ ls-SK-ROCK samples (with $s = 15$) from (5.18). The step-sizes for each method are reported in Table 5.5.

Table 5.5: Image inpainting experiment: Summary of the values for the step-size δ for each of the MCMC methods applied to the three imaging experiments: cameraman, boat and man.

MCMC method	**Cameraman**	**Boat**	**Man**
MYULA	0.194	0.104	0.109
SK-ROCK ($s = 15$)	78.698	42.238	44.470
SGS	0.283	0.152	0.160
ls-MYULA	0.283	0.152	0.160
ls-SK-ROCK ($s = 15$)	114.717	61.627	64.874

With the generated samples, we proceed to plot the results of these experiments in Figure 5.7. We first notice the acceleration that can be obtained from ls-SK-ROCK in the burn-in stage, illustrated by the evolution of the scalar estimate $\log p(X_n|y,\beta)$ of the MCMC samples. Then, we show the evolution of the mean-squared error (MSE) between the mean of the samples and the true image x after burn-in and, as can be seen, ls-SK-ROCK is computationally efficient in being the fastest method to reach the MMSE in all three experiments, even outperforming the MAP estimate in terms of accuracy in all three experiments; moreover, the improvement of ls-MYULA over SGS, in terms of accuracy is evidently similar to our previous results.

We also plot the autocorrelation function of the pixel values for the slowest component in Figures 5.7(c),(f),(i) and, as can be seen, the ACF of the ls-SK-ROCK samples decays faster than all the other MCMC methods. In addition, Table 5.6 shows the effective sample sizes (ESS) associated with these autocorrelation plots, and one can notice that ls-SK-ROCK reaches the largest ESS. We also illustrate in Figure 5.8 the MMSE of all the MCMC methods for all three inpainting experiments and, as in previous numerical results, we can see in Figures 5.7(b),(e),(h)

that ls-SK-ROCK is the fastest method in compute this estimate.

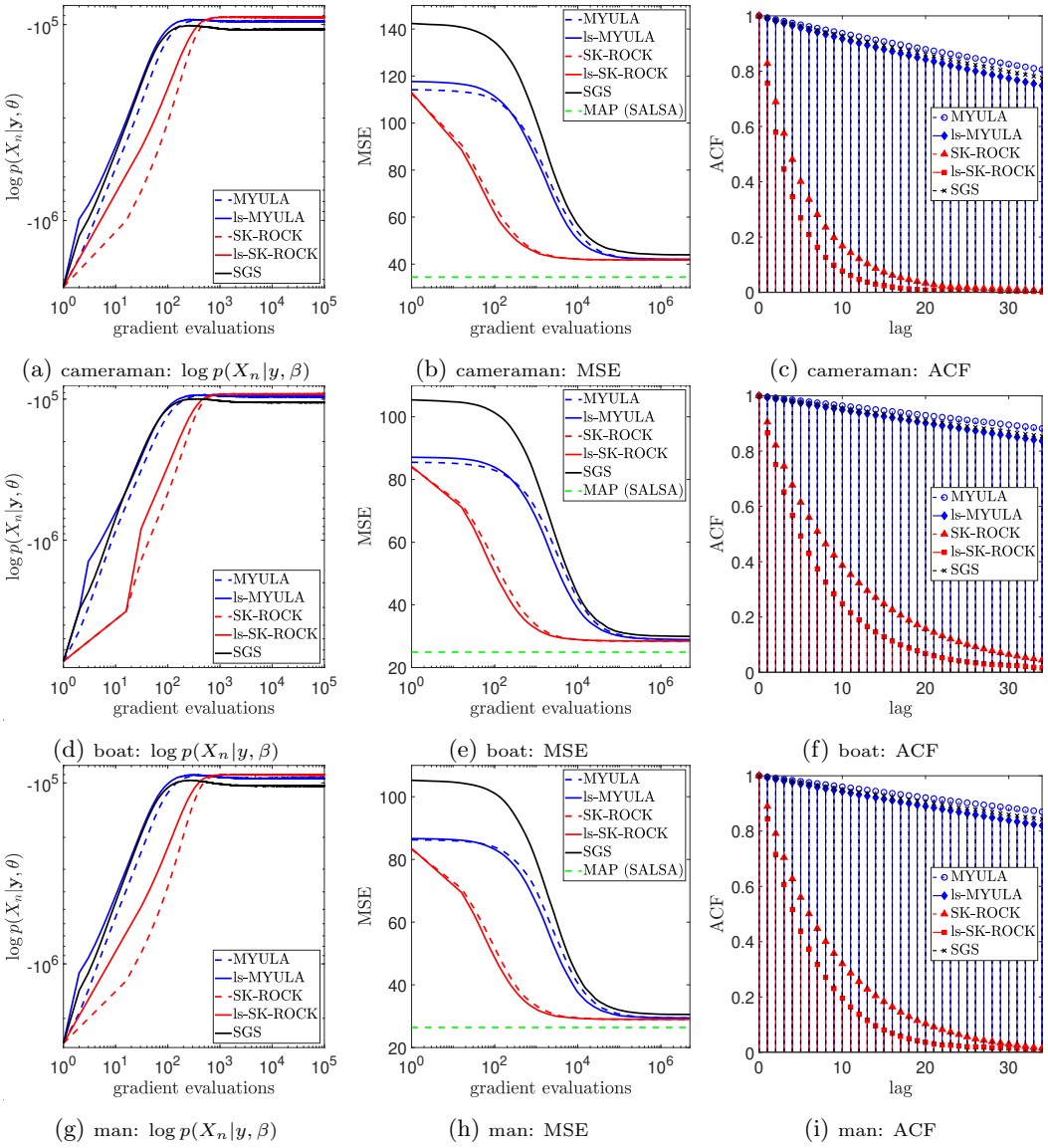
Table 5.6: Image inpainting experiments: Effective sample sizes of the slowest component, after generating 15×10^3 samples using the five algorithms discussed in this work, and the speed increase (i.e., sp. inc.) achieved by the algorithms w.r.t. MYULA.

	MYULA	**SK-ROCK**		**SGS**		**ls-MYULA**		**ls-SK-ROCK**	
	ESS	ESS	Sp. inc.	ESS	Sp. inc.	ESS	Sp. inc.	ESS	Sp. inc.
Cam.	14	281	20.07	14	1	21	1.5	448	32
Man	8	163	20.37	9	1.13	8	1	255	31.88
Boat	5	113	22.6	8	1.6	5	1	133	26.6

Finally, Figure 5.9 presents uncertainty quantification plots by showing pixel-wise standard deviation estimates for the cameraman inpainting experiment. In this case, the uncertainty is concentrated on the unobserved pixels, which is expected given the nature of the inpainting problem. One can notice that ls-MYULA and ls-SK-ROCK are slightly less noisy than SGS, showing the good performance of these algorithms in sampling the augmented posterior distribution.

Table 5.7: Summary of the execution times (in seconds) to produce one sample (i.e., after one iteration) on each of the MCMC algorithms implemented for each experiment.

Imaging Experiment	**MYULA**	**SK-ROCK (s=15)**	**ls-MYULA**	**ls-SK-ROCK (s=15)**	**SGS**
deblurring	3.8×10^{-2}	6.1×10^{-1}	4.3×10^{-2}	6.1×10^{-1}	4.7×10^{-2}
inpainting	4.1×10^{-2}	5.8×10^{-1}	3.8×10^{-2}	5.6×10^{-1}	4.7×10^{-2}

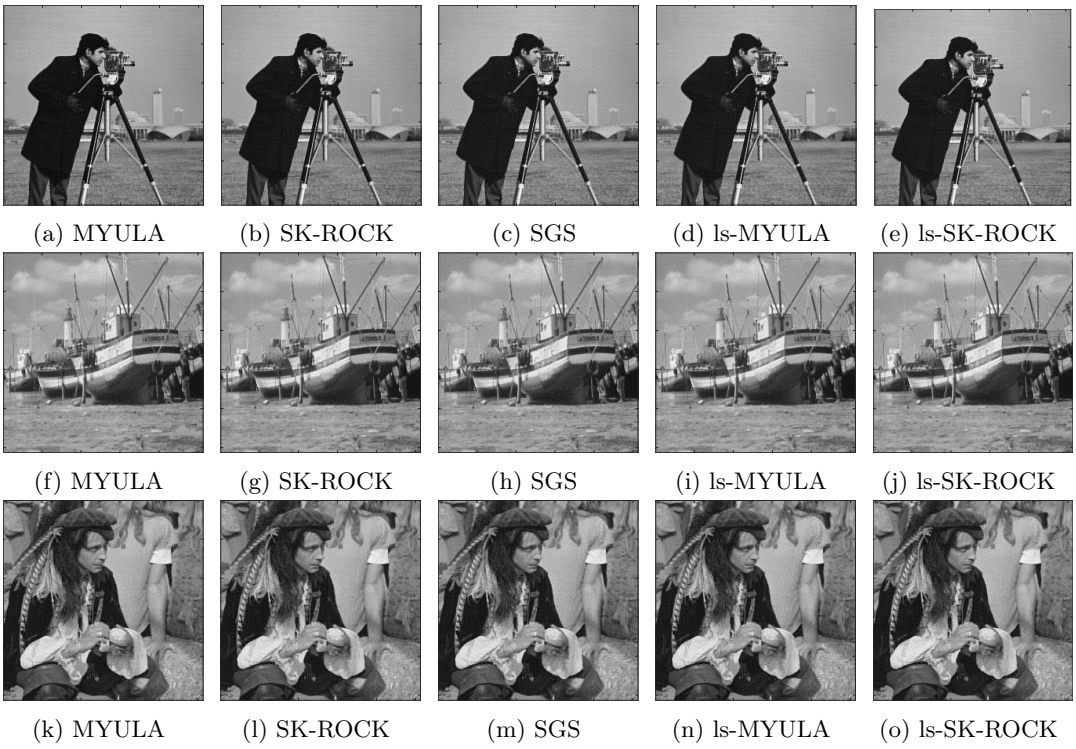


(a) cameraman: $\log p(X_n|y,\beta)$ (b) cameraman: MSE (c) cameraman: ACF

(d) boat: $\log p(X_n|y,\beta)$ (e) boat: MSE (f) boat: ACF

(g) man: $\log p(X_n|y,\beta)$ (h) man: MSE (i) man: ACF

Figure 5.3: Image deblurring experiments: (a),(d),(g) Convergence to the typical set of the posterior distribution (5.15) and (5.16) for the first 10^5 MYULA, SGS and ls-MYULA samples, and the first $10^5/s$ SK-ROCK and ls-SK-ROCK samples ($s = 15$). (b),(e),(h) MSE between the mean of the algorithms and the true image, measured using 5×10^6 MYULA, SGS and ls-MYULA samples, and $5\times10^6/s$ SK-ROCK and ls-SK-ROCK samples ($s = 15$), after burn-in. (c),(f),(i) Autocorrelation function for the values of the slowest component of the samples.

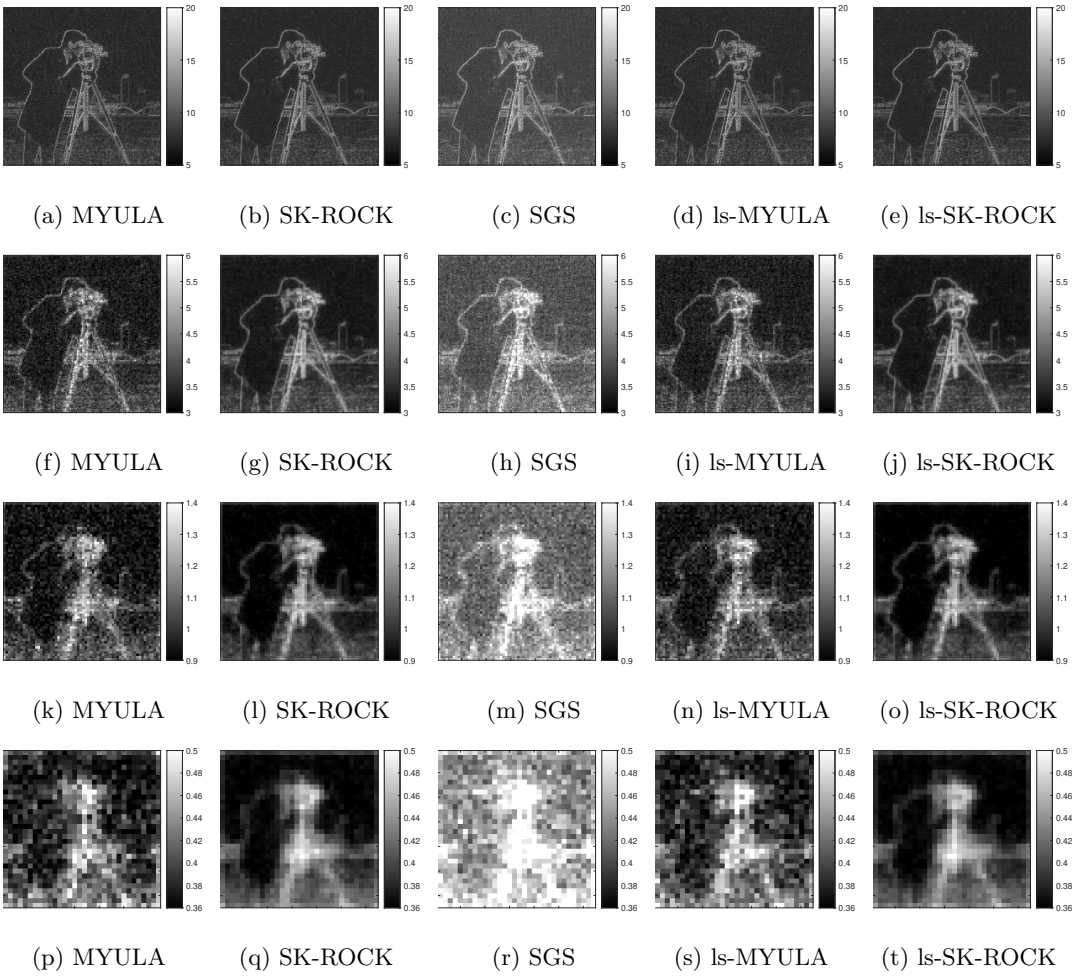


(a) MYULA (b) SK-ROCK (c) SGS (d) ls-MYULA (e) ls-SK-ROCK

(f) MYULA (g) SK-ROCK (h) SGS (i) ls-MYULA (j) ls-SK-ROCK

(k) MYULA (l) SK-ROCK (m) SGS (n) ls-MYULA (o) ls-SK-ROCK

Figure 5.4: Image deblurring experiments: MMSE computed using 5×10^6 MYULA, SGS and ls-MYULA samples, and $5 \times 10^6/s$ SK-ROCK and ls-SK-ROCK samples, after burn-in.



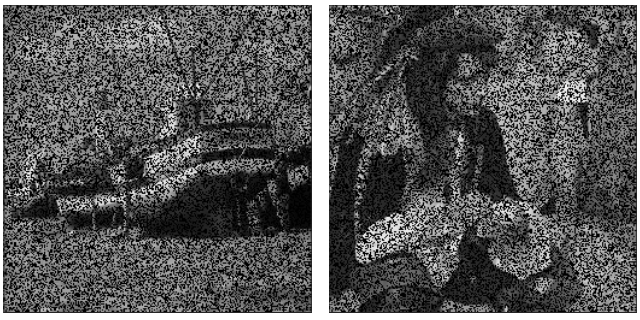
(a) MYULA (b) SK-ROCK (c) SGS (d) ls-MYULA (e) ls-SK-ROCK

(f) MYULA (g) SK-ROCK (h) SGS (i) ls-MYULA (j) ls-SK-ROCK

(k) MYULA (l) SK-ROCK (m) SGS (n) ls-MYULA (o) ls-SK-ROCK

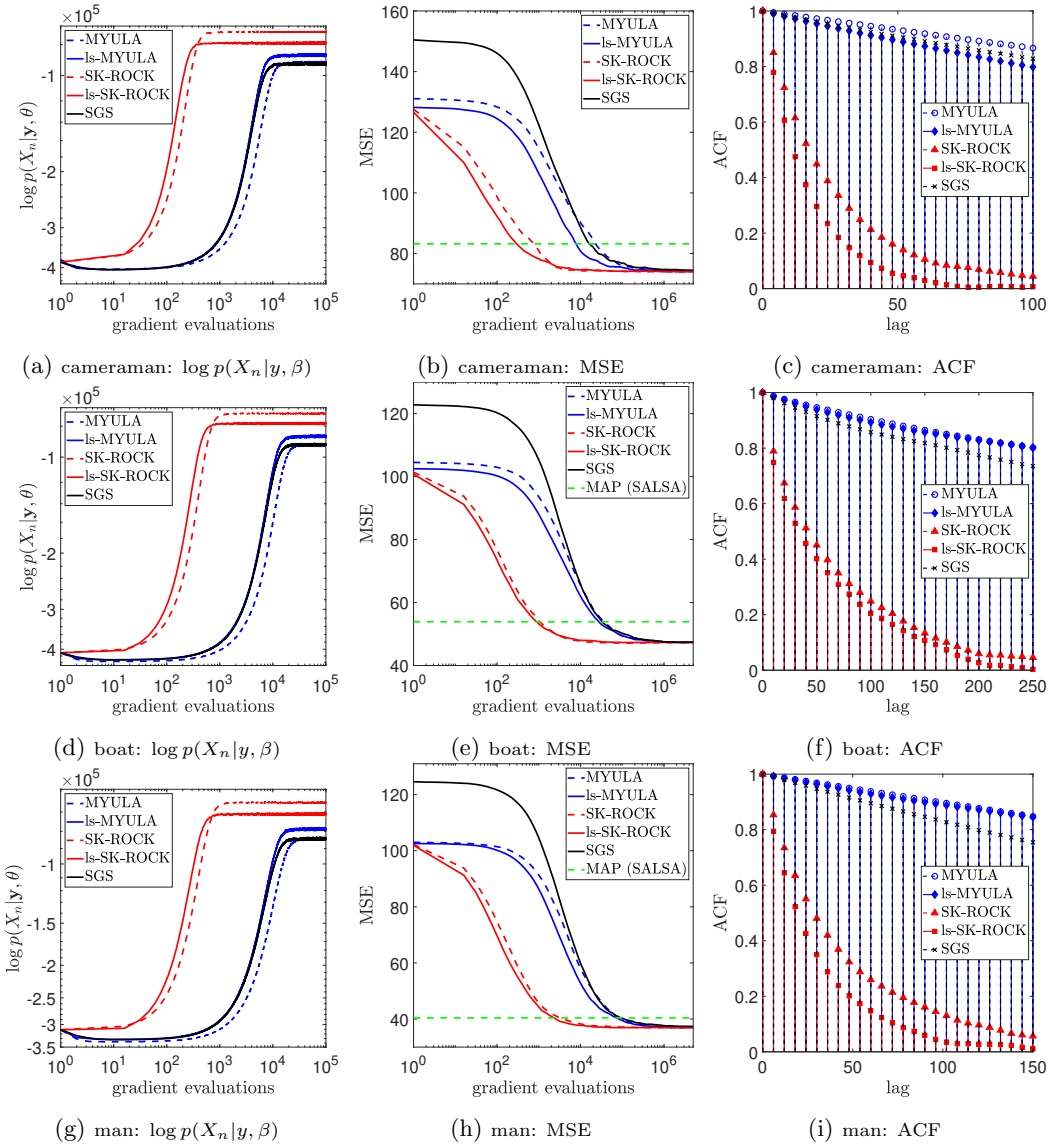
(p) MYULA (q) SK-ROCK (r) SGS (s) ls-MYULA (t) ls-SK-ROCK

Figure 5.5: Image deblurring experiments - cameraman: pixel-wise standard deviation computed using 10^4 MYULA, SGS and ls-MYULA samples with a 1-in-15 thinning, and 10^4 SK-ROCK and ls-SK-ROCK samples with $s = 15$, using (a)-(e) the original sample size (256×256) and with downsampling by a factor of (f)-(j) 2, (k)-(o) 4 and (p)-(t) 8.



(a) *boat*: observation y (b) *man*: observation y

Figure 5.6: Image inpainting experiments: noisy and incomplete observations y (pixels in black represent unobserved components).



(a) cameraman: $\log p(X_n|y,\beta)$ (b) cameraman: MSE (c) cameraman: ACF

(d) boat: $\log p(X_n|y,\beta)$ (e) boat: MSE (f) boat: ACF

(g) man: $\log p(X_n|y,\beta)$ (h) man: MSE (i) man: ACF

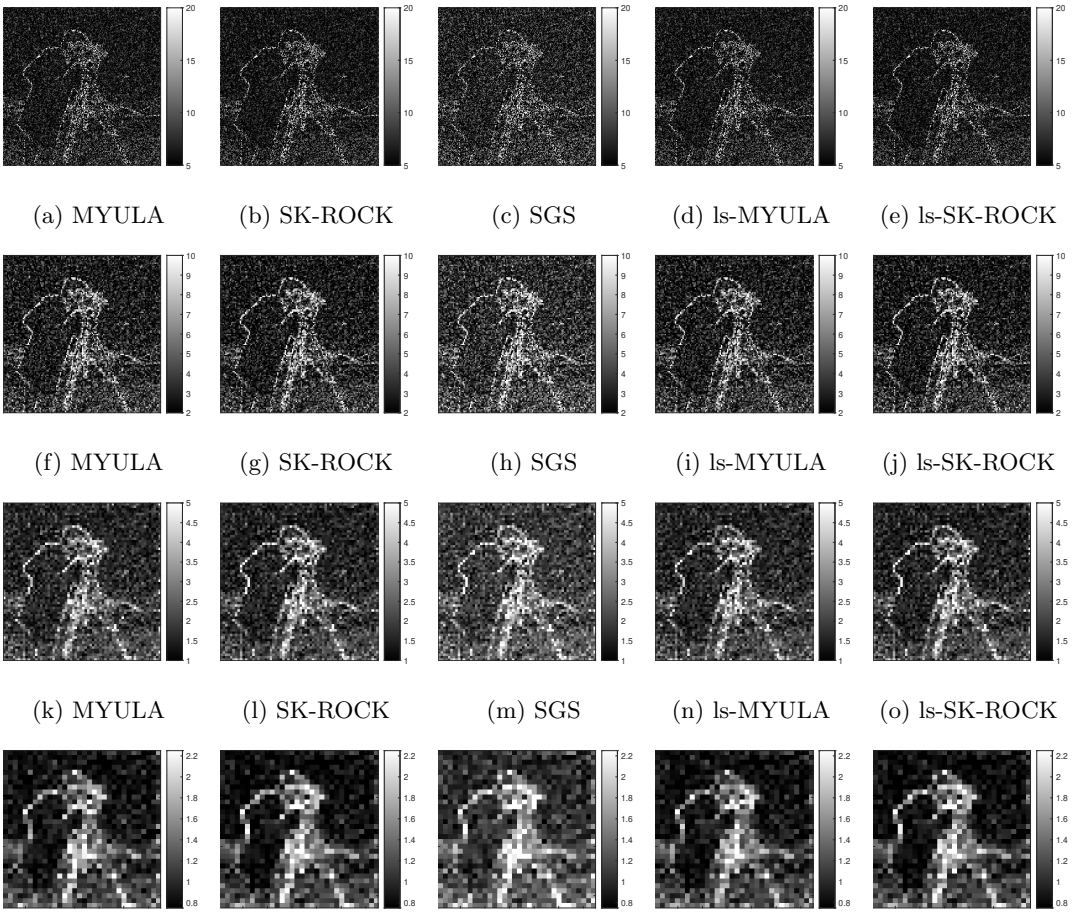
Figure 5.7: Image inpainting experiments: (a),(d),(g) Convergence to the typical set of the posterior distribution (5.17) and (5.18) for the first 10^5 MYULA, SGS and ls-MYULA samples, and the first $10^5/s$ SK-ROCK and ls-SK-ROCK samples ($s = 15$). (b),(e),(h) MSE between the mean of the algorithms and the true image, measured using 5×10^6 MYULA, SGS and ls-MYULA samples, and $5\times10^6/s$ SK-ROCK and ls-SK-ROCK samples ($s = 15$), after burn-in. (c),(f),(i) Autocorrelation function for the slowest component of the samples.

(a) MYULA (b) SK-ROCK (c) SGS (d) ls-MYULA (e) ls-SK-ROCK

(f) MYULA (g) SK-ROCK (h) SGS (i) ls-MYULA (j) ls-SK-ROCK

(k) MYULA (l) SK-ROCK (m) SGS (n) ls-MYULA (o) ls-SK-ROCK

Figure 5.8: MMSE for the image inpainting experiment.



(a) MYULA (b) SK-ROCK (c) SGS (d) ls-MYULA (e) ls-SK-ROCK

(f) MYULA (g) SK-ROCK (h) SGS (i) ls-MYULA (j) ls-SK-ROCK

(k) MYULA (l) SK-ROCK (m) SGS (n) ls-MYULA (o) ls-SK-ROCK

(p) MYULA (q) SK-ROCK (r) SGS (s) ls-MYULA (t) ls-SK-ROCK

Figure 5.9: Image inpainting experiments - cameraman: pixel-wise standard deviation computed using 10^4 MYULA, SGS and ls-MYULA samples with a 1-in-15 thinning, and 10^4 SK-ROCK and ls-SK-ROCK samples with $s = 15$, using (a)-(e) the original sample size (256×256) and with downsampling by a factor of (f)-(j) 2, (k)-(o) 4 and (p)-(t) 8.

Chapter 6

Conclusions

This thesis presented new proximal MCMC methods to perform Bayesian computation in inverse problems related to imaging sciences. The explicit EM approximation struggles in problems that are ill-posed or ill-conditioned because of the corresponding severe step-size restrictions. These same issues arise in the case of gradient descent and proximal gradient optimisation algorithms that also suffer from step-size restrictions.

In Chapter 4 we have addressed this issue proposing an explicit stochastic Runge-Kutta-Chebyshev discretisation of the Moreau-Yosida regularised overdamped Langevin diffusion process named SK-ROCK. The proposed method employs a significantly more advanced discrete approximation that has better convergence properties in problems that are ill-posed and ill-conditioned, compared to the conventional EM discrete approximation of the Langevin diffusion. The SK-ROCK approximation used in this thesis achieves a similar acceleration quality to accelerated proximal optimisation algorithms [9]. For Gaussian models, we prove rigorously the acceleration of the Markov chains in the 2-Wasserstein distance as a function of the condition number κ when moderate accuracy is required. The superior behaviour of this method is further demonstrated with a range of numerical experiments, including non-blind image deblurring, tomographic reconstruction, and hyperspectral unmixing, with total-variation and ℓ_1 priors. The generated Markov chains exhibit faster mixing, achieve larger effective sample sizes, and produce lower mean square estimation errors at an equal computational budget. This allows, for example, to accurately estimate high order statistical moments and perform uncertainty quantification analyses in a more computationally efficient way.

In Chapter 5, we presented a strategy to combine MYULA or proximal SK-ROCK with augmentation and relaxation in the manner of SGS. This was achieved by first establishing that SGS is equivalent to a noisy ULA scheme applied to the marginal distribution of the latent variable z in an augmented Bayesian model $x, z|y, \beta, \rho^2$. This then naturally led to two new samplers that apply MYULA and SK-ROCK to the latent marginal distribution $z|y, \beta, \rho^2$. Probabilities and expectations w.r.t. the primal marginal $x|y, \beta, \rho^2$ are then straightforwardly computed by using a Rao-Blackwellised Monte Carlo estimator. Moreover, we also observed empirically that there is a range of values for ρ^2 for which convergence speed and model quality improve (in the sense of the model evidence). Increasing ρ^2 beyond this range leads to improvements in convergence speed at the expense of significant estimation bias. We, therefore, proposed to adopt an empirical Bayesian approach and set ρ^2, together with the regularisation parameter β, by maximum marginal likelihood estimation from y. This was achieved by using a SAPG scheme that converges in very few iterations. We illustrated the benefits of adopting the proposed methodology with two experiments, image deblurring and image inpainting. The results showed that the new proximal SK-ROCK algorithm that benefits from augmentation and relaxation outperforms the other methods from the state of the art in terms of computational efficiency.

Furthermore, this thesis opens a series of interesting directions for future research. An important perspective is to theoretically analyse the non-asymptotic convergence properties of SK-ROCK for non-Gaussian log-concave models and derive bounds in total-variation and Wasserstein metrics; this is highly technical and will require the development of new analysis techniques. It would also be interesting to explore possible Metropolis-adjusted variants of

the stochastic Runge-Kutta-Chebyshev and the latent-space methods discussed in this thesis. Future studies may also address the exploration of other envelopes in the sampling methods studied in this thesis, such as the forward-backward envelope, recently proposed in [50], together with the empirical Bayesian computation algorithms reviewed in this Thesis [123, 36] in order to investigate whether the accuracy of the MMLE is improved and, therefore, the estimation of the hyperparameters and the relaxation parameter of the augmented distribution. Another interesting direction is to extend the proposed augmented approach to plug-and-play priors encoded by neural network denoisers [80], and on establishing non-asymptotic convergence results for the latent-space methods.

Appendix A

Wasserstein distance - Gaussian process

We begin computing the distribution Q_n of the n samples generated by the approximation (4.3). We will work in the one-dimensional case but the results easily extend to higher dimensions, as can be seen later. First, we can notice that the solution of (4.3) can be expressed by the following recursive formula:

$$X_n = (R_1(z))^n X_0 + \sqrt{2\delta}\sum_{i=1}^{n}(R_1(z))^{n-i}(R_2(z))\xi_i,$$

where X_0 is the initial condition of the problem. Computing expectations on both sides of the latter equation, we have:

$$\mathbb{E}(X_n) = (R_1(z))^n X_0.$$

Then, we compute the variance as follows:

$$\begin{aligned}
\mathbb{E}(X_n^2) - \mathbb{E}(X_n)^2 &= 2\delta\sum_{i=1}^{n}(R_1(z))^{2(n-i)}(R_2(z))^2, \\
&= 2\delta(R_1(z))^{2n}(R_2(z))^2\sum_{i=1}^{n}\frac{1}{(R_1(z))^{2i}}, \\
&= 2\delta(R_1(z))^{2n}(R_2(z))^2\frac{1}{(R_1(z))^2}\left[\frac{1-\frac{1}{(R_1(z))^{2n}}}{1-\frac{1}{(R_1(z))^2}}\right], \\
&= 2\delta(R_2(z))^2\left[\frac{(R_1(z))^{2n}-1}{(R_1(z))^2-1}\right],
\end{aligned}$$

thus, the approximated distribution Q_n of the n-th sample produced by the numerical scheme (4.3) is defined, as follows:

$$Q_n = \mathcal{N}\left((R_1(z))^n X_0, 2\delta(R_2(z))^2\left[\frac{(R_1(z))^{2n}-1}{(R_1(z))^2-1}\right]\right).$$

We can now compute the Wasserstein distance between the two univariate Gaussian distributions P and Q_n:

$$W_2(P;Q_n)^2 = (R_1(z))^{2n}X_0^2 + \left[\sigma - \sqrt{2\delta}R_2(z)\left(\frac{1-(R_1(z))^{2n}}{1-(R_1(z))^2}\right)^{1/2}\right]^2.$$

As we mentioned at the beginning of this Appendix, we can trivially extend the last result for a d-dimensional Gaussian distribution i.e. let $P \sim N(0,\Sigma)$ where $\Sigma = \mathrm{diag}(\sigma_1^2,...,\sigma_d^2)$ and

$X_0 = (x_0^1, ..., x_0^d)^T$ and obtain the following expression for the Wasserstein distance:

$$W_2(P; Q_n)^2 = \sum_{i=1}^{d} (R_1(z_i))^{2n} (x_0^i)^2 + \sum_{i=1}^{d} \left[\sigma_i - \sqrt{2\delta} R_2(z_i) \left(\frac{1 - (R_1(z_i))^{2n}}{1 - (R_1(z_i))^2} \right)^{1/2} \right]^2,$$

where $z_i = -\delta/\sigma_i^2$. This concludes the proof.

Appendix B

Explicit bound for the Wasserstein distance

We begin applying the triangle inequality to $W_2(P;Q_{n+1})^2$ as follows:

$$W_2(P;Q_{n+1})^2 \leq W_2(P;\tilde{Q})^2 + W_2(\tilde{Q};Q_{n+1})^2, \tag{B.1}$$

where $\tilde{Q}$ is the unique invariant distribution to which (4.3) converges when $n \to \infty$ and it is defined as:

$$\tilde{Q} = \mathcal{N}\left(0, 2\delta(R_2(z))^2\left[\frac{1}{1-(R_1(z))^2}\right]\right),$$

thus, we have that:

$$\begin{aligned} W_2(\tilde{Q};Q_{n+1})^2 &= \sum_{i=1}^{d} R_1(z_i)^{2n+2}(x_0^i)^2 + \sum_{i=1}^{d}\left[\left(\frac{2\delta R_2(z_i)^2}{1-R_1(z_i)^2}\right)^{1/2}\right. \\ &\quad \left. -\sqrt{2\delta}R_2(z_i)\left(\frac{1-R_1(z_i)^{2n+2}}{1-R_1(z_i)^2}\right)^{1/2}\right]^2, \\ &= \sum_{i=1}^{d}\left[R_1(z_i)^{2n+2}(x_0^i)^2 + \frac{2\delta R_2(z_i)^2}{1-R_1(z_i)^2}\left(\sqrt{1}-\sqrt{1-R_1(z_i)^{2n+2}}\right)^2\right] \end{aligned} \tag{B.2}$$

It is easy to prove the following property:

$$\frac{1-\sqrt{1-x^{2n+2}}}{1-\sqrt{1-x^{2n}}}x^2 \leq x^2, \tag{B.3}$$

for $x \in (0,1)$. Thus, applying the latter in (B.2) we have:

$$\begin{aligned} W_2(\tilde{Q};Q_{n+1})^2 &\leq \sum_{i=1}^{d} R_1(z_i)^{2n+2}(x_0^i)^2 \\ &\quad + \sum_{i=1}^{d}\frac{2\delta R_2(z_i)^2}{1-R_1(z_i)^2}\left(R_1(z_i)^2\left[1-\sqrt{1-R_1(z_i)^{2n}}\right]\right)^2, \\ &\leq \sum_{i=1}^{d}\left[R_1(z_i)^{2n}(x_0^i)^2\right. \\ &\quad \left. + \frac{2\delta R_2(z_i)^2}{1-R_1(z_i)^2}\left(1-\sqrt{1-R_1(z_i)^{2n}}\right)^2\right]R_1(z_i)^2, \\ &\leq \max_{1\leq i\leq d} R_1(z_i)^2 W_2(\tilde{Q};Q_n)^2. \end{aligned}$$

Thus, (B.1) becomes:

$$W_2(P;Q_{n+1})^2 \leq W_2(P;\tilde{Q})^2 + \max_{1\leq i\leq d} R_1(z_i)^2 W_2(\tilde{Q};Q_n)^2.$$

Let:

$$C = \max_{1\leq i\leq d} R_1(z_i)^2,$$

applying (B.3) $n+1$ times, we finally have that:

$$W_2(P;Q_{n+1})^2 \leq W_2(P;\tilde{Q})^2 + C^{n+1} W_2(\tilde{Q};Q_0)^2,$$

concluding the proof.

As an attempt to minimise the bound found in the latter expression, we will try to accelerate the decay of the constant C composed by $R_1(z)$ in the stochastic ROCK methods. This approach follows closely the approach in [49]. In particular, in order to bound $R_1(z)$ by one, we need that $|\omega_0 + \omega_1 z| \leq 1$, in other words we need that:

$$-1 \leq \omega_0 - \omega_1 \frac{\delta}{\sigma_i^2} \leq 1.$$

Let $L := 1/\sigma_{\min}^2$ and $\ell := 1/\sigma_{\max}^2$, so we have that:

$$-1 \leq \omega_0 - \omega_1 L\delta \leq \omega_0 - \omega_1 \ell\delta \leq 1,$$

which it is the same as:

$$-1 \leq \omega_1 \ell\delta - \omega_0 \leq \omega_1 L\delta - \omega_0 \leq 1.$$

Working with the first two members on the left-hand side of the latter inequality, we have that:

$$\delta \geq \frac{\omega_0 - 1}{\ell\omega_1}. \tag{B.4}$$

We choose the smallest δ to have an efficient algorithm i.e., $\delta = (\omega_0 - 1)/\ell\omega_1$ and now working with the last two members on the right-hand side of the previous inequality, we have that:

$$\kappa := \frac{L}{\ell} \leq \frac{\omega_0 + 1}{\omega_0 - 1} = 1 + \frac{2s^2}{\eta},$$

where κ is the condition number of our Gaussian problem. We choose the smallest s to have an efficient algorithm and the latter expression determines the parameter s as:

$$s = \left[\sqrt{\frac{\eta}{2}(\kappa - 1)} \right], \tag{B.5}$$

where $[x]$ is the notation for the integer rounding of real numbers.

Bibliography

[1] A. Abdulle. "Explicit Stabilized Runge–Kutta Methods". In: *Encyclopedia of Applied and Computational Mathematics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 460–468.

[2] A. Abdulle, I. Almuslimani, and G. Vilmart. "Optimal Explicit Stabilized Integrator of Weak Order 1 for Stiff and Ergodic Stochastic Differential Equations". In: *SIAM/ASA Journal on Uncertainty Quantification* 6.2 (2018), pp. 937–964.

[3] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. "An Augmented Lagrangian Approach to the Constrained Optimization Formulation of Imaging Inverse Problems". In: *IEEE Transactions on Image Processing* 20.3 (2011), pp. 681–695.

[4] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. "Fast Image Recovery Using Variable Splitting and Constrained Optimization". In: *IEEE Transactions on Image Processing* 19.9 (2010), pp. 2345–2356.

[5] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan. "An Introduction to MCMC for Machine Learning". In: *Machine Learning* 50 (2003), pp. 5–43.

[6] S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb. "Solving inverse problems using data-driven models". In: *Acta Numerica* 28 (2019), pp. 1–174.

[7] R. G. Baraniuk. "Compressive Sensing [Lecture Notes]". In: *IEEE Signal Processing Magazine* 24.4 (2007), pp. 118–121.

[8] A. Barbu and S.-C. Zhu. *Monte Carlo Methods*. Springer, 2020.

[9] A. Beck. *First-order methods in optimization*. SIAM, 2017.

[10] D. Belomestny, L. Iosipoi, E. Moulines, A. Naumov, and S. Samsonov. "Variance reduction for Markov chains with application to MCMC". In: *Statistics and Computing* 30.4 (2020), pp. 973–997.

[11] M. Benning and M. Burger. "Modern regularization methods for inverse problems". In: *Acta Numerica* 27 (2018), pp. 1–111.

[12] M. Betancourt. *A Conceptual Introduction to Hamiltonian Monte Carlo*. 2018. arXiv: 1701.02434.

[13] M. Betancourt, S. Byrne, S. Livingstone, and M. Girolami. "The geometric foundations of Hamiltonian Monte Carlo". In: *Bernoulli* 23.4A (2017), pp. 2257–2298.

[14] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot. "Hyperspectral Unmixing Overview: Geometrical, Statistical, and Sparse Regression-Based Approaches". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 5.2 (2012), pp. 354–379.

[15] C. Blaiotta, M. Jorge Cardoso, and J. Ashburner. "Variational inference for medical image segmentation". In: *Computer Vision and Image Understanding* 151 (2016), pp. 14–28.

[16] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. "Variational Inference: A Review for Statisticians". In: *Journal of the American Statistical Association* 112.518 (2017), pp. 859–877.

[17] H. Boche, R. Calderbank, G. Kutyniok, and J. Vybíral. *Compressed Sensing and its Applications: MATHEON Workshop 2013*. Springer International Publishing, 2015.

[18] N. Bou-Rabee, A. Eberle, and R. Zimmer. “Coupling and convergence for Hamiltonian Monte Carlo”. In: *The Annals of Applied Probability* 30.3 (2020), pp. 1209–1250.

[19] N. Bou-Rabee and J. M. Sanz-Serna. “Randomized Hamiltonian Monte Carlo”. In: *The Annals of Applied Probability* 27.4 (2017), pp. 2159–2194.

[20] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Foundations and Trends® in Machine Learning* 3.1 (2011), pp. 1–122.

[21] K. Bredies, K. Kunisch, and T. Pock. “Total Generalized Variation”. In: *SIAM Journal on Imaging Sciences* 3.3 (2010), pp. 492–526.

[22] S. Brooks, A. Gelman, G. Jones, and X.-L. Meng. *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, 2011.

[23] C. L. Byrne. *Applied Iterative Methods*. CRC Press, 2007.

[24] N. Cai, Y. Zhou, S. Wang, B. W.-K. Ling, and S. Weng. “Image denoising via patch-based adaptive Gaussian mixture prior method”. In: *Signal, Image and Video Processing* 10.6 (2016), pp. 993–999.

[25] X. Cai, M. Pereyra, and J. D. McEwen. “Uncertainty quantification for radio interferometric imaging – I. Proximal MCMC methods”. In: *Monthly Notices of the Royal Astronomical Society* 480.3 (2018), pp. 4154–4169.

[26] X. Cai, M. Pereyra, and J. D. McEwen. “Uncertainty quantification for radio interferometric imaging: II. MAP estimation”. In: *Monthly Notices of the Royal Astronomical Society* 480.3 (2018), pp. 4170–4182. ISSN: 0035-8711.

[27] E. J. Candes, J. Romberg, and T. Tao. “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information”. In: *IEEE Transactions on Information Theory* 52.2 (2006), pp. 489–509.

[28] E. J. Candès, J. K. Romberg, and T. Tao. “Stable signal recovery from incomplete and inaccurate measurements”. In: *Communications on Pure and Applied Mathematics* 59.8 (2006), pp. 1207–1223.

[29] A. Chambolle. “An Algorithm for Total Variation Minimization and Applications”. In: *Journal of Mathematical Imaging and Vision* 20.3 (2004), pp. 89–97.

[30] A. Chambolle and T. Pock. “An introduction to continuous optimization for imaging”. In: *Acta Numerica* 25 (2016), pp. 161–319.

[31] S. Chaudhuri, R. Velmurugan, and R. Rameshan. “Blind Deconvolution Methods: A Review”. In: *Blind Image Deconvolution: Methods and Convergence*. Cham: Springer International Publishing, 2014, pp. 37–60.

[32] L. Condat. “Discrete Total Variation: New Definition and Minimization”. In: *SIAM Journal on Imaging Sciences* 10.3 (2017), pp. 1258–1290.

[33] A. Dalalyan. “Further and stronger analogy between sampling and optimization: Langevin Monte Carlo and gradient descent”. In: *Proceedings of the 2017 Conference on Learning Theory*. Vol. 65. Proceedings of Machine Learning Research. PMLR, 2017, pp. 678–689.

[34] A. S. Dalalyan and L. Riou-Durand. “On sampling from a log-concave density using kinetic Langevin diffusions”. In: *Bernoulli* 26.3 (2020), pp. 1956–1988.

[35] V. De Bortoli, A. Durmus, M. Pereyra, and A. F. Vidal. “Efficient stochastic optimisation by unadjusted Langevin Monte Carlo”. In: *Statistics and Computing* 31.3 (2021).

[36] V. De Bortoli, A. Durmus, M. Pereyra, and A. F. Vidal. “Maximum Likelihood Estimation of Regularization Parameters in High-Dimensional Inverse Problems: An Empirical Bayesian Approach. Part II: Theoretical Analysis”. In: *SIAM Journal on Imaging Sciences* 13.4 (2020), pp. 1990–2028.

[37] A. P. Dempster, N. M. Laird, and D. B. Rubin. “Maximum Likelihood from Incomplete Data Via the EM Algorithm”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1 (1977), pp. 1–22.

[38] D. Donoho, M. Elad, and V. Temlyakov. “Stable recovery of sparse overcomplete representations in the presence of noise”. In: *IEEE Transactions on Information Theory* 52.1 (2006), pp. 6–18.

[39] R. Douc, É. Moulines, and D. Stoffer. *Nonlinear Time Series: Theory, Methods and Applications with R Examples*. Texts in statistical science. Boca Raton, FL: Chapman & Hall/CRC, 2014.

[40] S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth. “Hybrid Monte Carlo”. In: *Physics Letters B* 195.2 (1987), pp. 216–222.

[41] A. B. Duncan, T. Lelievre, and G. A. Pavliotis. “Variance reduction using nonreversible Langevin samplers”. In: *Journal of Statistical Physics* 163.3 (2016), pp. 457–491.

[42] A. Durmus, S. Majewski, and B. Miasojedow. “Analysis of Langevin Monte Carlo via Convex Optimization”. In: *Journal of Machine Learning Research* 20.73 (2019), pp. 1–46.

[43] A. Durmus and É. Moulines. “High-dimensional Bayesian inference via the unadjusted Langevin algorithm”. In: *Bernoulli* 25.4A (2019), pp. 2854–2882.

[44] A. Durmus and É. Moulines. “Nonasymptotic convergence analysis for the unadjusted Langevin algorithm”. In: *The Annals of Applied Probability* 27.3 (2017), pp. 1551–1587.

[45] A. Durmus, É. Moulines, and M. Pereyra. “Efficient Bayesian Computation by Proximal Markov Chain Monte Carlo: When Langevin Meets Moreau”. In: *SIAM Journal on Imaging Sciences* 11.1 (2018), pp. 473–506.

[46] R. Dwivedi, Y. Chen, M. J. Wainwright, and B. Yu. “Log-concave sampling: Metropolis-Hastings algorithms are fast!” In: *Proceedings of the 31st Conference On Learning Theory*. Vol. 75. Proceedings of Machine Learning Research. PMLR, 2018, pp. 793–797.

[47] D. A. van Dyk and X.-L. Meng. “The Art of Data Augmentation”. In: *Journal of Computational and Graphical Statistics* 10.1 (2001), pp. 1–50.

[48] O. Eches, N. Dobigeon, and J.-Y. Tourneret. “Enhancing Hyperspectral Image Unmixing With Spatial Correlations”. In: *IEEE Transactions on Geoscience and Remote Sensing* 49.11 (2011), pp. 4239–4247.

[49] A. Eftekhari, B. Vandereycken, G. Vilmart, and K. C. Zygalakis. “Explicit stabilised gradient descent for faster strongly convex optimisation”. In: *BIT Numerical Mathematics* 61.1 (2021), pp. 119–139.

[50] A. Eftekhari, L. Vargas, and K. Zygalakis. *The Forward-Backward Envelope for Sampling with the Overdamped Langevin Algorithm*. 2022. arXiv: 2201.09096.

[51] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari. “Image inpainting: A review”. In: *Neural Processing Letters* 51.2 (2020), pp. 2007–2028.

[52] L. Fan, F. Zhang, H. Fan, and C. Zhang. “Brief review of image denoising techniques”. In: *Visual Computing for Industry, Biomedicine, and Art* 2.1 (2019).

[53] G. Fort, E. Ollier, and A. Samson. “Stochastic proximal-gradient algorithms for penalized mixed models”. In: *Statistics and Computing* 29 (2019), pp. 231–253.

[54] C. W. Fox and S. J. Roberts. “A tutorial on variational Bayesian inference”. In: *Artificial Intelligence Review* 38.2 (2012), pp. 85–95.

[55] A. Gelman, W. R. Gilks, and G. O. Roberts. “Weak convergence and optimal scaling of random walk Metropolis algorithms”. In: *The Annals of Applied Probability* 7.1 (1997), pp. 110–120.

[56] C. J. Geyer. “Practical Markov Chain Monte Carlo”. In: *Statistical Science* 7.4 (1992), pp. 473–483.

[57] C. Gilavert, S. Moussaoui, and J. Idier. “Efficient Gaussian Sampling for Solving Large-Scale Inverse Problems Using MCMC”. In: *IEEE Transactions on Signal Processing* 63.1 (2015), pp. 70–80.

[58] W. R. Gilks, S. Richardson, and D. Spiegelhalter. *Markov chain Monte Carlo in practice*. Chapman & Hall, 1998.

[59] M. Girolami and B. Calderhead. “Riemann manifold Langevin and Hamiltonian Monte Carlo methods”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73.2 (2011), pp. 123–214.

[60] J. V. Goldman, T. Sell, and S. S. Singh. “Gradient-Based Markov Chain Monte Carlo for Bayesian Inference With Non-differentiable Priors”. In: *Journal of the American Statistical Association* 0.0 (2021), pp. 1–12.

[61] M. González, A. Almansa, and P. Tan. “Solving Inverse Problems by Joint Posterior Maximization with Autoencoding Prior”. In: *SIAM Journal on Imaging Sciences* 15.2 (2022), pp. 822–859.

[62] J. Goodman and J. Weare. “Ensemble samplers with affine invariance”. In: *Communications in Applied Mathematics and Computational Science* 5.1 (2010), pp. 65–80.

[63] C. Guillemot and O. Le Meur. “Image Inpainting : Overview and Recent Advances”. In: *IEEE Signal Processing Magazine* 31.1 (2014), pp. 127–144.

[64] O. Güler. “New Proximal Point Algorithms for Convex Minimization”. In: *SIAM Journal on Optimization* 2.4 (1992), pp. 649–664.

[65] J. Hadamard. “Sur les problèmes aux dérivées partielles et leur signification physique”. In: *Princeton university bulletin* (1902), pp. 49–52.

[66] E. Hairer, C. Lubich, and G. Wanner. “Geometric numerical integration illustrated by the Störmer–Verlet method”. In: *Acta Numerica* 12 (2003), pp. 399–450.

[67] U. Hämarik, B. Kaltenbacher, U. Kangro, and E. Resmerita. “Regularization by discretization in Banach spaces”. In: *Inverse Problems* 32 (2016), p. 035004.

[68] W. K. Hastings. “Monte Carlo sampling methods using Markov chains and their applications”. In: *Biometrika* 57.1 (1970), pp. 97–109. ISSN: 0006-3444.

[69] D. J. Higham. “A-Stability and Stochastic Mean-Square Stability”. In: *BIT Numerical Mathematics* 40.2 (2000), pp. 404–409.

[70] D. J. Higham. “Mean-Square and Asymptotic Stability of the Stochastic Theta Method”. In: *SIAM Journal on Numerical Analysis* 38.3 (2000), pp. 753–769.

[71] M. D. Hoffman and A. Gelman. “The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo”. In: *Journal of Machine Learning Research* 15.47 (2014), pp. 1593–1623.

[72] M. Holden, M. Pereyra, and K. C. Zygalakis. “Bayesian Imaging with Data-Driven Priors Encoded by Neural Networks”. In: *SIAM Journal on Imaging Sciences* 15.2 (2022), pp. 892–924.

[73] J. Hsieh, B. Nett, Z. Yu, K. Sauer, J.-B. Thibault, and C. A. Bouman. “Recent advances in CT image reconstruction”. In: *Current Radiology Reports* 1.1 (2013), pp. 39–51.

[74] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza. “Sparse Unmixing of Hyperspectral Data”. In: *IEEE Transactions on Geoscience and Remote Sensing* 49.6 (2011), pp. 2014–2039.

[75] M.-D. Iordache, J. M. Bioucas-Dias, and A. Plaza. “Total Variation Spatial Regularization for Sparse Hyperspectral Unmixing”. In: *IEEE Transactions on Geoscience and Remote Sensing* 50.11 (2012), pp. 4484–4502.

[76] H. Ishwaran. “Applications of Hybrid Monte Carlo to Bayesian Generalized Linear Models: Quasicomplete Separation and Neural Networks”. In: *Journal of Computational and Graphical Statistics* 8.4 (1999), pp. 779–799.

[77] J. Kaipio and E. Somersalo. *Statistical and computational inverse problems*. New York, NY: Springer New York, 2005.

[78] B. Kaltenbacher, A. Kirchner, and B. Vexler. “Adaptive discretizations for the choice of a Tikhonov regularization parameter in nonlinear inverse problems”. In: *Inverse Problems* 27 (2011), p. 125008.

[79] S. Kullback and R. A. Leibler. “On Information and Sufficiency”. In: *The Annals of Mathematical Statistics* 22.1 (1951), pp. 79–86.

[80] R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra. "Bayesian Imaging Using Plug & Play Priors: When Langevin Meets Tweedie". In: *SIAM Journal on Imaging Sciences* 15.2 (2022), pp. 701–737.

[81] E. Levitin and B. Polyak. "Constrained minimization methods". In: *USSR Computational Mathematics and Mathematical Physics* 6.5 (1966), pp. 1–50.

[82] W.-K. Ma, J. M. Bioucas-Dias, T.-H. Chan, N. Gillis, P. Gader, A. J. Plaza, A. Ambikapathi, and C.-Y. Chi. "A Signal Processing Perspective on Hyperspectral Unmixing: Insights from Remote Sensing". In: *IEEE Signal Processing Magazine* 31.1 (2014), pp. 67–81.

[83] A. Mahalakshmi and B. Shanthini. "A survey on image deblurring". In: *2016 International Conference on Computer Communication and Informatics (ICCCI)*. 2016, pp. 1–5.

[84] F. Maire, R. Douc, and J. Olsson. "Comparison of asymptotic variances of inhomogeneous Markov chains with application to Markov chain Monte Carlo methods". In: *The Annals of Statistics* 42.4 (2014), pp. 1483–1510.

[85] Y. Marnissi, A. Benazza-Benyahia, E. Chouzenoux, and J.-C. Pesquet. "Majorize-Minimize adapted metropolis-hastings algorithm. Application to multichannel image recovery". In: *2014 22nd European Signal Processing Conference (EUSIPCO)*. 2014, pp. 1332–1336.

[86] Y. Marnissi, E. Chouzenoux, J.-C. Pesquei, and A. Benazza-Benyahia. "An auxiliary variable method for Langevin based MCMC algorithms". In: *2016 IEEE Statistical Signal Processing Workshop (SSP)*. 2016, pp. 1–5.

[87] G. Mazzieri, R. Spies, and K. Temperini. "Existence, uniqueness and stability of minimizers of generalized Tikhonov–Phillips functionals". In: *Journal of Mathematical Analysis and Applications* 396.1 (2012), pp. 396–411.

[88] M. McDonnell. "Box-filtering techniques". In: *Computer Graphics and Image Processing* 17.1 (1981), pp. 65–70.

[89] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. "Equation of State Calculations by Fast Computing Machines". In: *The Journal of Chemical Physics* 21.6 (1953), pp. 1087–1092.

[90] P. Milanfar. *Super-resolution imaging*. CRC press, 2017.

[91] P. Monmarché. "High-dimensional MCMC with a standard splitting scheme for the underdamped Langevin diffusion." In: *Electronic Journal of Statistics* 15.2 (2021), pp. 4117–4166.

[92] Y. Nesterov. "A Method for Solving a Convex Programming Problem with Convergence Rate $\mathcal{O}(1/K^2)$". In: *Doklady Akademii Nauk SSSR*. Vol. 269. 3. 1983, pp. 543–547.

[93] T. Park and G. Casella. "The Bayesian Lasso". In: *Journal of the American Statistical Association* 103.482 (2008), pp. 681–686.

[94] M. Pereyra. "Maximum-a-Posteriori Estimation with Bayesian Confidence Regions". In: *SIAM Journal on Imaging Sciences* 10.1 (2017), pp. 285–302.

[95] M. Pereyra. "Proximal Markov chain Monte Carlo algorithms". In: *Statistics and Computing* 26.4 (2016), pp. 745–760.

[96] M. Pereyra. "Revisiting Maximum-A-Posteriori Estimation in Log-Concave Models". In: *SIAM Journal on Imaging Sciences* 12.1 (2019), pp. 650–670.

[97] M. Pereyra, L. V. Mieles, and K. C. Zygalakis. "Accelerating Proximal Markov Chain Monte Carlo by Using an Explicit Stabilized Method". In: *SIAM Journal on Imaging Sciences* 13.2 (2020), pp. 905–935.

[98] M. Pereyra, P. Schniter, É. Chouzenoux, J.-C. Pesquet, J.-Y. Tourneret, A. O. Hero, and S. McLaughlin. "A Survey of Stochastic Simulation and Optimization Methods in Signal Processing". In: *IEEE Journal of Selected Topics in Signal Processing* 10.2 (2016), pp. 224–241.

[99] M. Pereyra, L. A. Vargas-Mieles, and K. C. Zygalakis. *The split Gibbs sampler revisited: improvements to its algorithmic structure and augmented target distribution*. 2022. arXiv: 2206.13894.

[100] L. J. Rendell, A. M. Johansen, A. Lee, and N. Whiteley. "Global Consensus Monte Carlo". In: *Journal of Computational and Graphical Statistics* (2020), pp. 1–11.

[101] A. Repetti, M. Pereyra, and Y. Wiaux. "Scalable Bayesian Uncertainty Quantification in Imaging Inverse Problems via Convex Optimization". In: *SIAM Journal on Imaging Sciences* 12.1 (2019), pp. 87–118.

[102] C. P. Robert. *The Bayesian Choice*. Vol. 2. Springer, 2007.

[103] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Vol. 2. Springer, 2004.

[104] G. O. Roberts and R. L. Tweedie. "Exponential convergence of Langevin distributions and their discrete approximations". In: *Bernoulli* 2.4 (1996), pp. 341–363.

[105] L. I. Rudin, S. Osher, and E. Fatemi. "Nonlinear total variation based noise removal algorithms". In: *Physica D: Nonlinear phenomena* 60.1-4 (1992), pp. 259–268.

[106] L. I. Rudin, S. Osher, and E. Fatemi. "Nonlinear total variation based noise removal algorithms". In: *Physica D: Nonlinear Phenomena* 60.1 (1992), pp. 259–268.

[107] J. M. Sanz-Serna. "Markov Chain Monte Carlo and Numerical Differential Equations". In: *Current Challenges in Stability Issues for Numerical Differential Equations*. Springer International Publishing, 2014, pp. 39–88.

[108] J. M. Sanz-Serna and K. C. Zygalakis. "Wasserstein distance estimates for the distributions of numerical approximations to ergodic stochastic differential equations". In: *Journal of Machine Learning Research* 22.242 (2021), pp. 1–37.

[109] O. Scherzer, M. Grasmair, H. Grossauer, M. Haltmeier, and F. Lenzen. *Variational Methods in Imaging*. New York, NY: Springer New York, 2009.

[110] T. Schuster. *The Method of Approximate Inverse: Theory and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007.

[111] M. Simões, J. Bioucas-Dias, L. B. Almeida, and J. Chanussot. "A Convex Formulation for Hyperspectral Image Superresolution via Subspace-Based Regularization". In: *IEEE Transactions on Geoscience and Remote Sensing* 53.6 (2015), pp. 3373–3388.

[112] A. M. Stuart. "Inverse problems: A Bayesian perspective". In: *Acta Numerica* 19 (2010), pp. 451–559.

[113] S. Sun. "A review of deterministic approximate inference techniques for Bayesian machine learning". In: *Neural Computing and Applications* 23.7 (2013), pp. 2039–2050.

[114] M. A. Tanner and W. H. Wong. "The Calculation of Posterior Distributions by Data Augmentation". In: *Journal of the American Statistical Association* 82.398 (1987), pp. 528–540.

[115] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, and C.-W. Lin. "Deep learning on image denoising: An overview". In: *Neural Networks* 131 (2020), pp. 251–275.

[116] J. Tian and K.-K. Ma. "A survey on super-resolution imaging". In: *Signal, Image and Video Processing* 5 (2011), pp. 329–342.

[117] R. Tibshirani. "Regression Shrinkage and Selection Via the Lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1 (1996), pp. 267–288.

[118] A. N. Tikhonov, A. V. Goncharsky, V. V. Stepanov, and A. G. Yagola. *Numerical Methods for the Solution of Ill-Posed Problems*. Dordrecht: Springer, 1995.

[119] A. N. Tikhonov. "On the solution of ill-posed problems and the method of regularization". In: *Doklady Akademii Nauk SSSR*. Vol. 151. 3. 1963, pp. 501–504.

[120] J. A. Tropp and A. C. Gilbert. "Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit". In: *IEEE Transactions on Information Theory* 53.12 (2007), pp. 4655–4666.

[121] D. Ulyanov, A. Vedaldi, and V. Lempitsky. "Deep Image Prior". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[122] A. Vehtari, A. Gelman, D. Simpson, B. Carpenter, and P.-C. Bürkner. "Rank-Normalization, Folding, and Localization: An Improved $\widehat{R}$ for Assessing Convergence of MCMC (with Discussion)". In: *Bayesian Analysis* 16.2 (2021), pp. 667–718.

[123] A. F. Vidal, V. D. Bortoli, M. Pereyra, and A. Durmus. "Maximum Likelihood Estimation of Regularization Parameters in High-Dimensional Inverse Problems: An Empirical Bayesian Approach Part I: Methodology and Experiments". In: *SIAM Journal on Imaging Sciences* 13.4 (2020), pp. 1945–1989.

[124] S. J. Vollmer, K. C. Zygalakis, and Y. W. Teh. "Exploration of the (Non-)Asymptotic Bias and Variance of Stochastic Gradient Langevin Dynamics". In: *Journal of Machine Learning Research* 17.159 (2016), pp. 1–48.

[125] M. Vono, N. Dobigeon, and P. Chainais. "Split-and-Augmented Gibbs Sampler—Application to Large-Scale Inference Problems". In: *IEEE Transactions on Signal Processing* 67.6 (2019), pp. 1648–1661.

[126] M. Vono, N. Dobigeon, and P. Chainais. "Asymptotically Exact Data Augmentation: Models, Properties, and Algorithms". In: *Journal of Computational and Graphical Statistics* 30.2 (2021), pp. 335–348.

[127] M. Vono, N. Dobigeon, and P. Chainais. "High-Dimensional Gaussian Sampling: A Review and a Unifying Approach Based on a Stochastic Proximal Point Algorithm". In: *SIAM Review* 64.1 (2022), pp. 3–56.

[128] M. Vono, N. Dobigeon, and P. Chainais. "Sparse Bayesian binary logistic regression using the split-and-augmented Gibbs sampler". In: *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*. 2018, pp. 1–6.

[129] M. Vono, D. Paulin, and A. Doucet. "Efficient MCMC Sampling with Dimension-Free Convergence Rate using ADMM-type Splitting". In: *Journal of Machine Learning Research* 23.25 (2022), pp. 1–69.

[130] G. Wang, J. C. Ye, and B. De Man. "Deep learning for tomographic image reconstruction". In: *Nature Machine Intelligence* 2.12 (2020), pp. 737–748.

[131] Y.-Q. Wang and J.-M. Morel. "SURE Guided Gaussian Mixture Image Denoising". In: *SIAM Journal on Imaging Sciences* 6.2 (2013), pp. 999–1034.

[132] M. T. Wells, G. Casella, and C. P. Robert. "Generalized accept-reject sampling schemes". In: *A Festschrift for Herman Rubin*. Vol. 45. Institute of Mathematical Statistics, 2004, pp. 342–347.

[133] H. White. "Maximum Likelihood Estimation of Misspecified Models". In: *Econometrica* 50.1 (1982), pp. 1–25.

[134] A. Wibisono. "Sampling as optimization in the space of measures: The Langevin dynamics as a composite optimization problem". In: *Proceedings of the 31st Conference On Learning Theory*. Vol. 75. Proceedings of Machine Learning Research. PMLR, 2018, pp. 2093–3027.

[135] C. Williams and C. Rasmussen. "Gaussian Processes for Regression". In: *Advances in Neural Information Processing Systems*. Vol. 8. MIT Press, 1995.

[136] D. Wipf and H. Zhang. "Revisiting Bayesian Blind Deconvolution". In: *Journal of Machine Learning Research* 15.111 (2014), pp. 3775–3814.

[137] T. Würfl, M. Hoffmann, V. Christlein, K. Breininger, Y. Huang, M. Unberath, and A. K. Maier. "Deep Learning Computed Tomography: Learning Projection-Domain Weights From Image Domain in Limited Angle Problems". In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1454–1463.

[138] G. Yu, G. Sapiro, and S. Mallat. "Solving Inverse Problems With Piecewise Linear Estimators: From Gaussian Mixture Models to Structured Sparsity". In: *IEEE Transactions on Image Processing* 21.5 (2012), pp. 2481–2499.

[139] Q. Zhou, T. Yu, X. Zhang, and J. Li. “Bayesian Inference and Uncertainty Quantification for Medical Image Reconstruction with Poisson Data”. In: *SIAM Journal on Imaging Sciences* 13.1 (2020), pp. 29–52.