EPFL

Thèse n° 8951

HIERARCHICAL MARKOV CHAIN MONTE CARLO METHODS FOR BAYESIAN INVERSE PROBLEMS

Présentée le 22 avril 2022

Faculté des sciences de base
Calcul scientifique et quantification de l'incertitude - Chaire CADMOS
Programme doctoral en mathématiques

pour l'obtention du grade de Docteur ès Sciences

par

Juan Pablo MADRIGAL CIANCI

Acceptée sur proposition du jury

Prof. C. Hongler, président du jury
Prof. F. Nobile, directeur de thèse
Prof. . B. Sprungk, rapporteur
Prof. R. Scheichl, rapporteur
Dr G. Obozinski, rapporteur

École polytechnique fédérale de Lausanne

2022

Para mi mamá, Apolito, y el resto de mi familia. Lo logramos.

Acknowledgements

First and foremost, I would like to thank my doctoral advisor, Prof. Fabio Nobile, and my (unofficial) mentor, Prof. Raúl Tempone, for giving me the opportunity of working with them, for their guidance, patience, and expertise throughout these years.

Next, I would also like to thank Prof. Rob Scheichl, Prof. Björn Sprungk, and Dr. Guillaume Obozinski for agreeing to be part of doctoral jury, and to Prof. Clément Hongler for presiding it.

I was fortunate enough to have worked alongside plenty of brilliant and kind people that I consider my friends. In no particular order, I would like to thank Sundar, Evita (in particular for always being keen on seeing unsolicited pictures of my dog –who incidentally happens to be *the goodest* boy), Davide, Thomas, Yoshihito, Tommasso, Quentin, Celia (y nuestro *goss session* de las mañanas junto a un cafecito), Panos, Sebastian, Hakon, Matthieu, Jonas and Luis. A special shoutout to Panos and Seb, who as postdocs at the EPFL, guided me a lot at the early stages of my Ph.D., and to Celia and Jonas, who provided extremely valuable feedback during the writing of this Thesis.

I would also like to thank all the amazing people that I've met in Switzerland; *The buddies, The quizzards*, and other poorly chosen names. All jokes aside; Gabri, Aurel, Otta, Nadia (my favourite friend of a friend), HCW, Seb, Kira, David, Fede, and the Rios-Pierog family: thanks a lot guys for so many great memories. To Leks (who has saved my ass ∞-many times), Lolo (and Lou), Marco and Vale: I'm trully blessed to have met you. Your friendship was easily one of the highest points of my Ph.D.

Para mis amigos de Medellín, en especial a Turbo, Alex y el Gringo: gracias por brindarme su amistad y apoyarme desde la distancia durante todos estos años.

Para mi familia: gracias por aguantarme y por siempre brindarme tanto cariño de manera incondicional. Para Ricardo y Meil: espero ser un buen modelo a seguir para Uds.

Para mi mamá: éste es un trinfo de los dos. Todo lo que soy, y todo lo que tengo, es gracias al amor y la crianza que siempre me has dado. Espero poder seguir llenándote de orgullo.

And lastly, paraphrasing Calvin Cordozar Broadus Jr.:[1]

> I want to thank me for believing in me. I want to thank me for doing all this hard work. I want to thank me for having no days off. I want to thank me for never quitting. I want to thank me for always being a giver and trying to give more than I

[1] Better known as Snoop Dogg.

receive. I want to thank me for trying to do more right than wrong. I want to thank me for just being me at all times. [Juani], you a bad motherf**ker."

Lausanne, February 14, 2022 JPMC.

Abstract

This thesis is devoted to the construction, analysis, and implementation of two types of hierarchical Markov Chain Monte Carlo (MCMC) methods for the solution of large-scale Bayesian Inverse Problems (BIP).

The first hierarchical method we present is based on the idea of parallel tempering and is well-suited for BIP whose underlying posterior measure is multi-modal or concentrates around a lower-dimensional, non-linear manifold. In particular, we present two generalizations of the Parallel Tempering algorithm in the context of discrete-time Markov chain Monte Carlo methods for Bayesian inverse problems. These generalizations use state-dependent swapping rates and are inspired by the so-called continuous-time Infinite Swapping algorithm presented in Plattner et al. [J Chem Phys 135(13):134111, 2011]. We present a thorough analysis of the convergence of our proposed methods and show that they are reversible and geometrically ergodic. Numerical experiments conducted over an array of BIP show that our proposed algorithms significantly improve sampling efficiency over competing methodologies.

Our second hierarchical method is based on multi-level MCMC (ML-MCMC) techniques. In this setting, instead of sampling directly from a sufficiently accurate (and computationally expensive) posterior measure, one introduces a sequence of accuracy levels for the solution of the underlying computational model, which induces a hierarchy of posterior measures with increasing accuracy and cost to sample from. The key point of this algorithm is to construct highly coupled Markov chains together with the standard Multi-level Monte Carlo argument to obtain a better cost-tolerance complexity than a single-level MCMC algorithm. We present two types of multi-level MCMC algorithms which can be thought of as an extension of the ideas presented in Dodwell, et al. [SIAM-ASA J. Uncertain. Quantif (2015): 1075-1108].

Our first ML-MCMC method extends said ideas to a setting where a wider class of Independent Metropolis-Hastings (IMH) proposals are considered. We provide a thorough theoretical analysis and provide sufficient conditions on the proposals and the family of posteriors so that there exists a unique invariant probability measure for the coupled chains generated by our method, and the convergence to it is uniformly ergodic. We also generalize the cost-tolerance theorem of Dodwell et al., to our setting, and propose a self-tuning continuation-type ML-MCMC algorithm.

Our second ML-MCMC method presents an algorithm that admits state-dependent proposals by using a maximal coupling approach. This is desirable, from a methodological perspective, whenever it is difficult to construct suitable IMH proposals, or when the empirical measure resulting from samples from the posterior at the previous level does not satisfy the assumptions required for convergence of the ML-MCMC method. We present a theoretical analysis of the method at

hand and show that this new method has an invariant probability measure and converges to it with geometric ergodicity. We also extend the cost-tolerance theorem of Dodwell et. al. to this algorithm, albeit with quite restrictive assumptions. We illustrate both of the proposed ML-MCMC methodologies on several numerical examples.

Keywords: Bayesian inversion · Parallel tempering · Infinite swapping · Markov chain Monte Carlo · Multi-level Monte Carlo · Multi-level Markov chain Monte Carlo · Uncertainty quantification

Résumé

Cette thèse est consacrée à la construction, l'analyse et la mise en œuvre de deux types de méthodes hiérarchiques Markov Chain Monte Carlo (MCMC) pour la résolution de problèmes inverses bayésiens (BIP) à grande échelle.

La première méthode hiérarchique que nous présentons est basée sur l'idée de parallel tempering, et est bien adaptée pour BIP dont la distribution à-posteriori est multimodale ou se concentre autour d'une variété non linéaire de dimension inférieure. Nous présentons deux généralisations de l'algorithme Parallel Tempering dans le contexte des méthodes de Monte Carlo à chaîne de Markov à temps discret pour les BIPs. Ces généralisations utilisent des taux d'échange dépendant de l'état et s'inspirent de l'algorithme Infinite swapping en temps continu présenté par Plattner et al. (J Chem Phys 135(13):134111, 2011). Nous présentons une analyse approfondie de la convergence de nos méthodes proposées et montrons qu'elles sont réversibles et géométriquement ergodiques. Nous implémentons notre méthode proposée sur plusieurs BIPs. Les résultats numériques montrent que nos méthodes proposées améliorent considérablement l'efficacité de l'échantillonnage par rapport aux méthodologies concurrentes.

Notre deuxième méthode hiérarchique est basée sur des techniques MCMC multi-niveaux (ML-MCMC). Dans ce cadre, au lieu d'échantillonner directement à partir d'une mesure postérieure suffisamment précise (et coûteuse en calculs), on introduit une séquence de niveaux de précision pour la solution du modèle de calcul sous-jacent, ce qui induit une hiérarchie de mesures postérieures avec une précision et un coût d'échantillonnage croissants. Le point clé de cet algorithme est de construire des chaînes de Markov hautement couplées combinées par la technique standard de Monte Carlo multi-niveaux standard pour obtenir une meilleure complexité computationnelle de tolérance aux coûts qu'un algorithme MCMC à un seul niveau. Nous présentons deux types d'algorithmes MCMC multi-niveaux qui peuvent être considérés comme une extension des idées présentées dans Dodwell, et al. (SIAM-ASA J. Incertain. Quantif (2015) : 1075-108).

Notre première méthode ML-MCMC étend ces idées à un cadre où une classe plus large de distributions des propositions de Métropolis-Hastings indépendantes (IMH) est considérée. Nous fournissons une analyse théorique approfondie et fournissons des conditions suffisantes sur les destr. des propositions et la famille de distr. à-posteriori pour qu'il existe une mesure de probabilité invariante unique pour les chaînes couplées générées par notre méthode, et que de telles chaînes couplées convergent uniformément ergodiques vers elle. Nous généralisons également le théorème de complexité de Dodwell et al., à notre cadre, et proposons un algorithme ML-MCMC de type continuation à réglage automatique.

Notre deuxième méthode ML-MCMC présente un algorithme qui admet des distr. des propositions dépendantes de l'état de la chaîne en utilisant un algorithme de couplage maximal. Ceci est souhaitable, d'un point de vue méthodologique, chaque fois qu'il est difficile de construire des propositions IMH, ou lorsque la mesure empirique résultant des échantillons de la postérieure au niveau précédent ne satisfait pas les hypothèses requises pour la convergence de la méthode ML-MCMC. Nous présentons une analyse théorique de la méthode en question et montrons que cette nouvelle méthode a une mesure de probabilité invariante et la convergence vers elle est géométriquement ergodique. Nous étendons également le théorème de complexité Dodwell et. Al. à cet algorithme, mais avec des hypothèses plus restrictives. Nous illustrons les deux méthodologies ML-MCMC proposées sur plusieurs exemples numériques.

Mots clés : Inversion bayésienne · Parallel tempering · Infinite swapping · Monte Carlo par chaînes de Markov · Monte Carlo multi-niveaux · Monte Carlo par chaînes de Markov multi-niveau · Quantification de l'incertitude

Contents

Acknowledgements i

Abstract (English/Français) iii

1 Introduction 1
- 1.1 Uncertainty quantification and inverse problems . . . 3
- 1.2 Model problems in geophysics . . . 7
  - 1.2.1 Subsurface flow . . . 7
  - 1.2.2 Seismic inversion . . . 7
- 1.3 Literature review and contributions of this thesis . . . 9
  - 1.3.1 Tempering . . . 10
  - 1.3.2 Multi-level methods . . . 12
- 1.4 Outline . . . 16

2 Bayesian inverse problems 19
- 2.1 Preliminaries . . . 19
  - 2.1.1 Probability theory . . . 19
  - 2.1.2 Gaussian measures . . . 22
  - 2.1.3 Spaces of probability measures . . . 24
- 2.2 Bayesian inverse problems . . . 26
  - 2.2.1 Prior modeling . . . 28
  - 2.2.2 Well-posedness . . . 33
  - 2.2.3 Approximation and convergence . . . 35
- 2.3 Solving BIPs . . . 38
  - 2.3.1 Markov chain Monte Carlo . . . 38
  - 2.3.2 Approximate methods . . . 39
  - 2.3.3 Sequential methods . . . 45

3 Markov chain Monte Carlo 47
- 3.1 Markov Chain Monte Carlo . . . 47
- 3.2 Convergence . . . 51
- 3.3 MSE bounds . . . 59
  - 3.3.1 Non-asymptotic bounds on the MSE: known results . . . 59
  - 3.3.2 Non-asymptotic bounds on the MSE: new result for non-reversible chains 61

3.4 Review of common techniques and algorithms . . . 65
3.4.1 Construction of Q . . . 68

4 Generalized parallel tempering on Bayesian inverse problems 77
4.1 Introduction . . . 77
4.2 Problem setting . . . 79
4.2.1 Notation . . . 79
4.2.2 Tempering . . . 80
4.3 Generalizing Parallel Tempering . . . 81
4.3.1 The swapping kernel $\mathbf{q}$ . . . 83
4.3.2 The Parallel Tempering case . . . 86
4.3.3 Unweighted Generalized Parallel Tempering . . . 88
4.3.4 Weighted Generalized Parallel Tempering . . . 91
4.4 Ergodicity of Generalized Parallel Tempering . . . 94
4.4.1 Geometric ergodicity and L_2-spectral gap for GPT . . . 94
4.5 Numerical experiments . . . 104
4.5.1 Implementation remarks . . . 104
4.5.2 Experimental setup . . . 106
4.5.3 Density concentrated over a quarter circle-shaped manifold . . . 107
4.5.4 Multiple source elliptic BIP . . . 109
4.5.5 1D wave source inversion . . . 114
4.5.6 Acoustic wave source inversion . . . 116
4.5.7 High-dimensional acoustic wave inversion . . . 119
4.5.8 Application to a (semi-)realistic seismic source inversion problem: Tanzania case study . . . 122

5 A class of Multi-Level MCMC algorithms based on Independent Metropolis-Hastings 129
5.1 Introduction . . . 130
5.2 Multi-level Markov Chain Monte Carlo . . . 132
5.3 Convergence analysis of the ML-MCMC algorithm . . . 136
5.3.1 Convergence of the level-wise coupled chain . . . 137
5.3.2 Non-asymptotic bounds on the level-wise ergodic estimator . . . 140
5.4 Cost analysis of the ML-MCMC algorithm . . . 143
5.4.1 Proof of Theorem 5.4.1 . . . 146
5.5 Implementation . . . 156
5.5.1 A continuation-type ML-MCMC . . . 158
5.6 Numerical experiments . . . 160
5.6.1 Nested Gaussians . . . 160
5.6.2 Shifting Gaussians . . . 163

5.6.3 Subsurface flow . . . 168
5.6.4 High dimensional subsurface flow with Laplace's approximation . . . 170

6 Multi-level Markov chain Monte Carlo method based on maximally coupled proposals 181
6.1 Introduction . . . 181
6.2 ML-MCMC based on Maximal Coupling . . . 182
6.2.1 Reflection maximal coupling for Gaussian proposals . . . 183
6.2.2 Generating coupled chains . . . 185
6.2.3 Re-synchronizing the chains . . . 187
6.3 Convergence of the level-wise coupled pCN chain . . . 188
6.4 Numerical experiments . . . 195
6.4.1 Moving Gaussians, revisited . . . 195
6.4.2 Subsurface flow: moderate-dimensions . . . 200
6.5 Appendix . . . 205
6.5.1 A.1. Higher-dimensional subsurface flow revisited: some numerical results 205
6.5.2 A.2. Some results towards the complexity study of the maximal coupling algorithm . . . 209

7 Finalizing remarks 215
7.1 Summary and conclusions . . . 215
7.2 Perspectives . . . 217
7.2.1 Normalizing flows and ML-MCMC . . . 217
7.2.2 On the use and analysis of more efficient couplings . . . 219
7.2.3 Towards a multi-level generalized parallel tempering . . . 219

Bibliography 225

List of Figures

1.1 Depiction of forward and inverse problems. 4
1.2 Depiction of an un-normalized posterior density of interest, $\mu^y = \mu_2^y$, together with 2 un-normalized tempered versions μ_0^y, μ_1^y. 11
1.3 Depiction of accuracy and cost of $\mathcal{F}_\ell$ vs ℓ, where $\text{cost}_i < \text{cost}_{i+1}$ for three different levels. 14

2.1 Random field generated using the KL expansion of a random field with different truncation levels $K = 10, 50, 100$. 30
2.2 Top: Discretization meshes generated with a Laplace-like operator $\mathcal{A}^{-2}$. Bottom: Discretized random fields corresponding to each mesh 32

4.1 Cost per sample vs K for $S_K = \mathscr{S}_K$ for the forward model in Section 4.5.5 and the forward model in 4.5.7. 105
4.2 Tempered densities (with $T_1 = 1,\ T_2 = 17.1,\ T_3 = 292.4,\ T_4 = 5000$) for the density concentrated around a quarter circle-shaped manifold example. As we can see, the density becomes less concentrated as the temperature increases, which allows us to use RWM proposals with larger step sizes. 108
4.3 Scatter-plots of the samples from μ^y obtained with each algorithm on a single run. Top, from left to right: random walk Metropolis, PT and PSDPT. Bottom, from left to right: UGPT, WGPT (after re-weighting the samples), and WGPT, before re-weighting the samples. 110
4.4 True tempered densities for the elliptic BIP example. Notice that the density is not symmetric, due to the additional random noise. 112
4.5 Scatterplots of the samples from μ^y obtained with different algorithms on a single run. Top, from left to right: random walk Metropolis, PT and PSDPT. Bottom, from left to right: UGPT, WGPT (after re-weighting the samples), and WGPT, before re-weighting the samples. As we can see, WGPT (before re-weighting) is able to "connect" the parameter space. 113
4.6 Multi-modal potential for the Cauchy problem. Notice the minima around $u = -3$ and $u = 3$. 115
4.7 Plot of the log-likelihood for different values of s_1, s_2 and fixed values of $\alpha = 10$ and $\beta = 5000$. The magenta points represent the reciever locations R_1, R_2, R_3. The black point represents the true location of the source $(s_1, s_2) = (1.5, 1.0)$. 118

4.8 True field $\beta(x)$. Notice the anisotropy on the field. The magenta points represent the receiver locations. The black line represents the zero-level set of the field. . . 121

4.9 Posterior samples obtained with the UW GPT algorithm. Notice the resemblance to Figure 4.8. 122

4.10 (Top). Aerial view of the source-receiver geometry. Receivers are denoted in red and source location is in blue. Figure reproduced from [4], with permission from the publisher (Springer Nature). (Bottom). Depiction of the computational domain of the Tanzania test-case. Blue represents the PML. 124

4.11 Density of source location 127

5.1 Schematic of the possible configurations S_1, S_2, S_3, S_4. The sampler moves to the diagonal $\Delta = \{(u_{\ell,\ell-1}, u_{\ell,\ell}) \in \mathsf{X}^2 \text{ s.t } u_{\ell,\ell-1} = u_{\ell,\ell}\}$ whenever both chains accept (regardless of their current state) or when both chains reject, assuming that they were at the diagonal. 136

5.2 Family of posteriors μ_ℓ^y and fixed proposal distribution Q for the nested Gaussians example. 160

5.3 True posterior μ_ℓ^y for different levels $\ell = 0, 3, 6$ and histogram of the samples of $u_\ell \sim \mu_\ell^y$ obtained with the ML-MCMC algorithm described herein with $Q_\ell = \mathcal{N}(1, 3)$ (Top row)and the sub-sampling ML-MCMC algorithm (Bottom row). Both methods are able to obtain samples from the right posterior distribution. 161

5.4 (Left) $|\mathbb{E}_{\nu_\ell}[Y_\ell]|$ Vs. level. (Right) $\mathbb{V}_{\nu_\ell}[Y_\ell]$ Vs. level. In both figures, the rates were estimated over 100 independent runs, with 50,000 samples per level, on each run. Solid lines indicate the average value, dashed lines indicate 95% confidence intervals. 162

5.5 (Left) Number of samples, Vs. level for both algorithms. (Right) Synchronization rate vs level for both algorithms. 162

5.6 Total squared error er^2 vs tol for the nested Gaussians example. Here, we used 100 independent runs of the full C-ML-MCMC algorithm for three different tolerances; $\mathsf{tol} = 0.025, 0.05, 0.1$ (black circles). The red cross denotes the estimated MSE over the 100 runs. 163

5.7 Illustration of the posterior densities π_ℓ and the proposal Q for the moving Gaussians example. 164

5.8 Sample histograms for one ML-MCMC run at levels $\ell = 0, 3, 6$ (Top row): Fixed Gaussian proposal . (Bottom row): Sub-sampling approach. As it can be seen, the sub-sampling approach is not able to properly sample from the posterior at higher levels. 164

5.9 (Top left) Estimated expected value of QoI_ℓ for both ML-MCMC algorithms and the true mean of QoI_ℓ for different values of ℓ. (Top right) Expected value of $Y_\ell = \mathsf{QoI}_\ell - \mathsf{QoI}_{\ell-1}$ obtained with both algorithms for different values of ℓ. (Bottom left): Variance of Y_ℓ obtained with both algorithms for different values of ℓ. (Bottom right): Number of samples per level for each method with $\mathsf{tol} = 0.07$. On all plots, dashed lines represent a 95% confidence interval estimated over 100 independent runs of each algorithm. 166
5.10 Synchronization rate for both algorithms. Dashed lines represent a 95% confidence interval. As expected, the chains become more and more synchronized as the number of levels increases. 167
5.11 Total squared error er^2 vs tolerance tol for the moving Gaussian example. . . . 168
5.12 Decays of $\mathbb{E}_{\nu_\ell}[Y_\ell]$ and $\mathbb{V}_{\nu_\ell}[Y_\ell]$ vs level ℓ.As we can see, both quantities decay with the same rate, as predicted by the theory. 170
5.13 Computed squared error er^2 (using Equation 5.31) vs tol for the elliptic PDE example. 170
5.14 Plots of synchronization and acceptance rates using the Laplace-approximation proposal 176
5.15 Realization of u_{true} and the MAP $m_{\text{map},\ell}$ at each level. 176
5.16 Three samples from $u \sim \mu_\ell^y$ per each level; from top to bottom $\ell = 0, 1, 2, 3$. . 177
5.17 Plots of $\mathbb{E}_{\nu_\ell}[Y_\ell]$ and $\mathbb{V}_{\nu_\ell}[Y_\ell]$ (in $\log_2$-scale) vs level for the high-dimensional example. 178
5.18 Joint samples of $(\mathsf{QoI}_{\ell-1}, \mathsf{QoI}_\ell)$ for different levels $\ell = 1, 2, 3$. 178
5.19 Costs for the high-dimensional example. Left: number of samples vs level for different tolerances. Right: complexity of ML-MCMC vs a single-level MCMC estimator. 179

6.1 Depiction of a two-dimensional small set. 193
6.2 Histograms of samples obtained with a (left) maximal coupling of the proposals (center) with a sub-sampling algorithm and (right) with the independent Metrpolis-Hastings algorithm, for different pairs of accuracy levels. Each histogram is obtained with 20000. 196
6.3 Average synchronization rate for the chains generated with maximally coupled proposals (blue) those generated by the sub-sampling algorithm (orange), and those obtained with IMH (burgundy). 95% confidence intervals are shown in dashed lines. 197
6.4 Mean multi-level estimator $\log_2\left(|\overline{\mathsf{QoI}}_\ell^{(\omega)}|\right)$, using the sub-sampling re-synchronization kernel. Estimates where computed for each value of ω, from 50 independent runs. 199
6.5 Histograms of the samples from μ_ℓ^y obtained with our Algorithm. Each row corresponds to a different value of $\omega = 0.1$ (top), $\omega = 0.5$ (middle), $\omega = 0.7$ (bottom). 199

6.6 Mean synchronization rate for different values of ω and different synchronization kernels. (Left) re-sync. via sub-sampling kernel. (Right) re-sync using IMH kernel. Dashed lines represent 95% confidence intervals. 200
6.7 (Left) true log-field $\log(\kappa(x, u^*))$. (Right) Posterior mean estimator of $\log(\kappa(x, u))$ at level $\mathsf{L} = 4$. 202
6.8 Diagonal plots of QoI_ℓ vs $\mathsf{QoI}_{\ell-1}$ for $\ell = 1, 2, 3, 4$. 203
6.9 (Left) Estimated synchronization rate vs level. (Right) Mean autocorrelation plot (ACF) at lag 100. In both plots, dashed lines represent a 95% confidence interval obtained over 50 independent runs. 204
6.10 Convergence vs level. (Left) weak convergence, (right) strong convergence (log_2 scale in y axis). Dashed lines represent a 95% confidence interval obtained over 50 independent runs. 204
6.11 (Left). Number of samples $N_\ell(\mathsf{tol}_i)$ for a given tolerance tol_i, $i = 1, 2, 3, 4$. (Right). Comparison of the cost against a single-level MCMC algorithm. . . . 205
6.12 Rates for cost-tolerance theorem for the high-dimensional example with maximal-coupling. ($\log_2$-scale in the y-axis) 207
6.13 Diagonal plots of $\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1})$ 208
6.14 (Left) Number of samples per level for different tolerances. (Right). Total computational cost vs tolerance of ML-MLMC and single-level MCMC. 208

7.1 Histograms of samples for different ; from top to bottom: $\ell = 0, 1, 2$. 220
7.2 diagonal plots of samples $(u_{\ell,\ell-1}, u_{\ell,\ell})$ for different levels level; from top to bottom: $\ell = 0, 1, 2$. 221
7.3 (Left). Samples from $\mathcal{N}(0, I_{17\times 17})$. (Middle) posteriors samples form a subsurface flow BIP. (Right) Samples obtained with a normalizing flow. 222

List of Tables

4.1 Step size of the RWM proposal distribution for the manifold experiment. . . . 109
4.2 Results for the density concentrated around a circle-shaped manifold experiment. As we can see, both GPT algorithms provide an improvement over PT, PSDPT and RWM. The computational cost is comparable across all algorithms. 109
4.3 Step size of the RWM proposal distribution for the elliptic BIP experiment. . . 114
4.4 Results for the elliptic BIP problem. The computational cost is comparable across all algorithms, given that the cost of each iteration is dominated by the cost of solving the underlying PDE. 114
4.5 Step size of the RWM proposal distribution for the Cauchy BIP experiment. . 116
4.6 Results for the 1D Cauchy BIP problem. The computational cost is comparable across all algorithms. 116
4.7 Step size of the RWM proposal distribution for the acoustic BIP experiment. Here Diag$(d_1, d_2, \ldots, d_N)$ is to be understood as the $N \times N$ diagonal matrix with entries $d_1, d_2, \ldots, d_N$. 119
4.8 Results for the acoustic BIP problem. Once again, we can see that both GPT algorithm provide an improvement over RWM, PT and PSDPT. The computational cost is comparable across all algorithms, given that the cost of each iteration is dominated by the cost of solving the underlying PDE. 119
4.9 Values of ρ_k for the pCN kernel for the high-dimensional wave inversion problem. 122
4.10 Results for the high-dimensional acoustic BIP problem. As for the previous examples, The computational cost is comparable across all algorithms. 123
4.11 Set of true parameters, by which the data are synthetically generated, approximated to the closest unit. 125

Contents

I Introduction

Computational simulations, together with the mathematical algorithms that drive them, have rapidly become a central part of the scientific paradigm over the last several decades. Indeed, these approaches greatly complement the relationship between *theory and experimentation* in the sciences, with such techniques being at the core of the design, prediction, and optimization of a multitude of processes and phenomena arising in the natural sciences and engineering. Such is the case of *Uncertainty Quantification* (UQ), understood to be the field of knowledge tasked with quantifying and controlling the sources of uncertainty associated to a given natural phenomenon, an engineering process, an estimation or learning procedure, and which, at its core, relies heavily upon mathematical, computational and experimental techniques [58, 74, 147, 168]. In the context of this thesis, we will focus on what is often referred to as *inverse UQ*, where, given a set of experimental measurements of a process together with a computational model describing it, one is tasked with (i) estimating the discrepancy between the measured and simulated data and (ii), estimating the uncertainty in the unknown parameters that could have generated the data, the latter of which will be the focus of this thesis. This problem of parameter identification can be understood in a Bayesian sense, usually referred to as a *Bayesian Inverse Problem* (BIP). In a rather informal way (we will present this more precisely in the following), using the symbols u and y to denote parameters and data, respectively, together with the symbol $\mathbb{P}[\cdot]$ to denote probability, and assuming that both u and y are random variables, the solution to a BIP can be understood (in a broad sense) as the process of obtaining information from the probability distribution $\mathbb{P}[u \text{ given } y]$, which in light of Bayes theorem (c.f. Theorem 2.2.1 for a rigorous statement of this theorem) can be written as

$$\mathbb{P}[u \text{ given } y] = \frac{\mathbb{P}[y \text{ given } u] \times \mathbb{P}[u]}{\mathbb{P}[y]},$$

where informally, $\mathbb{P}[y \text{ given } u]$, quantifies how *likely* it was to obtain the data y for a given u, $\mathbb{P}[u]$ encodes the *prior* information or knowledge on u before data was observed, and $\mathbb{P}[y]$ can be understood as a term describing the information contained in the data y.
One way of extracting such an information is by sampling from $\mathbb{P}[u \text{ given } y]$. Although there are several different approaches to perform this task (c.f. Section 2.3), in this thesis we will focus on a class of algorithms known as *Markov Chain Monte Carlo* (MCMC). Modern computational facilities and recent advances in computational techniques have made the use of MCMC methods feasible for many Bayesian Inverse Problems. However, for some *large-scale* applications in physics

or engineering, which often involve differential models, the computational cost associated with a Bayesian inversion procedure by MCMC, when seen as

$$\text{Cost} = \text{Number of samples} \times \text{Cost per sample},$$

can still be prohibitively expensive.
In this thesis, we present, analyze, and implement several novel *hierarchical* MCMC techniques for the acceleration of such *large-scale* BIPs. In the context of this work, we say that a BIP is a *large-scale* problem if either (i) the evaluation of the *likelihood*, denoted by $\mathbb{P}[y \text{ given } u]$, is deemed to be computationally expensive, and involves large-scale computations, such as the solution of a non-linear or time-dependent Partial Differential Equation (PDE), approximated on a sufficiently fine grid, or (ii) those for which the parameter space is high dimensional, such as BIP on random fields discretized on a fine grid, or more realistically, when both (i) and (ii) hold. By *hierarchical methods* we mean the set of techniques that exploit a sequence of approximations of the probability measure of interest, with given accuracy and which are possibly easier to sample from. This can be understood in terms of a hierarchy of discretizations of the underlying mathematical model, in the spirit of Multi-level Monte Carlo [59, 30, 31, 66], or as a hierarchy of so-called temperatures, in the spirit of parallel tempering [52, 90]. We will be more precise about what we mean by "hierarchies" in Section 1.3, and present such methods in further detail in upcoming chapters, which are based upon the following works:

[95] Latz, J., Madrigal-Cianci, J. P., Nobile, F., & Tempone, R. (2021). Generalized parallel tempering on Bayesian inverse problems. *Statistics and Computing, 31(5), 1-26.*

[108] Madrigal-Cianci, J. P., Nobile, F., & Tempone, R. (2021). Analysis of a class of Multi-Level Markov Chain Monte Carlo algorithms based on Independent Metropolis-Hastings. *ArXiv preprint arXiv:2105.02035.* (Submitted for publication).

[107] Madrigal-Cianci, J. P., & Nobile, F. (2021). Multi-Level Markov Chain Monte Carlo algorithms based on maximally-coupled proposals. *In preparation*.

The rest of this introductory chapter is organized as follows. In Section 1.1 we present the uncertainty quantification framework and introduce the notion of inverse problems. We briefly present the two main paradigms used to solve such problems, namely the *frequentist's* (or *deterministic*) and the *Bayesian* approach, and make a case for the need of the latter for the types of applications that are addressed in this work. We then present two large-scale model BIPs that will be studied throughout this thesis in Section 1.2 and argue about the necessity of hierarchical methods to tackle them effectively. In section 1.3 we present a literature review of the state of the art of hierarchical MCMC methods and discuss the main contributions of this thesis. Lastly, we present the outline for the rest of the thesis in Section 1.4.

1.1 Uncertainty quantification and inverse problems

Broadly speaking, Uncertainty Quantification (UQ) is the scientific discipline tasked with determining appropriate uncertainties associated with model-based predictions [58]. In general, these models are subject to different sources of uncertainty; including, uncertainties in the model inputs parameters (such as unknown material properties, forcing terms, initial or boundary conditions), observation error, uncertainties in the mathematical model itself, among others. Being able to accurately and efficiently quantify these uncertainties is of paramount importance in many fields of science and engineering.

UQ can be classified into two main approaches: *forward* and *inverse* UQ. On the one hand, in forward UQ one aims at assessing the impact of uncertain input parameters in the model output, usually taken to be a physical Quantity of Interest (QoI), and understood to be a function of this uncertain input. To that end, the input parameters u are modeled as random variables with known distribution μ_{pr}, and one is then interested in quantifying the effects of this forward propagation of uncertainty for the QoI through the mapping $u \mapsto \mathsf{QoI}(u)$, $u \sim \mu_{\mathsf{pr}}$, which typically involves the solution of a complex differential problem. This is done by estimating statistical properties of QoI, such as its moments, or the probability of QoI exceeding a given threshold value, usually written in terms of expectations under μ_{pr}. The literature on numerical methods for forward UQ is vast, see e.g., [58] and the references therein. When u is a high (or even infinite) dimensional parameter, arguably the most straightforward approach to solving this type of problem is the *Monte Carlo method* [3], where these expectations over μ_{pr} are approximated by first sampling N independently and identically distributed (iid) realizations of u, and then estimating the effects of the forward propagation of uncertainty $u \mapsto \mathsf{QoI}(u)$ with the usual Monte Carlo average over these N realizations. Monte Carlo methods have been in active development for the last several decades. Of particular relevance to the work outlined in this thesis are *Multi-level Monte Carlo methods* [31, 59, 69, 118], a set of variance reduction techniques [3] which can greatly reduce the computational cost associated with plain Monte Carlo by introducing a hierarchy of discretization *levels* of the underlying differential mathematical model with increasing accuracy and cost, and performing most simulations with low accuracy (and hence cost), with relatively few simulations being performed with the highly accurate, computationally expensive model, in such a way that the final accuracy of the estimator is equivalent to that of using plain Monte Carlo at the finest discretization level, albeit with an overall much lower complexity.

On the other hand, in Inverse UQ [85, 158], one is instead interested in characterizing and reducing the uncertainty on the input parameters of the model, based on some available, noise-polluted, experimental data, assumed to have been obtained from the underlying physical process (c.f. Figure 1.1). We now proceed to formalize this idea. Let $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$ and $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}})$ be two separable Banach spaces with associated Borel σ-algebra $\mathcal{B}(\mathsf{X})$, $\mathcal{B}(\mathsf{Y})$. We will refer to X as the *parameter space* and to Y as the *data space*, and define the *forward mapping operator* $\mathcal{F} : \mathsf{X} \to \mathsf{Y}$ as a mapping between these two spaces. Broadly speaking, given some recorded, potentially noise-polluted data $y \in \mathsf{Y}$, the goal of an inverse problem is to *characterize* (we will be more precise about what we mean by

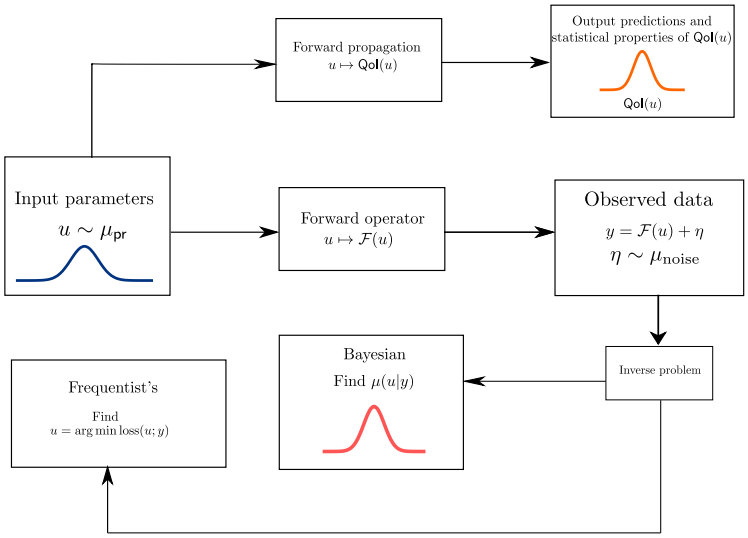

Figure 1.1: Depiction of forward and inverse problems.

this statement shortly after) the set of unknown, possibly infinite-dimensional parameters $u \in \mathsf{X}$ that could have generated the data y, where the relationship between y and u is given by

$$y = \mathcal{F}(u) + \eta, \quad \eta \sim \mu_{\text{noise}},$$

with $\eta \in \mathsf{Y}$ some additive noise with known distribution μ_{noise} on $(\mathsf{Y}, \mathcal{B}(\mathsf{Y}))$. In our context, the forward mapping operator $\mathcal{F}$ is to be understood as a mathematical model of the physical process that generated the data y, which is based on possibly non-linear and/or time-dependent PDEs.
Characterizing these input uncertainties can be achieved by two main paradigms, a frequentist's (also called *deterministic* or *classical*) approach [158] or a Bayesian approach [156], the latter of which will be the focus of this work. We present a brief overview of these methods in the following paragraphs, and a more thorough review of the Bayesian approach to inverse problems in the following Chapter.
We begin with a brief description of the frequentist's approach. For simplicity, suppose $y = (y_1, y_2, \ldots, y_M) \in \mathbb{R}^M$. In the context of this work we will assume that y is generated by a single realization of the underlying physical phenomena, which is observed at M different points in space or time, with $\{y_i\}_{i=1}^M$, corresponding to the set of observed values. In addition, consider a loss function $\mathsf{loss} : \mathsf{Y} \times \mathsf{X} \to \mathbb{R}_+$, measuring, in some sense, the misfit between the recorded data y and $\mathcal{F}(u)$ for some given u. This loss function can be, e.g.,

1. (squared error) $\mathsf{loss}(y, u) = \frac{1}{2} \sum_{i=1}^M |y_i - [\mathcal{F}(u)]_i|^2$

2. (absolute error) $\mathsf{loss}(y, u) = \sum_{i=1}^M |y_i - [\mathcal{F}(u)]_i|$.

where $[\mathcal{F}(u)]_i$ corresponds to the component of $\mathcal{F}(u)$ associated to the i^{th} measurement y_i. Let $u^* \in \mathsf{X}$ be the solution to the following optimization problem:

$$\begin{aligned} &\textit{find } u^* \in \mathsf{X} \textit{ that minimizes } J_\alpha(u) \\ &J_\alpha(u) := \mathsf{loss}(y, u) + \frac{\alpha}{2}\mathrm{reg}(u), \quad \alpha > 0, \end{aligned} \tag{1.1}$$

where the second term in (1.1), $\mathrm{reg} : \mathsf{X} \to \mathbb{R}_+$ is known as a the *regularization term* and it is usually included to improve the regularity and enforce well-posedness of the inverse problem (1.1) [85, 158]. A common choice for regularization parameter is the so-called *Tykhonov regularization* [158] given by

$$\mathrm{reg}(u) := \|u - u_0\|_{\mathsf{X}}^2, \quad u_0 \in \mathsf{X},$$

for some carefully chosen $u_0 \in \mathsf{X}$. Loosely speaking, this choice of regularization penalizes values of u that are *far* (in the X-norm) from u_0. The frequentist's approach to inverse UQ consists in first solving Problem (1.1), usually obtained using numerical optimization algorithms, see [119], and then using arguments and assumptions on $y, \eta, \mathcal{F}$, and u^*, proper of frequentist statistics (such as large amounts of data, normality and independence of the components of η, etc) to construct $(1-a)\%$, $a \in (0,1)$, confidence intervals [26, 155]. Furthermore, one can also use the *parametric bootstrap method* [6, 117] in order to do uncertainty quantification with this approach. Such a technique is a Monte Carlo method that estimates parameter uncertainty by repeatedly resampling observations and computing corresponding parameter estimates. This is achieved by repeatedly solving the (randomized) minimization problem

$$u^n = \arg\min_{u \in \mathsf{X}} \mathsf{loss}(y + \eta^n, u) + \frac{\alpha}{2}\mathrm{reg}(u), \quad \eta^n \sim \mu_{\text{noise}},\ n = 1, 2, \ldots,$$

which in some special cases can leads to samples $\{u^n,\ n = 1, 2, \ldots\}$ from the posterior distribution arising from the Bayesian approach (see, e.g., [6] for a precise statement).
There are, however, certain drawbacks associated to this method:

1. Although the use of a regularization aims at guaranteeing the existence and uniqueness of solutions to the problem (1.1) (see, e.g., [158]), the cost functional $J_\alpha(u)$ could still suffer from multiple local minima, and as such, the numerical optimization techniques used to minimize $J_\alpha(u)$ could potentially converge to a sub-optimal solution.

2. In general, the sample distribution of u obtained using the parametric bootstrap approach is not the posterior distribution induced by the Bayesian approach (c.f. next paragraph and [6]).

On the Bayesian paradigm, we model u, η and y as random variables, and aim at obtaining the probability distribution of u conditioned on y. For the sake of exposition, we briefly present such an approach in rather general terms in the following, and will present a detailed overview

of it in Chapter 2. In the Bayesian paradigm, one assumes that u follows a *prior* distribution μ_{pr} encoding the information available on u before any data is observed. Notice that this is a natural way of including *expert* information about u on the inversion procedure. Under the assumptions (i) $u \sim \mu_{\mathsf{pr}}$ before any data is observed, (ii) η and u are independent random variables, and (iii), $\mu_{\text{noise}}(\cdot)$ and $\mu_{\text{noise}}(\cdot - \mathcal{F}(u))$ (i.e., the measure μ_{noise} translated by $\mathcal{F}(u)$) have a density $\tilde{\mu}_{\text{noise}} : \mathsf{X} \to \mathbb{R}_+$ with respect to some dominating probability measure, one then has that $y|u$ has the same distribution as $\mu_{\text{noise}}(\cdot - \mathcal{F}(u))$, which allows us to define the *potential* function $\Phi(u; y) : \mathsf{X} \times \mathsf{Y} \mapsto \mathbb{R}$ as

$$\Phi(u; y) = -\log\left[\tilde{\mu}_{\text{noise}}(y - \mathcal{F}(u))\right],$$

where the function $\Phi(u; y)$ is a measure of the misfit between the recorded data y and the predicted value $\mathcal{F}(u)$, and often depends on $\|y - \mathcal{F}(u)\|_{\mathsf{Y}}$. Applying Bayes' theorem [94, 156], one can then pose the solution to the BIP as approximating the *posterior probability measure* μ^y given in terms of its Radon-Nikodym derivative with respect to the prior by

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u) = \frac{1}{Z} e^{-\Phi(u;y)}, \quad Z = \int_{\mathsf{X}} e^{-\Phi(u;y)} \mu_{\mathsf{pr}}(\mathrm{d}u). \tag{1.2}$$

Once such a posterior probability measure has been suitably approximated, one can compute expectations of a given μ^y-integrable quantity of interest $\mathsf{QoI} : \mathsf{X} \to \mathbb{R}$, i.e.,

$$\mathbb{E}_{\mu^y}[\mathsf{QoI}] = \int_{\mathsf{X}} \mathsf{QoI}(u) \mu^y(\mathrm{d}u).$$

Furthermore, one could, e.g., estimate moments of u (provided they exists), visualize its distribution, etc. This is in stark contrast to the deterministic paradigm, in the sense that the Bayesian approach provides a larger amount of information about u.

Remark 1.1.1 (On the drawbacks of the Bayesian approach): *There are, of course, some drawbacks associates to the Bayesian approach. We identify two of them in the following:*

1. *As we shall discuss shortly after (and throught this thesis), one way of approximating μ^y is by sampling from it, using, e.g., MCMC methods. This will lead, in general, to repeated evaluations of the forward mapping $\mathcal{F}$, which will in turn result in an overall more expensive inversion procedure.*

2. *The Bayesian formulation is heavily-dependent on the choice of prior, which is, in turn, subjective. Choosing an appropriate prior is delicate, as a completely misspecified prior will in turn lead to erroneous results. Furthermore, the construction of credible regions rely upon the choice of prior; thus, a poorly chosen prior might compromise the interpretability of the results.*

As previously mentioned, the approximation of μ^y is usually done by sampling from it. We will present a detailed survey of commonly-used methods to generate samples (approximately)

distributed according to μ^y in Chapters 2 and 3. Perhaps the most powerful and robust tools for this task are MCMC methods, where one generates samples that are (asymptotically) distributed according to μ^y by creating a Markov chain having μ^y as its invariant probability measure. The one drawback of these methods is that, in general, they require a large number of samples to converge. Furthermore, MCMC methods usually require an evaluation of the forward mathematical model for each sample. Thus, for those BIP for which the underlying forward mapping operator $\mathcal{F}$ is already costly to evaluate, as those considered in this work, MCMC methods can rapidly become prohibitively expensive. In the next section, we present two large-scale model problems that will be studied throughout this thesis.

1.2 Model problems in geophysics

1.2.1 Subsurface flow

Our first model problem is the inversion of parameters arising in a steady-state subsurface flow model. In this case, we are interested in characterizing the geophysical properties of an aquifer, given some noise-polluted measurements of the hydraulic head p throughout the domain. More formally, given a physical domain $D \subset \mathbb{R}^d$, $d = 1, 2, 3$, with boundary $\partial D = \Gamma_N \cup \Gamma_D$, $\mathring{\Gamma}_N \cap \mathring{\Gamma}_D = \emptyset$, the hydraulic head p of the aquifer follows Darcy's subsurface flow equation given by

$$\begin{cases} -\nabla \cdot (\kappa(x,u)\nabla \mathsf{p}(x,u)) = f(x,u), & x \in D,\ u \in \mathsf{X}, \\ \mathsf{p}(x,u) = G_D(x,u), & x \in \Gamma_D,\ u \in \mathsf{X}, \\ \partial_n \mathsf{p}(x,u) = G_N(x,u), & x \in \Gamma_N,\ u \in \mathsf{X}, \end{cases} \tag{1.3}$$

with u representing the possible sources of uncertainty. Here, $\kappa(x,u)$ represents the random permeability field in the aquifer (typically modeled as a log-normal random field), $f(x,u)$ represents a potentially unknown source term, and $G_N(x,u)$, $G_D(x,u)$ represent the (also potentially unknown) Neumann and Dirichlet boundary conditions of the model, respectively. Thus, given some noise polluted measurements of $\mathsf{p}(x,u)$ at given locations in D, one aims at characterizing one or more of $\kappa(x,u)$, $f(x,u)$, $G_N(x,u)$ or $G_D(x,u)$. In this case, the mapping $u \mapsto \mathcal{F}(u)$ can be understood as the solution of (1.3), observed at the location of the measurements. This is a common inverse problem in the management and risk analysis of radioactive waste material [81, 156, 157] and oil reservoir exploration [16, 121, 156].

1.2.2 Seismic inversion

A second example of a large-scale BIP is that of seismic inversion. In this case, given a set of recordings of the the displacement[1] of a seismic wave at different points in space and instants

[1] In practice, other measurable quantities can be considered as well, such as wave velocity or acceleration, see e.g., [1].

in time, one aims at characterizing (i) the physical properties of the earthquake, such as source location or moment tensor and/or (ii) the physical properties of the medium, such as its material densities or Lamé parameters. More precisely, consider once again a physical domain $D \subset \mathbb{R}^d$, $d = 2, 3$ and a time interval $I = [0, T]$, $T > 0$. We will model the wave propagation of an earthquake using either an elastic or an acoustic wave equation. For the first case, the forward model of the wave phenomena reads as *find a displacement field* $w : I \times D \times \mathsf{X} \to \mathbb{R}^d$ *such that:*

$$\begin{cases} \rho(x,u)w_{tt}(t,x,u) - \nabla \cdot \sigma(x,u,w) = f_{\text{el}}(t,x,u), & \text{in } I \times D \times \mathsf{X}, \\ w(0,x,u) = g_{1,\text{el}}(x,u),\ w_t(0,x,u) = g_{2,\text{el}}(x,u), & \text{on } \{t=0\} \times D \times \mathsf{X}, \end{cases} \tag{1.4}$$

where

$$\sigma(x,u,w) = \lambda(x,u)\nabla \cdot wI + m(x,u)(\nabla w + (\nabla w)^T),$$

together with suitable boundary conditions. Here, $\rho(x,u)$, represents the density of the material, $\lambda(x,u), m(x,u)$ represent the Lamé parameters, and $g_{i,\text{el}}(x,u)$, $i = 1, 2$, are the initial conditions. In the case where one considers the earthquake to be a point source (i.e., an explosion), the forcing term takes the form [1]

$$f_{\text{el}}(t,x,u) = -M(u) \cdot \nabla\delta(x - u_s)S(t,u_s), \tag{1.5}$$

where δ denotes the Dirac mass, $M(u) \in \mathbb{R}^{d\times d}$ represents the moment tensor of the earthquake, $\mathbb{R}^d \ni u_s \subset u$ represents the spatial location of the source and $S(\cdot,u) : I \to \mathbb{R}$ represents the time component of the forcing term (usually a Gaussian or Rickert wavelet parametrized by u)[1]. In practical computations, often a regularized version of (1.5) is considered, obtained by replacing $\delta(x - u_s)$ by e.g., $(|a|\sqrt{\pi})^{-1}\exp(-(\|x - u_s\|_2/a)^2)$, for some $|a| \ll 1$.
Alternatively, for the case where one models the forward wave propagation using an acoustic wave, we have that the forward model reads *find the acoustic pressure* $w : I \times D \times \mathsf{X} \to \mathbb{R}$ *such that*

$$\begin{cases} \rho(x,u)w_{tt}(t,x,u) - \nabla \cdot (\beta(x,u)\nabla w(t,x,u)) = f_{\text{ac}}(t,x,u), & \text{in } I \times D \times \mathsf{X} \\ w(0,x,u) = g_1(x,u),\ w_t(0,x,u) = g_2(x,u) & \text{on } \{t=0\} \times D \times \mathsf{X}, \end{cases}$$

together with suitable boundary conditions. Once again $\rho(x,u)$ represents the density of the medium and $\beta(x,u)$ is related to the acoustic wave velocity $c(x,u)$ in the medium by $\beta(x,u) = c^2(x,u)\rho(x,u)$. Furthermore, in this case, we model f_{ac} as

$$f_{\text{ac}}(t,x,u) = \delta(x - u_s)S(t,u),$$

or a regularized version of it. In either case, given some measurements of the wavefield at different points in time and space, we aim at obtaining the material properties (e.g., ρ, β, λ, m), assuming the source is known (which is known in the literature as *seismic imaging*), or, alternatively, we aim at

recovering the source location u_s and other parameters related to the source term, with an additional potential uncertainty in the material properties of the medium (known as *seismic source inversion*). For this problem the mapping $u \mapsto \mathcal{F}(u)$ can be understood as the displacement of the wavefield, observed at several points in the physical domain, at different moments in time. Seismic inversion (whether seismic imaging or source inversion) is of great importance to the seismology community and it is an active field of research (see e.g., [23, 75, 170, 162]). However, the computational cost associated with the evaluation of the forward computational model, together with possible multi-modalities arising in the associated posterior, motivates the development of efficient inversion techniques.

1.3 Literature review and contributions of this thesis

The overall aim of this thesis is to present, analyze, and implement hierarchical Markov chain Monte Carlo methods for accelerating large-scale Bayesian inverse problems, with a particular focus on BIPs arising in geophysics, such as those presented in Section 1.2. In this section, we introduce two types of hierarchical MCMC methods that are central to our work, we present a literature review of these methods, and state the main contributions of this thesis. Such contributions will be the subject of Chapters 4-6.

For most problems of interest, involving complex PDE models, it is often the case that one can not solve the underlying mathematical model (and hence, evaluate $\mathcal{F}$) exactly, and as such its solution needs to be approximated using numerical methods, such as finite elements (FE) or finite differences (FD). We denote by $\mathcal{F}_{\mathsf{L}}$ the numerical approximation of the forward map at an accuracy level L. Notice that this induces a *discretized potential* $\Phi_{\mathsf{L}}(u; y)$, which in turn induces a *discretized posterior measure*[2]

$$\mu^y_{\mathsf{L}}(\mathrm{d}u) = \frac{1}{Z_{\mathsf{L}}} \exp\left(-\Phi_{\mathsf{L}}(u; y)\right) \mu_{\mathsf{pr}}(\mathrm{d}u).$$

Under reasonable conditions (see [156]), one has that (in a suitable sense) $\mu^y_{\mathsf{L}} \to \mu^y$ as $\mathsf{L} \to \infty$. Given (i) the potentially multi-scale effects of the material properties (in both Problems 1.2.1 and 1.2.2) and (ii) computational restrictions on the forward model, such as the Courant-Friedrichs-Lewy (CFL) condition for Problem 1.2.2 (see e.g., [135]), the forward model $\mathcal{F}_{\mathsf{L}}$ must be approximated using a sufficiently fine grid, together with a sufficiently small time-step for the time-discretization. This in turn makes the computation of either forward problem extremely expensive, specially in the case where $d = 3$. Although this computational cost can be reduced by, e.g., using domain decomposition and other advanced techniques for the PDE solver, the cost associated to an evaluation of the forward model can still be quite large. In addition, posterior exploration via MCMC methods, requires, in general, a large number of samples in order to obtain meaningful and accurate results. Furthermore, when targeting posterior probability measures that are multi-modal or that concentrate around a lower dimensional non-linear manifold, as it is

[2] by dscretized posterior we reffer to a posterior measure associated to a discretized forward model, and this should not be confused with a posterior measure on a discrete state space

often the case for seismic inversion, the MCMC algorithm will typically require a larger number of samples, thus further increasing the computational cost associated to the Bayesian inversion. One way of overcoming these issues is with the use of *hierarchical models* on the posterior measure. Given an ordered set $\mathcal{J} = [1, 2, \dots, J]$, $J \in \mathbb{N}$, let $\{\mu_j^y, j \in \mathcal{J}\}$, be a family of approximations to μ^y with the following properties:

1. $\mu_J^y = \mu_{\mathsf{L}}^y$, and $\mu_J^y \to \mu^y$ as $J \to \infty$.

2. For any $j \in \mathcal{J}$, sampling from μ_j^y is either easier (in some sense) or cheaper than sampling from μ_{j+1}^y.

By exploiting properties 1 and 2, one can create novel sampling algorithms that can drastically reduce the cost associated to BIP. In particular, we will present algorithms based on hierarchies of temperatures (c.f. Chapter 4) and discretizations (c.f. Chapters 5-6).

1.3.1 Tempering

Tempering methods: literature review

In the Tempering case, we construct the hierarchy of models by introducing an increasing sequence of temperatures $1 = T_1 < T_2 < \cdots < T_J \leq \infty$, which induces the following sequence of posterior probability measures:

$$\mu_j^y(\mathrm{d}u) = \frac{1}{Z_j} \exp\left(-\frac{\Phi_{\mathsf{L}}(u; y)}{T_{J-j}}\right) \mu_{\mathsf{pr}}(\mathrm{d}u), \quad j = 0, \dots, J-1,$$

with the convention that if $T_J = \infty$, $\mu_1^y(\mathrm{d}u) = \mu_{\mathsf{pr}}(\mathrm{d}u)$. This hierarchy is specially useful in cases where the posterior measure μ^y is multi-modal or concentrates around a non-linear, lower-dimensional manifold. Indeed, the temperature term acts as an "inflation" parameter on $\frac{\mathrm{d}\mu_j^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u)$, which in turn makes the posterior μ_j^y easier to explore using traditional MCMC algorithms. This is depicted in Figure 1.2, where the un-normalized density of a target posterior $\mu^y = \mu_2^y$ is shown together with two of its tempered, un-normalized counterparts μ_0^y, μ_1^y. As it can be seen, μ_2^y is strongly concentrated around two well-separated peaks, while the peaks μ_0^y, μ_1^y present a larger overlap. A consequence of this, is that localized MCMC algorithms, such as Random Walk Metropolis (RWM) or Preconditioned Crank-Nicolson (pCN) (c.f. chapter 3) can "explore" μ_0^y, μ_1^y faster than μ_2^y, since jumping from one mode to the other using localized proposals (i.e., very small steps in comparison to the separation of the peaks) is, in practice, quite unlikely for μ_2^y, but much more likely for μ_0^y, μ_1^y.
Once this hierarchy has been introduced, the idea is then to sample from the joint posterior $\mu_1 \times \cdots \times \mu_J$ on the extended space $(\mathsf{X} \times \cdots \times \mathsf{X}, \mathcal{B}(\mathsf{X} \times \cdots \times \mathsf{X}))$ using a joint Markov transition kernel (c.f. Definition 3.1.2 and Equation (5.3)) to advance each "component" of the joint chain, together with a *swapping kernel*, which mixes the components between chains, thus

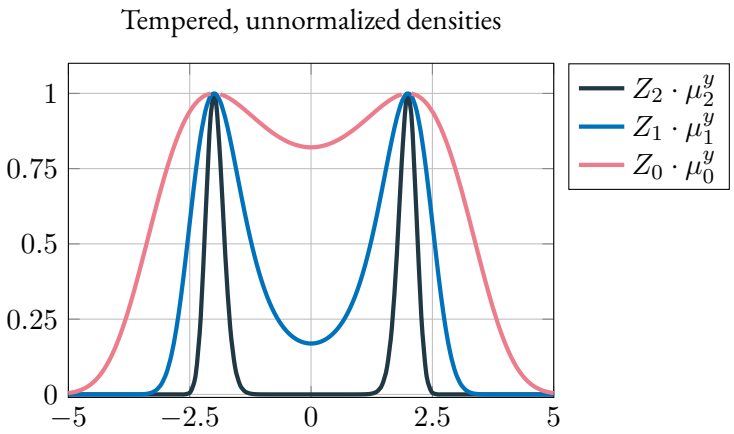

Figure 1.2: Depiction of an un-normalized posterior density of interest, $\mu^y = \mu_2^y$, together with 2 un-normalized tempered versions μ_0^y, μ_1^y.

providing an opportunity for each individual chain to better explore the parameter space. This hierarchical approach is done in the spirit of parallel tempering [52, 90, 114, 171].
In recent years, there has been an active development of computational techniques and algorithms to overcome the issues associated with sampling multi-modal measures, or those that concentrate around a non-linear, lower-dimensional manifold using several *tempering strategies* [42, 52, 96, 114, 167]. Of particular importance for the work presented here is the Parallel Tempering (PT) algorithm [52, 90, 114] (also known as *replica exchange*), which finds its origins in the physics and molecular dynamics community. The general idea behind such methods is to simultaneously run J independent MCMC chains, where each chain is invariant with respect to a *flattened* (referred to as *tempered*) version of the posterior of interest μ^y, while, at the same time, proposing to swap states between any two chains every so often. Such a swap is then accepted using the standard Metropolis-Hastings (MH) acceptance-rejection rule. Intuitively, chains with a larger smoothing parameter (referred to as *temperature*) will be able to better explore the parameter space. Thus, by proposing to exchange states between chains that target posteriors at different temperatures, it is possible for the chain of interest (i.e., the one targeting μ^y) to mix faster, and to avoid the undesirable behavior of some MCMC samplers of getting "stuck" in a mode. Moreover, the fact that such an exchange of states is accepted with the typical MH acceptance-rejection rule, will guarantee that the chain targeting μ^y remains invariant with respect to such probability measure [52].
Tempering ideas have been successfully used to sample from posterior distributions arising in different fields of science, ranging from astrophysics to machine learning [41, 52, 114, 163]. The works [106, 171] have studied the convergence of the PT algorithm from a theoretical perspective and provided minimal conditions for its rapid mixing. Moreover, the idea of tempered distributions has not only been applied in combination with parallel chains. For example, the simulated tempering method [109] uses a single chain and varies the temperature within this chain. In addition,

tempering forms the basis of efficient particle filtering methods for stationary model parameters in Sequential Monte Carlo settings [10, 11, 84, 86, 96] and Ensemble Kalman Inversion [34].
A generalization over the PT approach, originating from the molecular dynamics community, is the so-called *Infinite Swapping (IS)* algorithm [49, 133]. As opposed to PT, this IS paradigm is a continuous-time Markov process and considers the limit where states between chains are swapped infinitely often. It is shown in [49] that such an approach can in turn be understood as a swap of dynamics, i.e., kernel and temperature (as opposed to states) between chains. We remark that once such a change in dynamics is considered, it is not possible to distinguish particles belonging to different chains. However, since the stationary distribution of each chain is known, importance sampling can be employed to compute posterior expectations with respect to the target measure of interest.

Tempering methods: contributions

Infinite Swapping has been successfully applied in the context of computational molecular dynamics and rare event simulation [47, 50, 103, 133], however, it was only until our work [95] that an analogous version of this methods was formulated and implemented in the context of Bayesian Inverse Problems (which are, inherently discrete-time in nature). We present such a work in Chapter 4, where our contributions can be summarized as follows:

1. We present two generalizations of the Parallel Tempering algorithm, inspired by the so-called continuous-time Infinite Swapping algorithm of [47].

2. We provide a solid theoretical analysis of the convergence of such methods. In particular, we show that such algorithms are reversible and geometrically ergodic under some mild conditions.

3. We implement our proposed methods, together with several competing methodologies, and use them to solve an array of increasingly difficult Bayesian inverse problems. Our experimental results suggest a significant improvement with respect to competing methodologies.

We believe these methods present sufficient innovation such that the current work can be extended into multiple future works, both from a theoretical and computational perspective, as will be discussed in Chapter 7.

1.3.2 Multi-level methods

Multi-level methods: literature review

Multi-Level Monte Carlo (MLMC) methods are well-known computational techniques [59] used to compute expectations that arise in stochastic simulations in cases in which the stochastic model cannot be simulated exactly, but can be approximated at different levels of accuracy and different computational costs. Despite their wide-spread applicability, extending these MLMC

ideas to Multi-Level Markov Chain Monte Carlo (ML-MCMC) methods to compute expectations with respect to (w.r.t) a complex target distribution from which independent (whether exact or approximate) sampling is not accessible, has only recently been attempted, with only a handful of works dedicated to this task. This situation arises, for instance, in Bayesian inverse problems (BIPs) where the aim is to compute the expectation $\mathbb{E}_{\mu^y}[\mathsf{QoI}]$ of some output quantity of interest QoI. At their core, ML-MCMC methods for BIPs introduce a hierarchy of discretization levels $\ell = 0, 1, \ldots, \mathsf{L}$ of the underlying forward operator $\{\mathcal{F}_\ell\}_{\ell=0}^{\mathsf{L}}$, with increasing accuracy and cost to evaluate it, which, consequently, induces a family of posterior probability measures μ_ℓ^y, approximating μ^y with increasing levels of accuracy as $\ell \to \infty$. This hierarchy of forward mapping operators is depicted in Figure 1.3, where the mesh, the random field κ in (1.3), and the forward mapping operator[3] $\mathcal{F}_\ell(\kappa(x,u))$ with p as in (1.3) is shown at three different accuracy levels $\ell = 0, 1, 2$, with the understanding that the cost of evaluating $\mathcal{F}_\ell$ increases with $\ell = 0, 1, 2$.
Given some μ^y-integrable quantity of interest QoI, we can approximate the expectation of QoI over μ^y by a telescoping sum, as usually done in MLMC,

$$\begin{aligned}\mathbb{E}_{\mu^y}[\mathsf{QoI}] \simeq \mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{QoI}_{\mathsf{L}}] &= \mathbb{E}_{\mu_0^y}[\mathsf{QoI}_0] + \sum_{\ell=1}^{\mathsf{L}} \left(\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell] - \mathbb{E}_{\mu_{\ell-1}^y}[\mathsf{QoI}_{\ell-1}]\right) \\ &= \sum_{\ell=0}^{\mathsf{L}} \Delta\mathsf{E}_\ell, \end{aligned} \tag{1.6}$$

with $\Delta\mathsf{E}_\ell := \mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell] - \mathbb{E}_{\mu_{\ell-1}^y}[\mathsf{QoI}_{\ell-1}]$, $\Delta\mathsf{E}_0 = \mathbb{E}_{\mu_0^y}[\mathsf{QoI}_0]$ and where, for $\ell = 0, 1, \ldots, \mathsf{L}$, QoI_ℓ is a μ_ℓ^y-integrable, level ℓ approximation of the quantity of interest QoI. This telescoping sum presents the basis for various types of multi-level techniques for BIPs. The work [71], for example, approximates the expectation (1.6) by splitting each $\Delta\mathsf{E}_\ell$ into three different terms, which are then computed using a mixture of importance-sampling and MCMC techniques. A multi-index generalization of such method is presented in [78]. In addition, similar multi-level ideas have also been attempted in the context of Multi-Level Sequential Monte Carlo (MLSMC) in the works [13, 79, 96].
In this work, we follow the approach proposed in [45], which is probably the first proposition of multi-level ideas for BIPs and consists of approximating $\mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{QoI}_{\mathsf{L}}]$ using the following ergodic estimator:

$$\mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{QoI}_{\mathsf{L}}] \approx \frac{1}{N_0}\sum_{n=1}^{N_\ell} \mathsf{QoI}_0(u_{0,0}^n) + \sum_{\ell=1}^{\mathsf{L}} \frac{1}{N_\ell}\sum_{n=1}^{N_\ell} \underbrace{\mathsf{QoI}_\ell(u_{\ell,\ell}^n) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1}^n)}_{:= Y_\ell^n},$$

where $\{u_{\cdot,\ell}^n\}_{n=0}^{N_\ell}$ is an ergodic Markov chain with invariant distribution μ_ℓ^y. The key idea is to couple the chains $\{u_{\ell,\ell-1}^n, u_{\ell,\ell}^n\}_{n=0}^{N_\ell}$ so that they are highly correlated and the variance of

[3] Typically, the observation operator only gives the pressure value at a few locations, however, we plot the whole field for illustration purposes.

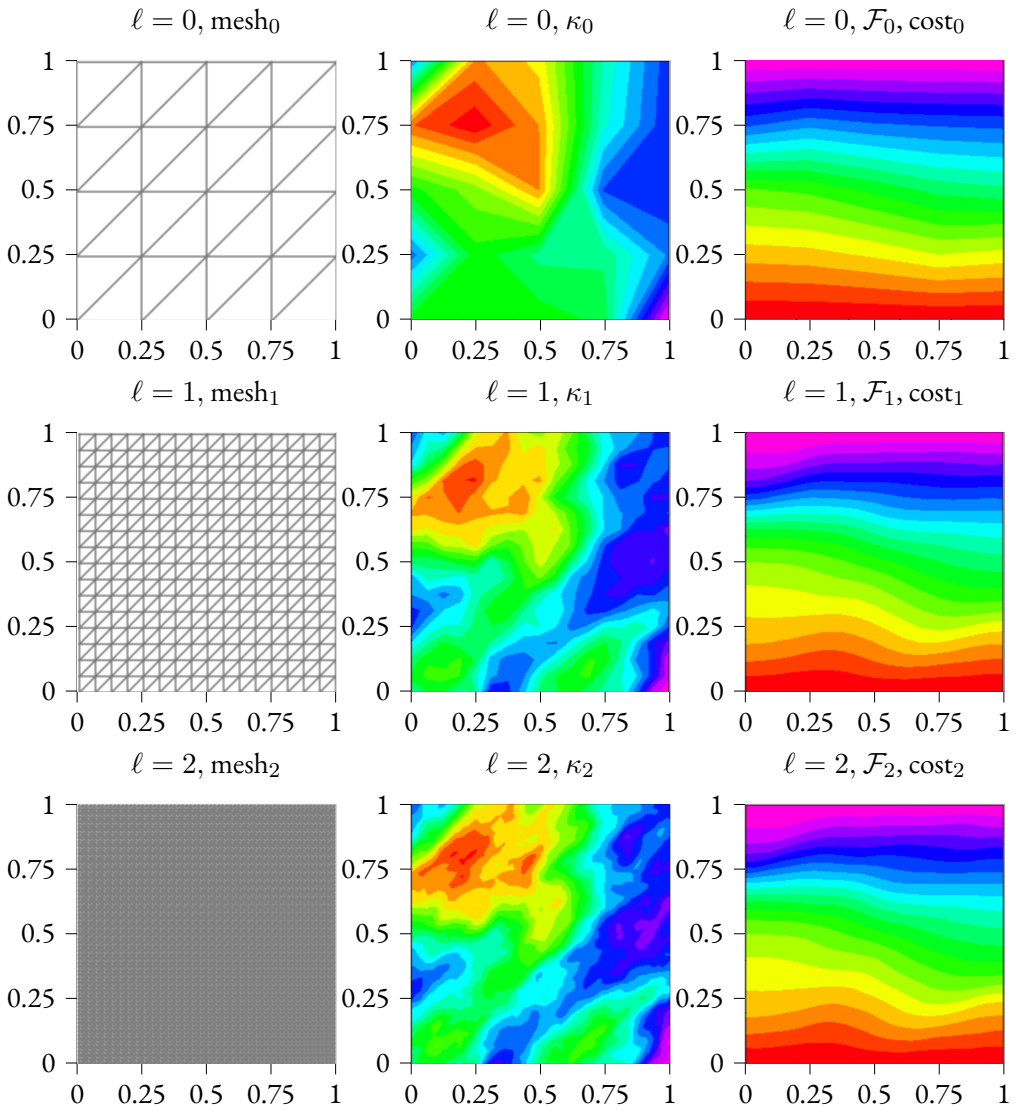

Figure 1.3: Depiction of accuracy and cost of $\mathcal{F}_\ell$ vs ℓ, where $\text{cost}_i < \text{cost}_{i+1}$ for three different levels.

the ergodic estimator $\mathbb{V}[N_\ell^{-1}\sum_n Y_\ell^n]$ becomes increasingly smaller as ℓ increases. By carefully choosing N_ℓ, this method can achieve a much better sampling complexity (in terms of cost versus tolerance) than its single-level counterparts (see [45]).

Few works have focused on constructing these types of couplings [35, 45]. In [45], the authors use (an approximation of) the posterior distribution at the previous discretization level $\ell - 1$ as a proposal for level ℓ. This is practically implemented by sub-sampling from the chain $\{u^n_{\ell-1,\ell-1}\}_{n=0}^{N_{\ell-1}}$. As it will be discussed later (c.f. Chapter 5), for such a method to converge (in the idealized case where one can sample from the posterior at the previous levels), however, it is required that the posterior at level $\ell - 1$ has not lighter tails than the posterior at level ℓ. This assumption can be relaxed by tempering the posteriors (as done in a single level in, e.g., [95]) at the previous discretization levels, however, it is not clear yet how to choose this tempering parameter. This sub-sampling method has been recently reviewed in [46]. Furthermore, from an implementation perspective, the work [152] presents a parallelization strategy for the ML-MCMC algorithm of [45], while the works [77] and [87] apply such an algorithm in the context of lattice field theory and statistical mechanics.

Such an idea has been recently expanded in [35], where the subsampling idea is combined with the so-called Dimension Independent Likelihood Informed (DILI) MCMC method of [36] to generate proposed samples at level 0 in their ML-MCMC algorithm, and, more recently, by the work [105], which proposes the use of the sub-sampling algorithm of [45] in the context of delayed-acceptance, with the aim of accelerating the mixing between chains generated by the ML-MCMC sampler. Some further work combining multi-level Monte Carlo ideas with Bayesian inference has been presented in [80], where the authors use rejection-free Markov transitions kernels, such as the Gibbs sampler, in order to couple the multi-level MCMC chains at two consecutive levels.

A different approach to coupling Markov chains, albeit in the context of unbiased estimation is given by *maximal coupling* techniques [53, 76, 82, 99]. Maximal coupling methods have been of interest, both from a theoretical and computational perspective, for a number of years. Traditionally, (maximal) coupling methods have been used as a tool in the convergence analysis of Markov chains [53, 139, 100, 159]. In this setting, one aims at estimating the so-called *mixing time* of a Markov chain by creating a coupling of two Markov chains X_n, Y_n, $n \in \mathbb{N}$, both having the same invariant measure, and estimating the first meeting time, i.e., $\tau = \min\{n \in \mathbb{N} : X_n = Y_n\}$ (c.f. Algorithm 1 in Chapter 3). Recently, these methods have gained a wider computational use; the works [70, 76] use coupling methods to construct unbiased Markov chain Monte Carlo estimators based on the seminal work of [62]. These methods have also been used to construct variance reduction techniques [3], such as *anthitetic variates* and *control variables*, for ergodic estimators obtained from Markov chains [131].

Multi-level methods: contributions

It is clear that ML-MCMC algorithms have started to become increasingly popular in the UQ community, as it can be evidenced by the impact of the work [45][4]. In this thesis we present several contributions to this emerging set of methodologies. In particular, in Chapters 5 and 6:

1. We propose two extensions of ideas presented in [45]. Our first extension can be seen as a generalization of their work to the case where a wider class of Independent Metropolis-Hastings (IMH) proposal distributions are considered (c.f Chapter 5). The second extension presents a ML-MCMC algorithm that admits state-dependent proposals, such as Random Walk Metropolis. These algorithm generates joint chains using a maximal coupling between proposal kernels and is presented in Chapter 6.

2. We present a thorough convergence analysis for the (coupled) ML-MCMC algorithm arising form these extensions and present conditions under which there exists a unique invariant probability measure induced by such algorithms, as well as quantifying their convergence rate to such a measure.

3. We present a non-asymptotic bound for the mean-square error for (non-necessarily) reversible Markov chains, such as the one induced by the ML-MCMC sampler. We remark that this contribution is interesting on its own and can be applied outside the scope of this work, however, such a bound is crucial to prove the complexity result of the ML-MCMC algorithm, as in [45]. We remark that this result is Presented in Chapter 3.

4. We extend the aforementioned complexity result of [45] to the case of ML-MCMC using IMH under some reasonable technical assumptions. Furthermore, we present an analogous result to the case with state-dependent proposals, albeit under more restrictive assumptions.

5. In the spirit of [132], we introduce a continuation-type ML-MCMC algorithm. Such a method obtains a robust estimation of the hyper-parameters in the ML-MCMC algorithm (e.g., number of samples needed for a given tolerance, c.f. Chapter 5) by estimating them on sequence of decreasing tolerances, ending when the required error tolerance is satisfied.

We implement these proposed methodologies on an array of BIPs and discuss their strengths and limitations. Lastly, we discuss several possible extensions to these ideas in Chapter 7.

1.4 Outline

The rest of this thesis is outlined as follows. Chapters 2 and 3 are devoted to a review of the theory and methodology of the methods of interest to this work, while chapters 4 through 6 present the main research body and contributions of this thesis. More precisely:

[4] Indeed, such a work has been a highlight of the *SIAM Journal of Uncertainty Quantification*, one of the authors has been awarded the SIAM UQ Early career Prize, and the paper has been republished in the SIGEST section of SIAM Review (vol. 61(3)) [46]

Chapter 2 is devoted to a thorough introduction to BIPs. We begin such a chapter presenting basics concepts of probability needed to construct the Bayesian solution to an inverse problem, and then present an overview of the theory and modeling choices of such an approach. We finalize this chapter discussing some non-MCMC based approaches to the solution of a BIP.
Chapter 3 is devoted to a review of theory and methodology of MCMC for Bayesian inverse problems. We begin this chapter by recalling some necessary concepts for Markov chains, such as Markov transition kernels, and give an overview of some common results regarding their convergence. We finalize this chapter with a survey of some common MCMC techniques.
Chapter 4 presents our first hierarchical method: *the generalized parallel tempering algorithm*. In this case the hierarchy is to be understood as a sequence of temperatures $\{T_k,\ k = 1, 2, \ldots\}$, which induce a posterior probability measure μ_k that gets increasingly "easier" to sample from as $T_k \to \infty$. Here, we introduce, analyze and implement two MCMC algorithms used to sample from this hierarchical model. This chapter is based on our published work [95].
Chapter 5 presents several contributions regarding the second hierarchical method (i.e., multilevel MCMC). In particular, such a Chapter introduces a ML-MCMC algorithm based on IMH proposals, together with a thorough analysis of the method.This chapter is based on the pre-print (currently under revision) [108]
Chapter 6 presents a new methodology for ML-MCMC methods based on the idea of maximal coupling [76]. This methodology allows for easily implemented ML-MCMC that can clearly overcome some of the difficulties associated to the methods of Chapter 5. We present a thorough theoretical analysis of our proposed method, and implement it for different BIPs. This chapter is based on ongoing work.
Lastly, **Chapter 7** summarizes and concludes this thesis and proposes several future research directions.

2 Bayesian inverse problems

In this chapter we present the conceptual and mathematical background of BIPs. We begin by recalling some basic concepts in probability, particularly on Gaussian measures, and then proceed to present BIPs in detail. We conclude this chapter by presenting a state of the art of some methods for solving BIPs. We remark that this is a review chapter written with the aim of making this thesis as self-contained as possible, and that no new material is presented here. Furthermore the content presented in this chapter is necessarily short, however we refer the interested reader to the monographs of, e.g., Dudley, or Ash [2, 48] for a detailed account on probability theory; to the books of, e.g., Bogachev or Da-Prato and Zabczyk [18, 37] for material regarding Gaussian measures on infinite-dimensional spaces; and to the seminal works of Dashti and Stuart [40, 156] for a detailed presentation of BIPs in infinite dimensions (or on Banach spaces), of which the material presented in this chapter is a (heavily) condensed version.

2.1 Preliminaries

2.1.1 Probability theory

The workhorse behind the Bayesian formulation of an inverse problem is, rather unsurprisingly, Bayes' theorem (c.f. Theorem 2.2.1), which lies at the heart of probability theory. We begin this chapter by recalling some necessary concepts and results from it that will be used throughout the rest of this thesis.

Definition 2.1.1 (Probability space): *A* probability space *(also known as* probability triple, or probability measure space*) is a triple* $(\Omega, \mathcal{X}, \mu)$, *where*

1. Ω *is the* sample space.
2. $\mathcal{X}$ *is a* σ-algebra *of subsets of* Ω.
3. $\mu : \mathcal{X} \to [0,1]$ *is a* probability measure*, i.e., a mapping satisfying the following two properties:*
   a) μ *is countably additive, i.e., given* $\{A_i\}_{i=1}^{\infty} \subseteq \mathcal{X}$, $A_i \cap A_j = \emptyset$, $\forall i \neq j$, *then* $\mu\left(\cup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mu(A_i)$.
   b) *The entire space has mass equal to one, i.e.,* $\mu(\Omega) = 1$.

We call the couple $(\Omega, \mathcal{X})$ *a* measurable space.

In particular, in this work we are interested in the case where $\Omega = (\mathsf{X}, \|\cdot\|_{\mathsf{X}})$ (resp. $\Omega = (\mathsf{X}, \langle\cdot,\cdot\rangle_{\mathsf{X}})$) is a separable Banach (resp. separable Hilbert) space, and where $\mathcal{X} = \mathcal{B}(\mathsf{X})$ is the Borel σ-algebra associated to X. Throughout this work, we will sometimes refer to X as the *state space* and to an element $A \in \mathcal{B}(\mathsf{X})$ as an *event*.
Given two probability measures μ, ν on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, we say that μ is *absolutely continuous* with respect to ν (denoted by $\mu \ll \nu$) if, for every measurable set A, $\nu(A) = 0$ implies $\mu(A) = 0$. We say that μ and ν are *equivalent in the sense of measures* ($\nu \simeq \mu$) if $\mu \ll \nu$ and $\nu \ll \mu$. Conversely, we say that μ and ν are *mutually singular* (denoted $\mu \perp \nu$) if there exist sets $A, B \in \mathcal{B}(\mathsf{X})$ such that $A \cap B = \emptyset$, $A \cup B = \mathsf{X}$ and $\mu(A) = \nu(B) = 0$.

Definition 2.1.2 (Radon-Nikodym derivative): *Let μ, ν be two probability measures on X with $\nu \ll \mu$. A $\mathcal{B}(\mathsf{X})$-measurable function $f : \mathsf{X} \to [0, \infty)$ is called the* Radon-Nikodym *derivative of ν with respect to μ if, for any measurable set $A \in \mathcal{B}(\mathsf{X})$, it holds that $\nu(A) = \int_A f(u)\mu(\mathrm{d}u)$. We will write $f(u) = \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(u)$.*

The Bayesian approach to inverse problems relies heavily upon the concept of *conditional probability*, defined next.

Definition 2.1.3 (Conditional probability): *Let $(\mathsf{X}, \mathcal{B}(\mathsf{X}), \mu)$ be a measure space, and let $A, B \in \mathcal{B}(\mathsf{X})$ be two events with $\mu(B) > 0$. The* conditional probability of A given B *is defined as*

$$\mu(A|B) := \frac{\mu(A \cap B)}{\mu(B)}. \tag{2.1}$$

Conversely, one then has that if $\mu(A) > 0$, then

$$\mu(B|A) := \frac{\mu(B \cap A)}{\mu(A)},$$

which when combined with (2.1), motivates the so-called *Bayes' formula*

$$\mu(A|B) = \frac{\mu(B|A)\mu(A)}{\mu(B)}. \tag{2.2}$$

Consider the case where $\mathsf{X} = \mathbb{R}^K$ for some $K \geq 1$, let ρ be a joint probability distribution on $(\mathsf{X} \times \mathsf{X}, \mathcal{B}(\mathsf{X} \times \mathsf{X}))$, with marginals μ, ν on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ having Lebesgue densities $\pi_u : \mathsf{X} \to \mathbb{R}_+, \pi_y : \mathsf{X} \to \mathbb{R}_+$, respectively. In this case, one can formulate Bayes' theorem in terms of such Lebesgue densities as

$$\pi(u|y) = \frac{\pi(y|u)\pi_u(u)}{\pi_y(u)}.$$

Although there exists an analogous form of (2.2) for the case where X is an infinite-dimensional space (and hence, no equivalent of the Lebesgue density exists), its formulation is less straightfor-

ward and we will delay its presentation until Section 2.2, Theorem 2.2.1. However,we introduce some concepts and technical results that will play a central role in its proof. We begin with the definition of transition probability (also called Markov transition kernel, transition probability kernel, or stochastic kernel, c.f. Definition 3.1.2).

Definition 2.1.4 (Transition probability kernel): *Let* $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ *and* $(\mathsf{Y}, \mathcal{B}(\mathsf{Y}))$ *be two measurable spaces. A* transition probability kernel *from* $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ *to* $(\mathsf{Y}, \mathcal{B}(\mathsf{Y}))$ *is a function* $p : \mathsf{X} \times \mathcal{B}(\mathsf{Y}) \to [0, 1]$ *satisfying:*

1. $\mathsf{X} \ni u \mapsto p(u, A)$ *is* $\mathcal{B}(\mathsf{X})$*-measurable for any* $A \in \mathcal{B}(\mathsf{Y})$.

2. $\mathcal{B}(\mathsf{Y}) \ni A \mapsto p(u, A)$ *is a probability measure on* $(\mathsf{Y}, \mathcal{B}(\mathsf{Y}))$ *for every* $u \in \mathsf{X}$.

Sometimes we will use the shorthand notation $p^u(\cdot) = p(u, \cdot)$. The Bayesian approach to inverse problems relies heavily upon the concept of *product regular conditional probability*, defined next.

Definition 2.1.5 (Product regular conditional probability): *Given two measurable spaces* $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ *and* $(\mathsf{Y}, \mathcal{B}(\mathsf{Y}))$*, set* $\mathsf{Z} := \mathsf{X} \times \mathsf{Y}, \mathcal{B}(\mathsf{Z}) := \mathcal{B}(\mathsf{X}) \otimes \mathcal{B}(\mathsf{Y})$*, and let* $(\mathsf{Z}, \mathcal{B}(\mathsf{Z}), \Pi)$ *be a (product) probability space. A* Product Regular Conditional Probability (P-RCP) *is a transition probability kernel* $p : \mathsf{Y} \times \mathcal{B}(\mathsf{X}) \to [0, 1]$ *satisfying* Π_{Y}*-a.e.,*

$$\Pi(A \times B) = \int_B p(y, A)\Pi_{\mathsf{Y}}(\mathrm{d}y) = \int_B p^y(A)\Pi_{\mathsf{Y}}(\mathrm{d}y), \quad \forall A \in \mathcal{B}(\mathsf{X}), B \in \mathcal{B}(\mathsf{Y}), \tag{2.3}$$

where Π_{Y} *is the* Y*-marginal of* Π*, i.e.,* $\Pi_{\mathsf{Y}}(\mathrm{d}y) = \int_{\mathsf{X}} \Pi(\mathrm{d}u, \mathrm{d}y)$*. In this setting we say that the regular conditional distribution of* u *given* y *(written* $u|y$*) exists and denote it by* p^y*.*

Notice that if Π is the product measure $\Pi = \mu \times \nu$, one can simply take $p^y = \mu$. It is known from [48, Theorem 10.1.1] that if a P-RCP exists, then, using the same notation as in the previous definition, it follows for any Π-integrable function g that

$$\mathbb{E}_\Pi[g] = \int_{\mathsf{Z}} g(u, y)\Pi(\mathrm{d}u, \mathrm{d}v) = \int_{\mathsf{Y}} \int_{\mathsf{X}} g(u, y)p^y(\mathrm{d}u)\Pi_{\mathsf{Y}}(\mathrm{d}y).$$

It is shown in [48, Theorem 10.2.2] that if Z is a *Polish space*[1], together with a Borel σ-algebra $\mathcal{B}(\mathsf{Z})$, then there exists a unique P-RCP $p^y()$ defined as in (2.3). We now present the following technical result from [150].

Theorem 2.1.1: *Let* $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ *and* $(\mathsf{Y}, \mathcal{B}(\mathsf{Y}))$ *be measurable spaces, and let* μ*,*ν *be probability measures on* $\mathsf{Z} = \mathsf{X} \times \mathsf{Y}$*, with* $\mathcal{B}(\mathsf{Z}) = \mathcal{B}(\mathsf{X}) \otimes \mathcal{B}(\mathsf{Y})$*. Assume that (i)* $\mu \ll \nu$ *with* $\frac{\mathrm{d}\mu}{\mathrm{d}\nu}(u, y) = f(u, y), \forall u \in \mathsf{X}, y \in \mathsf{Y}$ *and (ii) that the (product) regular conditional distribution of*

[1] i.e., a separable completely metrizable topological space

$u|y$ under ν, denoted by $\nu^y(\mathrm{d}u)$, exists. Then, the conditional distribution of $u|y$ over μ, $\mu^y(\mathrm{d}u)$ exists. Furthermore, $\mu^y \ll \nu^y$, with Radon-Nikodyn derivative given by

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\nu^y}(u) = \begin{cases} \frac{1}{Z(y)} f(u,y) & \text{if } 0 < Z(y) < \infty, \\ 1 & \text{otherwise}, \end{cases}$$

where $Z(y) := \int_{\mathsf{X}} f(u,y)\nu^y(\mathrm{d}u)$.

Proof. See [150, Theorem 1.3.1]. □

2.1.2 Gaussian measures

Gaussian measures are a class of commonly-used probability measures in the context of BIP. On the one hand, from a practical perspective, they are attractive for problems where either the mapping $u \mapsto \mathcal{F}(u)$ is (nearly) linear; indeed, if such a mapping is linear and the noise and prior measures are Gaussian, the resulting posterior measure will also be Gaussian. They are also often used as first-approximation to the posterior measure (c.f. [151, 23] and Section 2.3.2). On the other hand, from a theoretical point of view, they are widely used in the case where X is an infinite-dimensional normed space since, as opposed to the Lebesgue measure, they are well-defined in such spaces. Furthermore, as it will be further discussed in Section 2.2.1, a draw u from a Gaussian measure $\mathcal{N}(m, \mathcal{C})$ on a separable Hilbert space X, can be written as

$$u = m + \sum_{i=1}^{\infty} \sqrt{\lambda_i}\phi_i u_i,$$

where $\{\lambda_i\}_{i\in\mathbb{N}}$, $\{\phi_i\}_{i\in\mathbb{N}}$, are the (orthonormalized) eigenvalues and eigenfunctions of the covariance operator $\mathcal{C}$, and $u_i \overset{iid}{\sim} \mathcal{N}(0,1), \forall i = 1, 2, \ldots$. We now present a short survey of Gaussian measures on infinite-dimensional Banach spaces.

We begin by recalling some basic concepts of functional analysis. Given a Banach space X, we define its *dual space* as $\mathsf{X}^* := \{f : \mathsf{X} \to \mathbb{R} : \ f \text{ is a continuous, linear map}\}$. In the case where $(\mathsf{X}, \langle\cdot,\cdot\rangle_{\mathsf{X}})$ is a separable Hilbert space, we say that a linear operator $\mathcal{C} : \mathsf{X} \to \mathsf{X}$ is

1. *self-adjoint* (or *symmetric*) if for all $f, g \in \mathsf{X}$, $\langle \mathcal{C}f, g\rangle_{\mathsf{X}} = \langle f, \mathcal{C}g\rangle_{\mathsf{X}}$,

2. *positive-semidefinite* if $\forall f \in \mathsf{X}$, $\langle \mathcal{C}f, f\rangle_{\mathsf{X}} \geq 0$,

3. *trace-class* if given a complete orthonormal basis (CONB) $\{\phi_i\}_{i\in\mathbb{N}}$ of X, it follows that $\sum_{i\in\mathbb{N}} \langle \mathcal{C}\phi_i, \phi_i\rangle < +\infty$. Alternatively, if $\{\lambda_i\}_{i\in\mathbb{N}}$, $\{\phi_i\}_{i\in\mathbb{N}}$, are the (orthonormalized) eigenvalues and eigenvectors of $\mathcal{C}$ forming a CONB of X, $\mathcal{C}$ is a *trace-class operator* if $\sum_{i\in\mathbb{N}} \lambda_i < +\infty$.

Recall that a probability measure φ in $\mathbb{R}$ is a *1D-Gaussian measure* centered at $m \in \mathbb{R}$ with variance $\sigma^2 \geq 0$ if, $\forall A \in \mathcal{B}(\mathbb{R})$, it holds that

$$\varphi(A) = \int_A \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right) \mathrm{d}x =: \mathcal{N}(m, \sigma^2)(A).$$

In addition, we say that φ is a *Dirac distribution* (or *degenerate Gaussian*) if $\sigma = 0$, in which case we write

$$\varphi(A) = \delta_{x-m}(A) = \begin{cases} 0, & \text{if } x - m \notin A, \\ 1, & \text{if } x - m \in A. \end{cases}$$

It is well-known [18] that 1D-Gaussian measures are uniquely characterized by their mean m and variance σ^2. Now let $\mathsf{X} = \mathbb{R}^K$. For any $\lambda, u \in \mathsf{X}$, one can think of the map $\mathsf{X} \ni u \mapsto \langle \lambda, u \rangle_{\mathbb{R}^K} \in \mathbb{R}$ as a random variable on the measure space $(\mathsf{X}, \mathcal{B}(\mathsf{X}), \varphi)$; in this case, we say that φ is a *KD-Gaussian* measure if such a mapping induces a *1D-Gaussian* measure on $\mathbb{R}$ for each $\lambda \in \mathsf{X}$. This can be stated in more abstract terms in order to allow for the case where X is an infinite-dimensional separable Hilbert space.

Definition 2.1.6 ((abstract) Gaussian measure [37]): *Let $(\mathsf{X}, \langle \cdot, \cdot \rangle_{\mathsf{X}})$ be a (potentially infinite-dimensional) separable Hilbert space with associated Borel σ-algebra $\mathcal{B}(\mathsf{X})$. We say that a probability measure φ on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ is a* Gaussian measure *if, for each $f \in \mathsf{X}$, the map $\mathsf{X} \ni u \mapsto \langle f, u \rangle_{\mathsf{X}}$ induces a 1D Gaussian measure on $\mathbb{R}$ i.e., if there exists $m_f \in \mathbb{R}$ and $\sigma_f^2 \geq 0$ depending on f such that*

$$\varphi\left(\{u \in \mathsf{X} : \langle f, u \rangle \in A\}\right) = \mathcal{N}(m_f, \sigma_f^2)(A), \quad \forall A \in \mathcal{B}(\mathbb{R}).$$

Furthermore, we say that $\varphi = \mathcal{N}(m, \mathcal{C})$, with mean $m \in \mathsf{X}$ *and* covariance $\mathcal{C} : \mathsf{X} \to \mathsf{X}$, *a trace-class linear operator, if*

$$\begin{aligned} \int_{\mathsf{X}} \langle h, u \rangle_{\mathsf{X}} \varphi(\mathrm{d}u) &= \langle m, h \rangle_{\mathsf{X}}, \ \forall h \in \mathsf{X}, \\ \int_{\mathsf{X}} \langle h_1, u - m \rangle_{\mathsf{X}} \langle h_2, u - m \rangle_{\mathsf{X}} \varphi(\mathrm{d}u) &= \langle \mathcal{C} h_1, h_2 \rangle_{\mathsf{X}}, \ \forall h_1, h_2 \in \mathsf{X}. \end{aligned} \tag{2.4}$$

It is a clear consequence of Equation (2.4) that $\mathcal{C}$ is both symmetric and positive-(semi)definite. Similarly as for the 1D case, a Gaussian measure $\varphi = \mathcal{N}(m, \mathcal{C})$ is uniquely determined by its mean m and covariance operator $\mathcal{C}$ [37].

Theorem 2.1.2 (Fernique Theorem): *Let $(\mathsf{X}, \langle \cdot \rangle_{\mathsf{X}})$ be a separable Hilbert space, and let φ be a Gaussian measure. Then, there exists an $\alpha > 0$ such that*

$$\int_{\mathsf{X}} \exp(\alpha \|u\|_{\mathsf{X}}^2) \varphi(\mathrm{d}u) < +\infty.$$

In particular, this means that φ has moments of all orders; i.e., $\forall j \geq 0$ it holds that

$$\int_{\mathsf{X}} \|u\|_{\mathsf{X}}^j \, \varphi(\mathrm{d}u) < +\infty.$$

Definition 2.1.7 (Cameron-Martin space): *Let $(\mathsf{X}, \langle\cdot,\cdot\rangle_{\mathsf{X}})$ be a separable Hilbert space, and let $\varphi = \mathcal{N}(m, \mathcal{C})$, be a Gaussian measure on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, with $m \in \mathsf{X}$ and $\mathcal{C}$ a self-adjoint, positive semi-definite, trace-class covariance operator. The* Cameron-Martin space *of X associated to φ is defined as $Im(\mathcal{C}^{1/2})$ [37], and can be given a Hilbert structure with inner product $\langle \mathcal{C}^{-1/2}\cdot, \mathcal{C}^{-1/2}\cdot\rangle_{\mathsf{X}}$.*

The following result is a special case of the Feldman-Hajek theorem [18], and presents a rather important result in the theory of Gaussian measures in infinite-dimensional Hilbert-spaces: two Gaussian measures on an infinite-dimensional space are either equivalent or singular.

Theorem 2.1.3 (Cameron-Martin theorem): *Let $(\mathsf{X}, \langle\cdot,\cdot\rangle_{\mathsf{X}})$ be a separable Hilbert space, let $\mathcal{C}$ be a positive semidefinite, self-adjoint and trace-class covariance operator, and for $i = 1, 2$, with $m_i \in \mathsf{X}$, let $\varphi_{m_i} = \mathcal{N}(m_i, \mathcal{C})$, be Gaussian measures on X. Then, $\varphi_{m_1} \simeq \varphi_{m_2}$* if and only if *$m_1 - m_2 \in \mathrm{Im}(\mathcal{C}^{1/2})$, and*

$$\frac{\mathrm{d}\varphi_{m_2}}{\mathrm{d}\varphi_{m_1}}(u) = \exp\left(\langle m_2 - m_1, u - m_1\rangle_{\mathcal{C}} - \frac{1}{2}\|m_1 - m_2\|_{\mathcal{C}}^2 \right),$$

otherwise $\varphi_{m_1} \perp \varphi_{m_2}$. Here we have denoted $\langle a, b\rangle_{\mathcal{C}} = \langle \mathcal{C}^{-1/2}a, \mathcal{C}^{-1/2}b\rangle_{\mathsf{X}}$, $\forall a, b \in \mathsf{X}$.

This is in stark contrast to the finite-dimensional case, where absolute continuity between translated Gaussian probability measures holds for arbitrary translations. This is a fact of paramount importance when discussing BIP in infinite dimensions.

2.1.3 Spaces of probability measures

Let $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$ be a separable Banach space with associated Borel σ-algebra $\mathcal{B}(\mathsf{X})$. We will denote by $\overline{\mathcal{M}}(\mathsf{X})$ the set of real-valued signed measures on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, and by $\mathcal{M}(\mathsf{X}) \subset \overline{\mathcal{M}}(\mathsf{X})$ the set of probability measures on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$. Let $\mu \in \mathcal{M}(\mathsf{X})$ be a "reference" probability measure on X. In the context of Bayesian inverse problems, this reference probability measure should be understood as the posterior measure (i.e., $\mu = \mu^y$). Furthermore, we define the following spaces:

$$L_r = L_r(\mathsf{X}, \mu) = \left\{ f : \mathsf{X} \mapsto \mathbb{R},\ \mu\text{-integrable},\ \text{s.t}\ \|f\|_r^r := \int_{\mathsf{X}} |f(u)|^r \mu(\mathrm{d}u) < \infty \right\} \quad (2.5)$$

$$L_r^0 = L_r^0(\mathsf{X}, \mu) = \left\{ f \in L_r(\mathsf{X}, \mu),\ \text{s.t}\ \mu(f) := \int_{\mathsf{X}} f(u)\mu(\mathrm{d}u) = 0 \right\}.$$

Notice that, clearly, $L_r^0(\mathsf{X},\mu) \subset L_r(\mathsf{X},\mu)$. In the particular case where $r=2$, $L_2(\mathsf{X},\mu)$ (and hence L_2^0) is a Hilbert space with inner product given by

$$\langle f,g\rangle_{L_2} := \int_{\mathsf{X}} f(u)g(u)\mu(\mathrm{d}u), \quad f,g \in L_2(\mathsf{X},\mu).$$

Moreover, when $r=\infty$, we define

$$L_\infty(\mathsf{X},\mu) := \left\{ f : \mathsf{X} \mapsto \mathbb{R}, \mathcal{B}(\mathsf{X}) - \text{measurable s.t.} \inf_{\substack{\mu(B)=0 \\ B\in\mathcal{B}(\mathsf{X})}} \sup_{y\in\mathsf{X}\setminus B} |f(y)| < \infty \right\}.$$

In addition, for any $r \in [1,\infty]$, we define the spaces of (signed) measures

$$\mathcal{M}_r(\mathsf{X},\mu) := \{\nu \in \overline{\mathcal{M}(\mathsf{X})} \text{ s.t. } \nu \ll \mu,\ \|\nu\|_{L_r(\mathsf{X},\mu)} < \infty\},$$
$$\text{where} \quad \|\nu\|_{L_r(\mathsf{X},\mu)} := \left\| \frac{\mathrm{d}\nu}{\mathrm{d}\mu} \right\|_{L_r(\mathsf{X},\mu)},$$

together with

$$\mathcal{M}_r^0(\mathsf{X},\mu) := \{\nu \in \mathcal{M}_r(\mathsf{X},u),\ \text{s.t. } \nu(\mathsf{X}) = 0\}.$$

Once again, in the particular case where $r=2$, $\mathcal{M}_2(\mathsf{X},\mu)$ is a Hilbert space with inner product given by:

$$\langle \nu,\pi\rangle_{\mathcal{M}_2} := \int_{\mathsf{X}} \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(u)\frac{\mathrm{d}\pi}{\mathrm{d}\mu}(u)\mu(\mathrm{d}u), \quad \nu,\pi \in \mathcal{M}_2(\mathsf{X},\mu).$$

Notice that the definition of the L_r (respectively $\mathcal{M}_r$) norm depends on the reference measure μ on X. We remark that the function space $L_r(\mathsf{X},\mu)$ is isometrically isomorphic to the space of measures $\mathcal{M}_r(\mathsf{X},\mu)$, as stated in [143].
We now define some commonly-used (pseudo)*metrics* for a space of probability measures. We will use some of these metrics to study the convergence of the MCMC algorithms in Chapters 4, 5 and 6.

Definition 2.1.8 (Total variation distance): *Let $\mu,\nu \in \mathcal{M}(\mathsf{X})$ be absolutely continuous with respect to a common probability measure $\lambda \in \mathcal{M}(\mathsf{X})$. The* Total Variation (TV) distance *between μ and ν is given by*

$$d_{TV}(\mu,\nu) := \frac{1}{2}\int_{\mathsf{X}} \left| \frac{\mathrm{d}\mu}{\mathrm{d}\lambda}(u) - \frac{\mathrm{d}\nu}{\mathrm{d}\lambda}(u) \right| \lambda(\mathrm{d}u) = 1 - \int_{\mathsf{X}} \min\left\{ \frac{\mathrm{d}\mu}{\mathrm{d}\lambda}(u), \frac{\mathrm{d}\nu}{\mathrm{d}\lambda}(u) \right\} \lambda(\mathrm{d}u)$$
$$= \frac{1}{2}\|\mu-\nu\|_{\mathcal{M}_1(\mathsf{X},\lambda)},$$

where the second equality comes from the fact that $\min\{a,b\} = \frac{1}{2}(a+b-|a-b|)$

Notice that in the case where $\nu \ll \mu$, the TV distance between μ, ν is then given by

$$d_{\mathrm{TV}}(\mu,\nu) = \frac{1}{2}\int_{\mathsf{X}} \left|1 - \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(u)\right| \mu(\mathrm{d}u)$$

Definition 2.1.9 (Hellinger distance): *Let* $\mu, \nu \in \mathcal{M}(\mathsf{X})$ *be absolutely continuous with respect to a common probability measure* $\lambda \in \mathcal{M}(\mathsf{X})$. *The* Hellinger distance *between* μ *and* ν *is given by*

$$d_{Hell}(\mu,\nu) := \left(\frac{1}{2}\int_{\mathsf{X}} \left(\sqrt{\frac{\mathrm{d}\mu}{\mathrm{d}\lambda}(u)} - \sqrt{\frac{\mathrm{d}\nu}{\mathrm{d}\lambda}(u)}\right)^2 \lambda(\mathrm{d}u)\right)^{1/2}$$

Similarly as before, notice that in the case where $\nu \ll \mu$, the Hellinger distance is then given by

$$d_{\mathrm{Hell}}(\mu,\nu) := \left(\frac{1}{2}\int_{\mathsf{X}} \left(1 - \sqrt{\frac{\mathrm{d}\nu}{\mathrm{d}\mu}(u)}\right)^2 \mu(\mathrm{d}u)\right)^{1/2}$$

Definition 2.1.10 (Kullback-Liebler divergence): *Let* $\nu, \mu \in \mathcal{M}(\mathsf{X})$ *be two probability measures with* $\nu \ll \mu$. *The Kullback-Liebler (KL) divergence between* ν *and* μ *denoted by* $d_{KL}(\nu,\mu)$ *is given by*

$$d_{KL}(\nu,\mu) = \int_{\mathsf{X}} \log\left(\frac{\mathrm{d}\nu}{\mathrm{d}\mu}(u)\right) \nu(\mathrm{d}u)$$

Notice that $d_{KL}(\mu,\nu)$ *is not a proper metric since, in general,* $d_{KL}(\mu,\nu) \neq d_{KL}(\nu,\mu)$.

It is a consequence of Jensen's inequality [48] that $d_{\mathrm{KL}}(\mu,\nu) \geq 0$.

2.2 Bayesian inverse problems

We now present a rigorous derivation of Bayes' theorem in separable Banach spaces. Let $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$ and $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}})$ be two separable, potentially infinite-dimensional, Banach spaces, r equipped with Borel σ-algebra $\mathcal{B}(\mathsf{X})$ and $\mathcal{B}(\mathsf{Y})$, respectively, and let $\mathcal{F} : \mathsf{X} \to \mathsf{Y}$ be a measurable forward mapping operator. We are interested in obtaining the conditional probability distribution of $u \in \mathsf{X}$ given some noise-polluted measured data $y \in \mathsf{Y}$, where

$$y = \mathcal{F}(u) + \eta, \quad \eta \sim \mu_{\mathrm{noise}},$$

where $\eta \in \mathsf{Y}$ represents the random noise polluting the measured data and μ_{noise} is a probability measure on $(\mathsf{Y}, \mathcal{B}(\mathsf{Y}))$. Furthermore we assume that u follows a *prior distribution*, μ_{pr} on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, usually encoding the information available on u *before* any data has been observed. A

key component of this approach is that all its terms, namely y, η, u (and hence $\mathcal{F}(u)$), are random variables. Furthermore it is assumed that u and η are independent random variables.
Denote by μ^u_{noise} the translation of μ_{noise} by $\mathcal{F}(u)$. Our goal is to use the technical result presented in Theorem 2.1.1 to construct a (potentially infinite-dimensional) version of Bayes' theorem. We will require the following assumptions on μ_{pr}, μ_{noise} and μ^u_{noise} to hold.

Assumption 2.2.1 (Fundamental assumptions for Bayes' Theorem): *Given μ_{noise} and μ^u_{noise}, it holds that*

1. *for μ_{pr}-a.e. u, it holds that $\mu^u_{noise} \ll \mu_{noise}$. Furthermore, there exists a $\mathcal{B}(\mathsf{X}) \otimes \mathcal{B}(\mathsf{Y})$-measurable function $\Phi : \mathsf{X} \times \mathsf{Y} \to \mathbb{R}$ such that*

$$\frac{\mathrm{d}\mu^u_{noise}}{\mathrm{d}\mu_{noise}}(y) := \begin{cases} \frac{\exp(-\Phi(u;y))}{Z(y)} & \text{if } 0 < Z(y) < +\infty, \\ 1 & \text{otherwise} \end{cases}, \tag{2.6}$$

   with $Z(y) := \int_{\mathsf{X}} \exp(-\Phi(u;y))\mu_{\mathsf{pr}}(\mathrm{d}u)$.

2. $0 < Z(y) < +\infty$, $\quad \mu_{noise}$*-a.s.*

For any given $(u, y) \in \mathsf{X} \times \mathsf{Y}$, we will refer to $\Phi(u; y)$ as the potential *or* negative log-likelihood.

Remark 2.2.1: *Notice that Φ has the form $\Phi(u; y) = \tilde{\Phi}(\mathcal{F}(u); y)$, with $\tilde{\Phi} : \mathsf{Y} \times \mathsf{Y} \to \mathbb{R}$, $\mathcal{B}(\mathsf{Y} \times \mathsf{Y})$-measurable.*

Although Assumption 2.2.1.1 is relatively simple to satisfy in the finite-dimensional data case (i.e., when $\mathsf{Y} = \mathbb{R}^M$, with some $M \geq 1$), it is certainly not as straightforward to satisfy if Y is an infinite-dimensional Hilbert space. To visualize how this difficulty arises in the infinite-dimensional case, let $(\mathsf{Y}, \langle\cdot,\cdot\rangle)$ be an infinite-dimensional separable Hilbert space, and set $\mu_{\text{noise}} = \mathcal{N}(0, \Gamma)$, for $\Gamma : \mathsf{Y} \to \mathsf{Y}$ a self-adjoint, positive-definite, trace-class operator. Then, for (2.6) to hold true, it follows from Theorem 2.1.3 that $\mathcal{F}$ must satisfy $\mathcal{F}(u) \in \mathrm{Im}(\Gamma^{1/2})$, μ_{pr}-a.s. which is not necessarily the case. Conversely, if $\forall u \in \mathsf{X}$ it holds $\mathcal{F}(u) \in \mathrm{Im}(\Gamma^{1/2})$, it then follows from the Cameron-Martin theorem that $\Phi(u; y) = \frac{1}{2}\|\mathcal{F}(u)\|^2_\Gamma - \langle\mathcal{F}(u), y\rangle_\Gamma$, which is a measurable function in u and y.
Assumption 2.2.1.2 depends both on the functional form of Φ and the choice of prior, and can be satisfied under some relatively mild (and rather common [156]) assumptions on the structure of the BIP (c.f. Theorem 2.2.2).
We can now state the *general version* of Bayes' theorem. We remark that this is a well-known result (see, e.g., [40, 94, 153, 156]), however, we give its proof here for the sake of completeness.

Theorem 2.2.1 (Bayes' theorem): *Suppose Assumptions 2.2.1 hold. Then, the conditional distribution μ^y of $u|y$ exists and $\mu^y \ll \mu_{\mathsf{pr}}$, with*

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u) := \frac{1}{Z(y)} \exp\left(-\Phi(u; y)\right). \tag{2.7}$$

We will refer to μ^y as the posterior probability measure.

Proof. Define the joint probability measure $\Pi(\mathrm{d}u, \mathrm{d}y) := \mu_{\mathsf{pr}}(\mathrm{d}u)\mu_{\text{noise}}(\mathrm{d}y)$, corresponding to the process of sampling $u \sim \mu_{\mathsf{pr}}$ and $y \sim \mu_{\text{noise}}$ independently of each other. Similarly, define $\tilde{\Pi}(\mathrm{d}u, \mathrm{d}y) := \mu_{\mathsf{pr}}(\mathrm{d}u)\mu^u_{\text{noise}}(\mathrm{d}y)$, as the probability measure corresponding to the process associated to first sampling $u \sim \mu_{\mathsf{pr}}$, evaluating $\mathcal{F}(u)$, shifting μ_{noise} by $\mathcal{F}(u)$, and then sampling $y \sim \mu^u_{\text{noise}}$. It is clear from the Assumption 2.2.1 that $\tilde{\Pi} \ll \Pi$. Furthermore, since Π follows the process of sampling u and y independently of each other, then, clearly, $y|u$ exists under Π. The desired result then follows from an application of Theorem 2.1.1 with $\mu(\mathrm{d}u, \mathrm{d}y) = \tilde{\Pi}(\mathrm{d}u, \mathrm{d}y)$ and $\nu(\mathrm{d}u, \mathrm{d}y) = \Pi(\mathrm{d}u, \mathrm{d}y)$. □

Remark 2.2.2 (On the Bayesian formulation with finite-dimensional data): *Notice that in the case where Y is finite-dimensional and μ_{noise} has a Lebesgue density π_{noise}, the potential function would look like*

$$\begin{aligned}\Phi(u; y) &= -\log\left[\frac{\pi_{noise}(y - \mathcal{F}(u))}{\pi_{noise}(y)}\right] = -\log\left[\pi_{noise}(y - \mathcal{F}(u))\right] + \log\left[\pi_{noise}(y)\right] \\ &= -\log[\pi_{noise}(y - \mathcal{F}(u))] + c(y),\end{aligned}$$

where the constant $c(y)$ depends only on the data y and as such, can be absorbed as a redefinition of the normalization constant.

2.2.1 Prior modeling

One of the most important (and potentially challenging) aspects of a BIP is choosing an appropriate prior model. Choosing an appropriate prior typically requires some expert information on the model, and it is hence problem dependent. For finite-dimensional state spaces, such as $\mathbb{R}^K$, $K \geq 1$, one could, for example, assume that for each $i = 1, 2, \ldots, K$, u_i is independently distributed according to a prior measure $\mu_{\mathsf{pr},i}$, and then set $u \sim \mu_{\mathsf{pr}} := \otimes_{i=1}^K \mu_{\mathsf{pr},i}$. Alternatively, one could, of course, use priors that correlate some or all of the components of u, provided that such an information is available.

Prior modeling in function spaces is less straightforward, and a rather significant body of literature is devoted to the construction and analysis of special types of prior measures, such as Gaussian, Besov or uniform [32, 14, 156, 130] priors. These sort of priors arise in the case, for example, when the underlying mathematical model has a physical domain $D \subset \mathbb{R}^d$, $d = 1, 2, 3$, with $u \sim \mathsf{X}$ understood as a random field or function (rather than as a set of random parameters) on a space X of functions defined on D. In the model problems presented in Section 1.2, this corresponds, for example, to the spatially-varying permeability field $\kappa(x, \cdot), x \in D$ for model problem 1.2.1, or to the material density $\rho(x, \cdot), x \in D$ for model problem 1.2.2.

We now present the main idea behind the construction of infinite-dimensional prior measures. Let $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$ be a separable, infinite-dimensional Banach space of $\mathbb{R}$-valued functions defined on a domain D. In addition, consider a sequence of linearly independent functions $\{\phi_j\}_{j\in\mathbb{N}} \in \mathsf{X}$,

with $\|\phi_j\|_{\mathsf{X}} = 1, \forall j \in \mathbb{N}$, and let $\{u_j\}_{j\in\mathbb{N}}$ be a sequence of scalar, independently and identically distributed random variables $u_j \sim \nu_1$, with ν_j a probability measure on $(I, \mathcal{B}(I))$, $I \subset \mathbb{R}$. Furthermore, denote by $I_\infty = \times_{i\in\mathbb{N}} I$, together with the Borel σ-algebra $\mathcal{B}(I_\infty)$ generated by the cylindrical sets $A \subset I_\infty$, $A = \bigtimes_{i\in\mathbb{N}} A_i$, with finitely-many $A_i \neq I$ and define $\mu = \times_{j\in\mathbb{N}}\nu_j$, as a measure on $(I_\infty, \mathcal{B}(I_\infty))$. For some $p \in [1, \infty]$, let $\{w_i\}_{i\in\mathbb{N}} \in \ell^p$ be a deterministic sequence chosen such that, for some given $m_0 \in \mathsf{X}$, the series

$$u = m_0 + \sum_{j\in\mathbb{N}} \phi_j w_j u_j$$

converges in $L^p(\mathsf{X}, \mu)$ (which could happen, e.g., if $\nu_j \in \mathcal{N}$, $\forall j \in \mathbb{N}$). In this setting, one can then model $\mu_{\mathsf{pr}} := \text{Law}(m_0 + \sum_{j\in\mathbb{N}} \phi_j w_j u_j)$. This *spectral* representation of the prior lays the basis for constructing priors in infinite-dimensional space. Throughout this work we will limit ourselves to the use of Gaussian prior measures when modeling BIPs in infinite-dimensions, however, we refer the interested reader to, e.g., [22, 39, 73, 166] for the formulation, analysis, and solution of BIPs in function space using non-Gaussian priors.

We now proceed to describe such a methodology for Gaussian measures. Using a series expansion to model Gaussian priors in the context of BIP has been presented, e.g., in the seminal works of Stuart[156], Dashti and Stuart [40], and Cotter et. al., [31]. Let $(\mathsf{X}, \langle\cdot,\cdot\rangle_{\mathsf{X}})$ denote a separable Hilbert space, let $\mathcal{C} : \mathsf{X} \times \mathsf{X} \to \mathbb{R}$ be a self-adjoint and trace-class operator, and, without loss of generality, consider the prior $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$. We can generate samples from μ_{pr} be one of the following methods:

Karhunen-Loeve expansion

A first straightforward method to sample $u \sim \mu_{\mathsf{pr}}$ is to consider the Karhunen-Loeve expansion. Let $\{\lambda_i\}_{i\in\mathbb{N}}$, $\{\phi_i\}_{i\in\mathbb{N}}$ be the (orthonormalized) eigenvalues and eigenfunctions of $\mathcal{C}$. Since $\mathcal{C}$ is a trace-class operator, it then follows that $\sum_{i\in\mathbb{N}} \lambda_i < +\infty$, and as such, it can be seen that the series

$$u = \sum_{i=1}^{\infty} \sqrt{\lambda_i}\phi_i u_i, \quad u_i \overset{iid}{\sim} \mathcal{N}(0,1),\ i = 1, 2, \ldots, \tag{2.8}$$

converges in $L_2(\mathsf{X}, \mu)$, with $\mu = \times_{i\in\mathbb{N}}\mathcal{N}(0,1)$. Such a series is called the *Karhunen-Loeve* expansion of u. In practice, equation (2.8) needs to be truncated at a term K, leading to the finite dimensional approximation

$$u_K = \sum_{i=1}^{K} \sqrt{\lambda_i}\phi_i u_i, \quad u_i \overset{iid}{\sim} \mathcal{N}(0,1),\ i = 1, 2, \ldots, K. \tag{2.9}$$

One can then sample the (truncated) random variable $u_K \approx u$ from $\mu_{\mathsf{pr}}^K \approx \mu_{\mathsf{pr}}$ by sampling K independently and identically distributed random variables $u_i \sim \mathcal{N}(0,1)$, $i = 1, \ldots, K$,

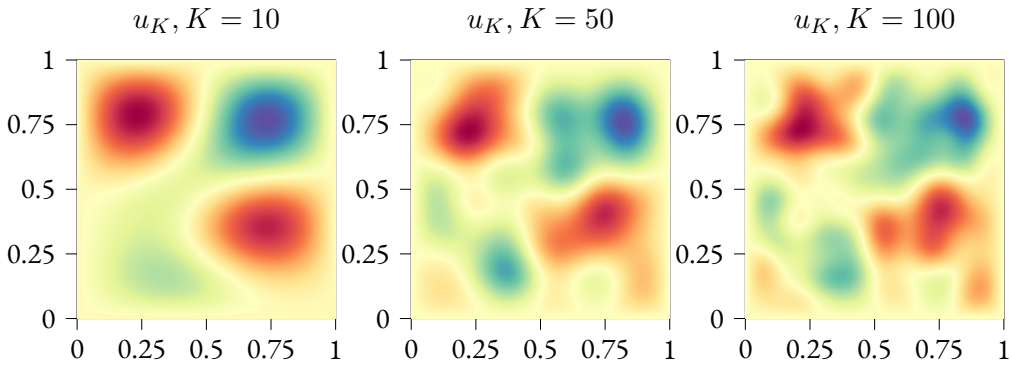

Figure 2.1: Random field generated using the KL expansion of a random field with different truncation levels $K = 10, 50, 100$.

and then summing the terms in equation (2.9). Of course, such a truncation induces a finite-dimensional vector u_K approximating u, which will in turn induce an approximate posterior measure $\mu^y_K \to \mu^y$ in a suitable sense as $K \to \infty$. We will discuss the convergence of $\mu^y_K \to \mu^y$ in Section 2.2.3. Such an approximation is depicted in Figure 2.1, where we present a realization of the permeability field $\kappa(x, u)$ in Example 1.2.1 truncated at three different values of K. As it can be seen, finer details on the field can be clearly appreciated as K increases. It is worth mentioning, however, that the "main features" of such a random field are captured by the first values u_i in the KL expansion. Informally speaking, larger values of K are able to *capture* higher levels of detail on the field. Throughout this thesis, we will employ this method of generating random fields u in Chapters 5 and 6.

PDE-based priors

An alternative approach to generating samples $u \sim \mathcal{N}(0, \mathcal{C})$ is by the characterization of the precision operator $\mathcal{A}^2 = \mathcal{C}^{-1}$ as a second-order "Laplace-like" differential operator on a bounded, open set $D \subset \mathbb{R}^d, d = 1, 2, 3$, with a domain chosen so that $\mathcal{A}$ is positive definite and invertible. This approach is particularly attractive when the random field u is defined on a physical domain D. Furthermore, one can control the regularity of the random field u by specifying the regularity of the operator $\mathcal{A}$. Lastly, there is a wide body of literature and computational resources for the efficient numerical solution of PDEs (see, e.g., [101]), making this approach also attractive from an implementation perspective. In order to model the precision operator $\mathcal{C}^{-1}$ as a "Laplace-like" differential operator, we need the following conditions to hold [156]:

Assumption 2.2.2: *The operator $\mathcal{A}$, densely defined on a Hilbert space $\mathsf{H} = L_2(D; \mathbb{R})$ satisfies the following properties:*

1. *$\mathcal{A}$ is positive definite, self-adjoint operator with compact inverse.*

2. *The eigenfunctions and eigenvalues,* $\{\phi_j\}_{j\in\mathbb{N}}$ *and* $\{\lambda_j\}_{j\in\mathbb{N}}$*, respectively, form an orthonormal basis for* H.

3. *There exist positive constants* c_m, C_M *such that for all* $j \in \mathbb{N}$ *it holds that*

$$c_m \leq \frac{\lambda_j}{j^{2/d}} \leq C_M.$$

4. *There exists* C' *such that*

$$\sup_{j\in\mathbb{N}} \left(\|\phi_j\|_{L_\infty} + \frac{1}{j} \|D\phi_j\|_{L_\infty} \right) \leq C'.$$

Before proceeding to describe how to generate samples $u \sim \mathcal{N}(0, \mathcal{A}^{-2})$ we first define *spatial white noise* (see, e.g., [33]).

Definition 2.2.1 (spatial white noise): *We say that the linear isometry* $\dot{W} : L_2(D;\mathbb{R}) \to L_2(\Omega;\mathbb{R})$*, with* $(\Omega, F, \mathbb{P})$ *a complete probability space, is a* white noise*, if given any* $\{\phi_j\}_{j\in\mathbb{N}} \in L_2(D;\mathbb{R})$*, then* $h_j := \langle \dot{W}, \phi_j \rangle$ *are Gaussian random variables with mean zero and covariance given by*

$$\mathbb{E}[h_i h_j] = \langle \phi_i, \phi_j \rangle_{L_2(D)}, \quad \forall i, j \in \mathbb{N},$$

where $\langle \dot{W}, \phi_j \rangle$ *denotes the action of* $\dot{W}$ *on* ϕ_j.

Given some white noise $\dot{W}$, one can generate samples $u \sim \mathcal{N}(0, \mathcal{A}^{-2})$ by solving $\mathcal{A}u = \dot{W}$, where the solution should be interpreted in an appropriate sense, as it will become clearer shortly after.
As an example, consider the following elliptic PDE:

$$\begin{cases} -\alpha\nabla\cdot(\Lambda\nabla u) + \alpha u &= \dot{W} \quad \text{in } D, \\ -\alpha\,(\Lambda\nabla u)\cdot \boldsymbol{n} &= 0 \quad \text{on } \partial D, \end{cases} \tag{2.10}$$

where $\boldsymbol{n}$ denotes the outward unit normal on ∂D, $\alpha > 0$ and $\Lambda \in \mathbb{R}^{d\times d}$ is a symmetric, uniformly bounded, positive-definite tensor denoting the anisotropy of the elliptic operator. In practice, the solution to equation (2.10) needs to be numerically approximated using, e.g., the finite element or the finite-difference method with accuracy parameter K (which denotes, e.g., the number of degrees of freedom in the approximation). Thus, just as with the Karhunen-Loeve approach, this method results in an approximate posterior $\mu^y_K \to \mu^y$ as $K \to \infty$, where again, the convergence is in a suitable sense. This is depicted in Figure 2.2, where a realization of the the permeability field $\kappa(x,\cdot)$ in Example 1.2.1 is shown at three levels of discretization. As it can be seen, finer details on the field can be clearly appreciated as the underlying (finite element) mesh becomes more refined. We will present this approximation in more detail in Section 2.2.3.

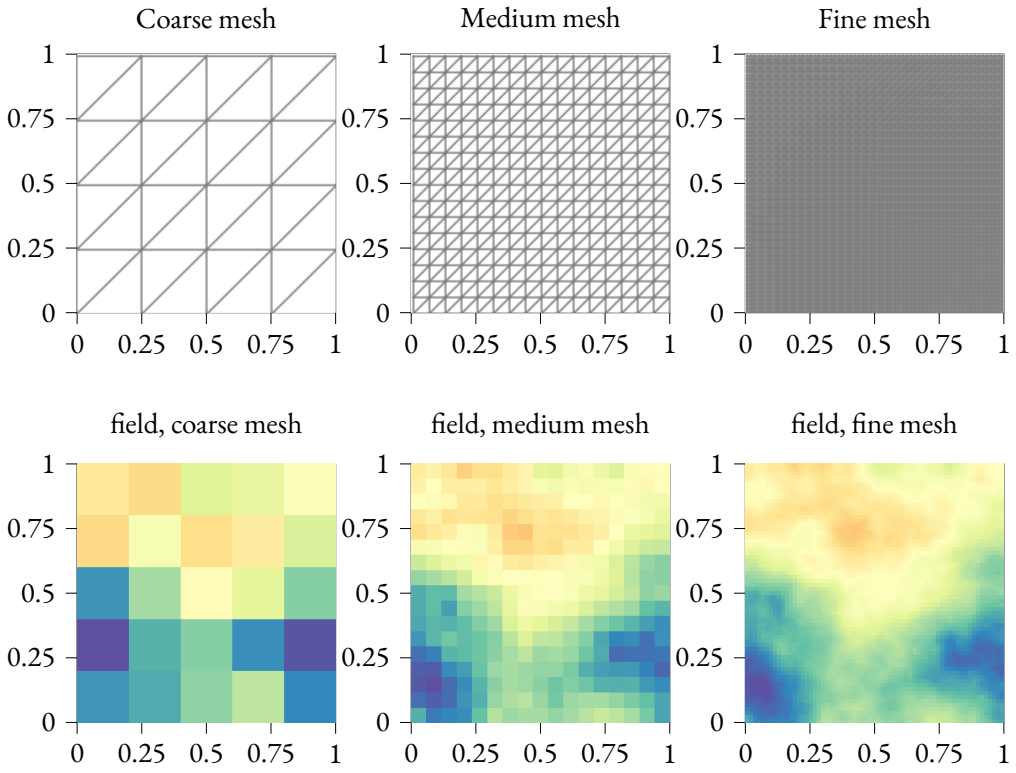

Figure 2.2: Top: Discretization meshes generated with a Laplace-like operator $\mathcal{A}^{-2}$. Bottom: Discretized random fields corresponding to each mesh

2.2.2 Well-posedness

A problem is said to be *well-posed* in the Hadamard sense, if (i) a solution exists and is unique and (ii) the behavior of such a solution changes continuously with the initial conditions, i.e., small changes on the input of the problem produce small changes on the output.
A problem that is not (Hadamard) well-posed, is said to be *ill-posed*. It is known that inverse problems (when seen, in a broad sense as "determining the input of a model given its solution") are often ill-posed. In the classical (i.e., frequentist's) approach to inverse problems, one typically aims at eliminating this ill-posedness by introducing a suitable *regularization term* in the minimization functional, however, we remark that these regularization techniques are outside the scope of this thesis, and invite the interested reader to the works, e.g., [85, 158] for its exposition in the context of inverse problems, and to the more recent book [55], for an introduction of this topic in the (closely-related) field of statistical learning.
Alternatively, the issue of well-posedness of an inverse problem can also be tackled from a Bayesian perspective. Indeed, broadly speaking (we will be more detailed shortly) a BIP is said to be well-posed if (i) *there exists a unique posterior probability measure*, and (ii) *small changes in the data produce small changes in the posterior*.
There are, arguably, two major notions of this well-posedness, namely the so-called *Lipschitz-Hellinger* well posedness, presented by Stuart and Dashti [40, 156], and the more general concept of (M, d)*-well posedness* of Latz [94], where M is to be understood as space of probability measures and d as a *(pseudo-)distance* between such measures. We will present the *Lipschitz-Hellinger* well posedness of Stuart, and refer the interesting reader to the works [94, 154] for a further study on Bayesian well-posedness.
Let $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}})$ be a separable, possibly infinite-dimensional Banach space, with associated Borel σ-algebra $\mathcal{B}(\mathsf{Y})$. A BIP in the form of (2.7) is *Lipschitz-Hellinger well-posed* [94, 156] if

i (Existence and uniqueness) There exists a unique posterior probability measure μ^y that is absolutely continuous with respect to μ_{pr}.

ii (stability) There exists a positive constant $C = C(r)$ such that for all $y, y' \in \mathsf{Y}$ with $r > \max\{\|y\|_{\mathsf{Y}}, \|y'\|_{\mathsf{Y}}\}$, it holds that $d_{\text{Hell}}(\mu^y, \mu^{y'}) < C(r)\,\|y - y'\|_{\mathsf{Y}}$, i.e., the mapping $y \mapsto \mu^y$ is locally Lipschitz continuous in the Hellinger metric.

It is shown in [40] that a BIP is well-posed under the following assumptions on the potential function (negative log-likelihood), $\Phi(u; y) : \mathsf{X} \times \mathsf{Y} \to \mathbb{R}$, and the prior μ_{pr}.

Assumption 2.2.3: *Let* $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$, $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}})$ *be two separable Banach spaces with* $u \in \mathsf{X}$ *and* $y \in \mathsf{Y}$. *We assume that* $\Phi : \mathsf{X} \times \mathsf{Y} \mapsto \mathbb{R}$ *has the following properties:*

1. $\forall \epsilon, r > 0, \exists M(\epsilon, r) > 0$ *such that* $\forall u \in \mathsf{X}, y \in \mathsf{Y}, \|y\|_{\mathsf{Y}} < r$,

$$\Phi(u; y) > M - \epsilon \|u\|_{\mathsf{X}}^2,$$

that is, there is a lower (quadratic) bound on the potential function.

2. $\forall r > 0, \exists K(r) > 0$ *measurable such that* $\forall u \in \mathsf{X},\ y \in \mathsf{Y}$, *with* $\max\{\|u\|_{\mathsf{X}}, \|y\|_{\mathsf{Y}}\} < r$,

$$\Phi(u; y) \leq K(r),$$

i.e, there is an upper bound on the potential.

3. $\forall r > 0, \exists L(r) > 0$ *such that* $\forall u, u' \in \mathsf{X}, y \in \mathsf{Y}$, *with* $\max\{\|u\|_{\mathsf{X}}, \|u'\|_{\mathsf{X}}, \|y\|_{\mathsf{Y}}\} < r$,

$$|\Phi(u; y) - \Phi(u'; y)| \leq L(r)\|u - u'\|_{\mathsf{X}},$$

which means that we have Lipschitz continuity of $\Phi(\cdot; y)$ *with respect to the first argument.*

4. $\forall \epsilon, r > 0,\ \exists C(\epsilon, r) > 0 \in \mathbb{R}$, *such that* $\forall\, y, y' \in \mathsf{Y}, u \in \mathsf{X}$, *with* $\max\{\|y\|_{\mathsf{Y}}, \|y'\|_{\mathsf{Y}}\} < r$,

$$|\Phi(u; y) - \Phi(u; y')| \leq \exp(\epsilon\|u\|_{\mathsf{X}}^2 + C(\epsilon, r))\|y - y'\|_{\mathsf{Y}}$$

which means that we have Lipschitz continuity of $\Phi(u; \cdot)$ *with respect to* y *for any* $u \in \mathsf{X}$, *with* u*-dependent Lipschitz constant.*

5. *Given a sufficiently small* $\epsilon > 0$,

$$\int_{\mathsf{X}} \exp\left(\epsilon \|u\|_{\mathsf{X}}^2\right) \mu_{\mathsf{pr}}(\mathrm{d}u) < +\infty. \tag{2.11}$$

6. *Any (small) ball has positive* μ_{pr}*-mass, i.e.,*

$$\int_{\|u\|_{\mathsf{X}} < r} \mu_{\mathsf{pr}}(\mathrm{d}u) > 0, \quad \forall r > 0.$$

Notice that in the case of finite-dimensional, additive Gaussian noise, $\mu_{\text{noise}} = \mathcal{N}(0, \Sigma)$, $\Sigma \in \mathbb{R}^{M \times M}$, we set $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}}) = (\mathbb{R}^M, \|\cdot\|_\Sigma)$ and we have that $\Phi(u; y) = \frac{1}{2}\|y - \mathcal{F}(u)\|_\Sigma^2$, where $\|a\|_\Sigma = \left\|\Sigma^{-1/2}a\right\|$, $\forall a \in \mathsf{Y}(= \mathbb{R}^M)$. In this case, Assumptions 2.2.3.1-4 can be simplified by the following proposition in [156], which relates to the properties of $\mathcal{F}$.

Lemma 2.2.1 (Lemma 2.8 in [156]): *Let* $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}}) = (\mathbb{R}^M, \|\cdot\|_\Sigma)$ *and suppose* $\mathcal{F} : \mathsf{X} \to \mathsf{Y}$ *satisfies the following:*

i) *Given* $\epsilon > 0,\ \exists\, M(\epsilon) \in \mathbb{R}$ *such that* $\forall u \in \mathsf{X}$,

$$\|\mathcal{F}(u)\|_\Sigma \leq \exp(\epsilon\|u\|_{\mathsf{X}}^2 + M).$$

ii) $\forall r > 0, \exists K(r) > 0$, *such that* $\forall u, u' \in \mathsf{X}$, *with* $\max\{\|u\|_{\mathsf{X}}, \|u'\|_{\mathsf{X}}, \} < r$,

$$\left\|\mathcal{F}(u) - \mathcal{F}(u')\right\|_\Sigma \leq K(r)\|u - u'\|_{\mathsf{X}}.$$

Then, Assumptions 2.2.3.1-4 hold for $\Phi(u;y) = \frac{1}{2}\|y - \mathcal{F}(u)\|_{\Sigma}^2$.

We can state the following theorem of [156] which gives conditions for which a BIP of the form 1.2 is *Lispchitz-Hellinger well-posed.*

Theorem 2.2.2 (Theorems 4.1 and 4.2 in [156]): *Suppose Assumption 2.2.3 holds. Then, the BIP associated to* (1.2) *is* Lipschitz-Hellinger well-posed.

Existence and uniqueness of μ^y are shown in [156, Theorem 4.1], while the stability of the posterior measure with respect to the data y is given by [156, Theorem 4.2]

Remark 2.2.3: *Assumptions 2.2.3 presented in [40] are sufficient, but not necessary, to prove the well-posedness of the BIP.*

Remark 2.2.4 (On the use of Gaussian priors): *In the particular case where* $\mu_{\mathsf{pr}} = \mathcal{N}(m, \mathcal{C})$, *with* $m \in \mathsf{X}$ *and* $\mathcal{C}$ *a trace-class, self-adjoint, and positive covariance operator, condition* (2.11) *is a consequence of Fernique's theorem (c.f. Theorem 2.1.2), while* (2.11) *holds since all balls on a separable Banach space have positive mass under a Gaussian measure [18].*

Throughout this work we will take for granted that Assumption 2.2.3 is satisfied, thus resulting on BIPs that are Lipschitz-Hellinger well-posed. The works [94, 154] present studies on well posedness under weaker assumptions.

2.2.3 Approximation and convergence

As mentioned in previous sections, in practice, it is often the case that one needs to sample from an approximate posterior μ_{L}^y instead of the "true" posterior μ^y. This approximate posterior μ_{L}^y is induced when

1. u_{L} is a finite-dimensional approximation of an object in function space, which can happen, e.g., when truncating the KL expansion of u with truncation parameter L, or in the case where u follows a PDE-based prior discretized at level L, as in Section 2.2.1.

2. The forward mapping operator $\mathcal{F}$ needs to be numerically approximated by $\mathcal{F}_{\mathsf{L}}$, which results in an approximate potential $\Phi_{\mathsf{L}}(u;y) := \|y - \mathcal{F}_{\mathsf{L}}(u)\|_{\mathsf{Y}}^2$. This is the case, e.g., when the underlying mathematical model is a PDE that, for implementation purposes, is numerically approximated (for example, via finite elements or finite differences) with accuracy level L.

In practice, these are non-mutually exclusive approximations; in fact, they tend to go hand-in-hand with one another, however, they each provide a different source of error; i.e., there is an error associated to the finite-dimensional discretization of the parameter space, and there is an additional error associated with the numerical approximaiton of the forward operator . Throughout this work, we will denote by $\Phi_{\mathsf{L}}(u;y)$ the approximation of $\Phi(u;y)$ at accuracy level L which takes into account both sources of error (i.e., finite-basis representation of u and numerical approximation of $\mathcal{F}$).

Finite-dimensional discretization for PDE-based priors

We now present a finite-dimensional approximation of an infinite-dimensional BIP, in the setting of PDE-based priors (c.f. Section 2.2.1). We follow a procedure similar to that of [23]. Let $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}})$ be a separable Banach space, let $\mathsf{X} = L_2(D;\mathbb{R})$ and let $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{A}^{-2})$ denote the prior measure on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ with $\mathcal{A} : \mathrm{Dom}(\mathcal{A}) \to L_2(D;\mathbb{R})$ a "Laplace-like" differential operator satisfying Assumption 2.2.2, with eigenvalues and eigenfunctions $\{\lambda_j\}_{j\in\mathbb{N}}$, $\{\phi_j\}_{j\in\mathbb{N}}$, respectively. For any $s \in \mathbb{R}$, define the space

$$V^s := \left\{ v \in L_2(D;\mathbb{R}) : \sum_{j\in\mathbb{N}} \langle v, \phi_j\rangle_{L_2}^2 \lambda_j^s < +\infty \right\},$$

with norm $\|v\|_{V^s}^2 := \sum_{j\in\mathbb{N}} \langle v, \phi_j\rangle^2 \lambda_j^s$. Setting $s = 1$, $\mathcal{A}$ can be extended continuously as $\mathcal{A} : V \to V^*$, denote now by X_{L} a finite-dimensional subspace of V of dimension K_{L} and let $\{\phi_j\}_{j=1}^{K_{\mathsf{L}}}$ be a basis for X_{L}. In X_{L} we can write the finite-dimensional problem *find* $\tilde{u}_{\mathsf{L}} \in \mathsf{X}_{\mathsf{L}}$ *satisfying*

$$\langle \mathcal{A}\tilde{u}_{\mathsf{L}}, v_{\mathsf{L}}\rangle_{V,V^*} = \langle \dot{W}, v_{\mathsf{L}}\rangle, \quad \forall v_{\mathsf{L}} \in \mathsf{X}_{\mathsf{L}}. \tag{2.12}$$

Expanding $\tilde{u}_{\mathsf{L}}$ on the basis $\tilde{u}_{\mathsf{L}} = \sum_{j=1}^{K_{\mathsf{L}}} (u_{\mathsf{L}})_j \phi_j$, $\tilde{u}_{\mathsf{L}}$ can be identified by the vector $u_{\mathsf{L}} \in \mathbb{R}$ with entries $(u_{\mathsf{L}})_j \in \mathbb{R}$.
Since $h_j = \langle \dot{W}, \phi_j\rangle$ is a mean-zero Gaussian random variable with covariance given by $\mathbb{E}[h_i h_j] = \langle \phi_i, \phi_j\rangle =: M_{\mathsf{L}_{i,j}}$, it follows that the term $b_{\mathsf{L}} := (\langle \dot{W}, \phi_1\rangle, \ldots, \langle \dot{W}, \phi_{K_{\mathsf{L}}}\rangle)$ is a Gaussian vector in $\mathbb{R}^{K_{\mathsf{L}}}$ with covariance matrix $M_{\mathsf{L}} \in \mathbb{R}^{K_{\mathsf{L}}\times K_{\mathsf{L}}}$, with entries $M_{\mathsf{L}_{i,j}}$. Furthermore, letting $A_{\mathsf{L}} \in \mathbb{R}^{K_{\mathsf{L}}\times K_{\mathsf{L}}}$ be the matrix with entries $A_{\mathsf{L}_{i,j}} = \langle \mathcal{A}\phi_i, \phi_j\rangle_{V,V^*}$, $i, j \in \{1, 2, \ldots, K_{\mathsf{L}}\}$, one can write a finite-dimensional version of (2.12) as

$$A_{\mathsf{L}} u_{\mathsf{L}} = b_{\mathsf{L}}, \quad b_{\mathsf{L}} \sim \mathcal{N}(0, M_{\mathsf{L}}).$$

From a practical perspective, one can then generate u_{L} by

$$u_{\mathsf{L}} = A_{\mathsf{L}}^{-1} M^{1/2} \xi, \quad \xi \sim \mathcal{N}(0, I_{K_{\mathsf{L}}\times K_{\mathsf{L}}}),$$

which in turn implies that the probability measure of $u_{\mathsf{L}} \in \mathbb{R}^{K_{\mathsf{L}}}$ is $\mu_{\mathsf{L}} \sim \mathcal{N}(0, A_{\mathsf{L}}^{-1} M_{\mathsf{L}} A_{\mathsf{L}}^{-1})$. Typically, A_{L} and M_{L} are the *stiffness* and *mass* matrices in the FE literature. The Gaussian measure of u_{L} then induces a Gaussian measure of $\tilde{u}_{\mathsf{L}}$ in X, which is actually concentrated on X_{L}. We can equivalently characterize $\tilde{u}_{\mathsf{L}} \in \mathsf{X}_{\mathsf{L}}$ as a suitable projection on u by the following procedure: given $u \sim \mathcal{N}(0, \mathcal{A}^{-2})$, define u_{L} such that $\langle \mathcal{A}u_{\mathsf{L}}, v_{\mathsf{L}}\rangle_{V,V^*} = \langle \mathcal{A}u, v_{\mathsf{L}}\rangle_{V,V*}$, $\forall v_{\mathsf{L}} \in \mathsf{X}_{\mathsf{L}}$. This procedure induces a "projection" operator $\mathcal{P}_{\mathsf{L}}^{\mathcal{A}} : \mathsf{X} \to \mathsf{X}_{\mathsf{L}}$ such that $u_{\mathsf{L}} = \mathcal{P}_{\mathsf{L}}^{\mathcal{A}} u$. Since V is

a Polish space, it follows from [48, Theorem 10.2.2] that one can write μ_{pr} in terms of a RCP $\widehat{\mu_{\mathsf{pr}}}(u_{\mathsf{L}}, \mathrm{d}z)$ with $z := u - u_{\mathsf{L}}$ and a marginal prior $\mu_{\mathsf{pr}_{\mathsf{L}}}(\mathrm{d}u_{\mathsf{L}})$ on $(\mathsf{X}_{\mathsf{L}}, \mathcal{B}(\mathsf{X}_{\mathsf{L}}))$ of the form:

$$\mu_{\mathsf{pr}}(\mathrm{d}u) = \widehat{\mu_{\mathsf{pr}}}(u_{\mathsf{L}}, \mathrm{d}z)\mu_{\mathsf{pr}_{\mathsf{L}}}(\mathrm{d}u_{\mathsf{L}}).$$

Lastly, consider the discretized forward operator $\mathcal{F}_{\mathsf{L}} : \mathsf{X}_{\mathsf{L}} \to \mathsf{Y}$ understood as the map taking into account the state space discretization, together with the numerical approximation of the mathematical model driving $\mathcal{F}$, and suppose that Assumption 2.2.1 holds with $\Phi : \mathsf{X} \times \mathsf{Y} \to \mathbb{R}$ replaced by $\Phi_{\mathsf{L}} : \mathsf{X} \times \mathsf{Y} \to \mathbb{R}$, where $\Phi_{\mathsf{L}}(u; y) = \tilde{\Phi}(\mathcal{F}_{\mathsf{L}}(\mathcal{P}_{\mathsf{L}}^{\mathcal{A}} u); y)$, with $\tilde{\Phi}$ as in Remark 2.2.1. It then follows from Theorem 2.2.1 that there exists a "discretized" posterior measure μ_{L}^{y} on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, such that $\mu_{\mathsf{L}}^{y} \ll \mu_{\mathsf{pr}}$, with

$$\frac{\mathrm{d}\mu_{\mathsf{L}}^{y}}{\mathrm{d}\mu_{\mathsf{pr}}}(u) := \frac{1}{Z_{\mathsf{L}}} \exp\left(-\Phi_{\mathsf{L}}(u; y)\right), \quad \text{with } Z_{\mathsf{L}} := \int_{\mathsf{X}} \exp\left(-\Phi_{\mathsf{L}}(u; y)\right) \mu_{\mathsf{pr}}(\mathrm{d}u).$$

Notice that by proceeding in this way, the approximate posterior μ_{L}^{y} is still defined in the infinite-dimensional space X. On the other hand,

$$\begin{aligned}\mu_{\mathsf{L}}^{y}(\mathrm{d}u) = \mu_{\mathsf{L}}^{y}(\mathrm{d}u_{\mathsf{L}}, \mathrm{d}z) &= \frac{1}{Z_{\mathsf{L}}} \exp\left(-\Phi_{\mathsf{L}}(u; y)\right) \mu_{\mathsf{pr}}(\mathrm{d}u) \\ &= \frac{1}{Z_{\mathsf{L}}} \exp\left(\tilde{\Phi}(u_{\mathsf{L}}; y)\right) \widehat{\mu_{\mathsf{pr}}}(u_{\mathsf{L}}, \mathrm{d}z)\mu_{\mathsf{pr}_{\mathsf{L}}}(\mathrm{d}u_{\mathsf{L}}) = \widehat{\mu_{\mathsf{L}}^{y}}(\mathrm{d}u_{\mathsf{L}})\widehat{\mu_{\mathsf{pr}}}(u_{\mathsf{L}}, \mathrm{d}z),\end{aligned}$$

with

$$\frac{\mathrm{d}\widehat{\mu_{\mathsf{L}}^{y}}}{\mathrm{d}\mu_{\mathsf{pr}_{\mathsf{L}}}}(u_{\mathsf{L}}) = \frac{1}{Z_{\mathsf{L}}} \exp\left(-\tilde{\Phi}(\mathcal{F}_{\mathsf{L}}(u_{\mathsf{L}}); y)\right),$$

i.e., the posterior μ_{L}^{y} can be factorized as a posterior $\widehat{\mu_{\mathsf{L}}^{y}}$ on X_{L} and the prior RCP of $u|u_{\mathsf{L}}$. In other words, the BIP only updated the distribution of u_{L} in X_{L} and leaves unchanged the conditional distribution of u given u_{L}, as this part is "not seen" by the approximate forward model $\mathcal{F}_{\mathsf{L}}$.
Stuart [156] presents the following result pertaining the convergence of the discretized posterior to μ^{y}.

Theorem 2.2.3 (Convergence of discretized posterior): *Suppose that both $\Phi(u; y)$ and $\Phi_{\mathsf{L}}(u; y)$ satisfy Assumptions 2.2.3.1 and 2.2.3.2 with constants independent of L. Suppose, furthermore that for any $\epsilon > 0$, there exists a finite $K' = K'(\epsilon) > 0$ such that*

$$\begin{aligned}|\Phi(u; y) - \Phi_{\mathsf{L}}(u; y)| \leq K' \exp(\epsilon \|u\|_{\mathsf{X}}^2)\Xi(\mathsf{L}), \quad \textit{and} \\ \int_{\mathsf{X}} \exp\left(2\epsilon \|u\|_{\mathsf{X}}^2\right) \mu_{\mathsf{pr}}(\mathrm{d}u) < +\infty,\end{aligned} \tag{2.13}$$

with $\Xi(\mathsf{L}) \to 0$ *as* $\mathsf{L} \to \infty$. *Then, there exists a positive constant* C_H *independent of* ℓ *such that*

$$d_{Hell}(\mu^y, \mu^y_{\mathsf{L}}) \leq C_H \Xi(\mathsf{L}).$$

Remark 2.2.5 (On the posterior convergence when using Gaussian priors): *In the case where* $\mu_{\mathsf{pr}} = \mathcal{N}(m, \mathcal{C})$, *with* $m \in \mathsf{X}$ *and* $\mathcal{C}$ *is a self-adjoint, positive, trace-class covariance operator, condition* (2.13) *is satisfied as a consequence of Fernique's theorem (c.f. Theorem 2.1.2) for sufficiently small* ϵ. *The previous theorem is a trivial adaptation of [156, Theorem 4.6], which was originally stated for Gaussian priors.*

2.3 Solving BIPs

So far this chapter has focused on the formulation of the Bayesian approach to inverse problems. As we have seen, under some technical conditions on the prior, the noise and the underlying mathematical model generating the data, there exists a well-defined posterior probability measure for the set of (potentially infinite-dimensional) random parameters conditioned on the observed (also, potentially infinite-dimensional), noise-polluted data. However, we have not discussed yet a notion of *solution* to such a problem. In a broad sense, we will understand the solution to a BIP (whether it is an exact solution or an approximation of it, as we will discuss shortly) as the process of extracting information about $u|y \sim \mu^y$.

In many applications arising in science and engineering, one aims at obtaining statistical quantities of a given μ^y-integrable quantity of interest $\mathsf{QoI} : \mathsf{X} \to \mathbb{R}$, such as its expected value over the posterior measure, $\mathbb{E}_{\mu^y}[\mathsf{QoI}]$, or the probability, under the same measure, of QoI exceeding a given threshold value A, $\mathbb{P}_{\mu^y}(\mathsf{QoI} > A) = \mathbb{E}_{\mu^y}[\mathbb{1}_{\{\mathsf{QoI}>A\}}]$. For most problems of interest, however the computation of $Z(y) = \int_{\mathsf{X}} \exp(-\Phi(u; y))\mu_{\mathsf{pr}}(\mathrm{d}u)$ can not be done explicitly, and even if Z was known, X is usually high-dimensional and the mapping $u \mapsto \mathcal{F}(u)$ is potentially non-linear. As such, one typically resorts to extract information from μ^y via sampling.

2.3.1 Markov chain Monte Carlo

These sampling techniques are broadly categorized as those which construct a Markov chain $\{u^n\}_{n\in\mathbb{N}}$ (c.f. Definition 3.1.1) starting from an initial probability measure μ_0 and whose invariant measure (c.f. Definition 3.1.4) is μ^y, in such a way that $\mathrm{Law}(u^n) \to \mu^y$ as $n \to \infty$, in some given sense. In practice, since the samples $u^n, n = 1, 2, \ldots, N$ are asymptotically distributed according to μ^y, one is then generally interested in running such a chain for a large N, discarding the first N_b samples as a so-called *burn-in*. These techniques constitute a set of quite powerful methods, with a broad body of literature devoted to their implementation and analysis, and a rather wide array of "generic" (i.e., problem-independent) algorithms to generate such chains. However, given that they usually require a large number of samples, they are undeniably costly. Furthermore, for large scale problems, for which generating a new sample u^{n+1} from u^n implies an evaluation of one or more computationally expensive forward models (such as a time-dependent, non-linear PDE), the

total computational cost associated to these methods can quickly become prohibitive. As stated in the previous chapter, the focus of this thesis is in the development, analysis and implementation of a special kind of MCMC techniques which exploit the structure of the problem and the availability of multiple approximations in a way that the total computational cost associated to the solution of the BIP is drastically reduced. Since MCMC methods are at the core of the work carried out in this thesis, we will postpone their presentation to Chapter 3, where we will discuss them in detail. For completeness, we review here alternative solution approaches that do not rely on the construction of Markov chains.

2.3.2 Approximate methods

These methods present a generally cheaper, albeit less accurate, alternative to MCMC methods. Instead of aiming to sample directly from μ^y, these set of techniques aim at first finding a probability measure ν^y such that ν^y is (i) *a sufficiently accurate* approximation of μ^y and (ii) the cost of sampling from ν^y is much lower than the cost of sampling from μ^y. This approach can be split into three main categories; *Laplace approximation and linearization* based methods [23, 56, 153], *variational methods* [55, 130, 129], and *transport methods* [110, 123, 145, 149].

Linearization and Laplace approximation

In short, these techniques proceed by first finding the *Maximum a Posteriori Point* (MAP) , and then linearizing $u \mapsto \mathcal{F}(u)$ around such a point u_{map}. If the prior is a Gaussian measure, this approach results then on a Gaussian measure $\nu^y = \mathcal{N}(\tilde{m}, \mathcal{K})$ approximating the posterior μ^y, with properly chosen $\tilde{m}, \mathcal{K}$. We now present this approach in more detail.
Let $(\mathsf{X}, \langle\cdot,\cdot,\rangle_{\mathsf{X}})$ be a separable Hilbert space, let $\mathsf{Y} = \mathbb{R}^M, M \geq 1$, equipped with the usual Euclidean norm and assume the following:

1. $\mu_{\text{pr}} = \mathcal{N}(m, \mathcal{C})$ for some $m \in \mathsf{X}$ and $\mathcal{C}$ a self-adjoint, positive-definite, trace-class covariance operator.

2. $\mu_{\text{noise}} = \mathcal{N}(0, \Gamma)$ for some symmetric, positive-definite matrix $\Gamma \in \mathbb{R}^{M\times M}$.

3. The mapping $u \mapsto \mathcal{F}(u)$ is Frechet-differentiable.

Under the additional assumption that the BIP is well-posed (i.e., Assumption 2.2.3 holds), one can pose the BIP as sampling from the posterior μ^y given by

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\text{pr}}}(u) = \frac{1}{Z}\exp(-\Phi(u;y)) = \frac{1}{Z}\exp\left(-\frac{1}{2}(y-\mathcal{F}(u))\Gamma^{-1}(y-\mathcal{F}(u))\right).$$

Furthermore, denote by $\mathsf{E} = \mathrm{Im}(\mathcal{C}^{1/2})$ endowed with the inner product $\langle\cdot,\cdot\rangle_{\mathcal{C}} = \langle\mathcal{C}^{-1/2}\cdot, \mathcal{C}^{-1/2}\cdot\rangle_{\mathsf{X}}$, and define $J(u) := \frac{1}{2}\|y-\mathcal{F}(u)\|^2_{\Gamma} + \frac{1}{2}\|u-m\|^2_{\mathcal{C}}$. The MAP u_{map} of μ^y, is defined as the point $u \in \mathsf{X}$ that asymptotically maximizes the μ^y-measure of a ball with radius ϵ centered around

it, divided by theμ_{pr}-measure of such a ball, as $\epsilon \to 0$. It is shown in [156] that an equivalent interpretation of the MAP is as the point u_{map} satisfying

$$u_{\text{map}} = \arg\min_{u \in \mathsf{E}} J(u). \tag{2.14}$$

Remark 2.3.1: *Notice that u_{map} need not be unique without any further assumptions on $\mathcal{F}$.*

Remark 2.3.2: *In the case where (in addition to the previous assumptions) X is a finite dimensional space (e.g., $\mathbb{R}^P$), u_{map} is understood as the point which maximizes the posterior density with respect to the Lebesgue measure.*

Denoting by $D\mathcal{F}$ the Frechet derivative of $\mathcal{F}$, one can then linearize $\mathcal{F}$ around u_{map} to obtain the following linear approximated model for the data:

$$y \approx \mathcal{F}(u_{\text{map}}) + D\mathcal{F}(u_{\text{map}})(u - u_{\text{map}}) + \eta. \tag{2.15}$$

It is then a consequence of Theorem 6.20 in [156] that the linearized model (2.15) induces a Gaussian probability measure $\nu^y = \mathcal{N}(\tilde{m}, \mathcal{K})$ approximating μ^y, where

$$\begin{aligned} \mathcal{K}^{-1} &:= [D\mathcal{F}(u_{\text{map}})]^* \Gamma^{-1} D\mathcal{F}(u_{\text{map}}) + \mathcal{C}^{-1}, \\ \tilde{m} &:= u_{\text{map}} \end{aligned}$$

where $[D\mathcal{F}(u_{\text{map}})]^* : \mathbb{R}^M \mapsto \mathsf{X}$ is defined as the adjoint of $[D\mathcal{F}(u_{\text{map}})]$ defined by $\langle [D\mathcal{F}(u_{\text{map}})]u, v\rangle_{\mathbb{R}^n} = \langle [D\mathcal{F}(u_{\text{map}})]^* v, u\rangle_{\mathsf{X}}, \forall v \in \mathbb{R}^M,\ u \in \mathsf{X}$. This Gaussian approximation of μ^y centered around its MAP is called *Laplace's approximation*.

From a computational perspective, problem (2.14) is solved using numerical optimization algorithms, such as Newton's method. Furthermore, it is commonly the case for the covariance operator $\mathcal{K}$ to be approximated by a low-rank matrix [23, 24]; this can in turn dramatically reduce the time required for sampling from ν^y.

Linearization techniques provide a first approach at approximating BIPs whose underlying mathematical model is extremely computationally expensive, and for which only a few draws from ν^y can be drawn under a reasonable budget. The work [23], for example, presents such an approach for a BIP in arising seismic-imaging at the global scale where the underlying mathematical model is a time-dependent PDE in 3 spatial dimensions. Furthermore, they also serve as a building block to some advanced MCMC methods, such as the so-called *generalized preconditioned Crank-Nicholson algorithm* (c.f. Section 3.4 and the original references [130, 144]).

These sort of techniques are also particularly useful when the posterior measure is well-concentrated around the MAP [153], which can occur, for example, in the case where the magnitude of the polluting noise goes to zero. This is the case of the work by Schillings et. al. [151], which utilizes importance sampling [3] to approximate integrals with respect to the posterior measure μ^y, using a

Laplace approximation as a biasing (importance) distribution in a (quasi) Monte Carlo quadrature. Furthermore, the work [151] analyzes the convergence of $\nu^y \to \mu^y$ as $\|\Gamma\|_{R^M} \to 0$, i.e., as the polluting noise goes to 0; a result closely related to the Bernstein-von Mises theorem for posterior consistency [25, 104]. Similarly, using a Laplace approximation as a biasing distribution in the context of importance sampling, has also been proposed in [8, 9, 102] to accelerate the computation of a so-called *inner-loop* integral for a problem arising in optimal experimental design. Of particular relevance to us is the work [9], where the authors create a (mesh-dependent) hierarchy of Laplace approximations, and exploit such a hierarchy using Multi-level Monte Carlo [59, 60].

Variational methods

Variational methods can be understood as a generalization of the previously discussed method. Indeed, given a family of probability measures H_Θ, parametrized by some $\theta \in \Theta$ (where Θ is a set of admissible parameters) the idea behind these methods is to find $\nu_\theta \in H_\Theta$ solving:

$$\nu_\theta^* = \arg \min_{\nu_\theta \in H_\Theta} \tilde{d}_{(\cdot)} \left(\nu_\theta, \mu^y \right),$$

for some suitable (pseudo-)distance $\tilde{d}_{(\cdot)}$ between probability measures, commonly taken as the KL divergence [130, 129]. We now present this approach in slightly more detail. Let $\nu_\theta \ll \mu_{\text{pr}}$ with

$$\frac{\mathrm{d}\nu_\theta}{\mathrm{d}\mu_{\text{pr}}}(u) := \frac{1}{Z_\nu} \exp\left(-\psi(u;\theta)\right),$$

for some measurable function $\psi(\cdot;\theta) : \mathsf{X} \to \mathbb{R}$, parametrized by $\theta \in \Theta$. We aim at finding $\nu_\theta \in H_\theta$ which minimizes:

$$d_{\text{KL}}(\nu_\theta, \mu^y) = \mathbb{E}_{\nu_\theta}\left[\log\left(\frac{\mathrm{d}\nu_\theta}{\mathrm{d}\mu^y}(u)\right)\right] \quad \text{or} \tag{2.16}$$

$$d_{\text{KL}}(\mu^y, \nu_\theta) = \mathbb{E}_{\mu^y}\left[\log\left(\frac{\mathrm{d}\mu^y}{\mathrm{d}\nu_\theta}(u)\right)\right], \tag{2.17}$$

provided that such Radon-Nikodym derivatives exist. Notice that, given the lack of symmetry of the KL divergence, if ν_θ^1 minimizes (2.16) and ν_θ^2 minimizes (2.17) over the same family of probability measures, one will have, in general, that $\nu_\theta^1 \neq \nu_\theta^2$. Furthermore, each formulation is better suited depending on the information available; in the case where one has access to some samples $\{u^n\}_{n=0}^N$ from μ^y obtained by a different methodology, one can aim at minimizing (2.17), since such an expectation can be approximated by a Monte Carlo quadrature using $\{u_n\}_{n=0}^N$. On the flip-side, if such samples are not available a priori, then minimizing (2.16) might be a more sensible approach. We will focus our presentation on the first direction (i.e., Equation (2.16)) and

reiterate that the reverse direction (Equation (2.17))) is also of interest. Under the assumption that $\nu_\theta \simeq \mu_{\text{pr}}$ and $\mu^y \simeq \mu_{\text{pr}}$, it is easy to show that Equation (2.16)) becomes

$$\tilde{J}(\theta) := d_{\text{KL}}(\nu_\theta, \mu^y) = \mathbb{E}_{\nu_\theta}[\Delta(u;\theta)] - \log\left(\mathbb{E}_{\nu_\theta}\left[\exp(-\Delta(u;\theta))\right]\right),$$

where $\Delta(u;\theta) := \Phi(u;y) - \psi(u;\theta)$. It is not difficult to see that minimizing $\tilde{J}$ over Θ is equivalent to minimizing $d_{\text{KL}}(\nu_\theta, \mu^y)$ over Θ, which is typically done using, e.g., the Robbins-Monro algorithm [137] as it in [110, 130, 129]. A common choice of H_Θ is the space of all Gaussian measures, which are, of course, uniquely characterized by their mean and covariance operator. This is a natural choice for many problems arising in infinite-dimensional spaces [129], since $\psi(\cdot,\theta)$ is known, as a consequence of the Feldman-Hajek theorem (see, e.g., [37]), where the parameter θ characterizes the mean and covariance operator of such a Gaussian approximation. Notice that, in the case of (2.16), this particular choice is quite similar to the linearization method discussed in the previous section.

Normalizing flows and measure transport

An additional set of techniques that has gained wide-spread popularity in recent years is sampling via measure transport [165]. Throughout this subsection we will limit ourselves to the finite dimensional case (i.e., $\mathsf{X} = \mathbb{R}^K$, $K \geq 1$). Let $u, z \in \mathbb{R}^K = \mathsf{X}$ and let ν be a probability measure on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ which has a density with resprct to the Lebesgue measure, satisfying the assumption that sampling from ν and evaluating its Lebesgue density at a given point is much cheaper than doing so for μ^y. With a slight abuse of notation, we will write $\nu(u)$ as the (Lebesgue) density of a measure ν evaluated at a point $u \in \mathsf{X}$. Given a diffeomorphism $T : \mathsf{X} \to \mathsf{X}$ such that $\mu^y = T_\sharp \nu$, we can generate samples from μ^y by first sampling $z \sim \nu$, and then setting $T(z) = u \sim \mu^y$. This procedure induces the change of probability density:

$$\mu^y(u) = \nu(z)\left|\det J_T(z)\right|^{-1} = \nu\left(T^{-1}(u)\right)\left|\det J_{T^{-1}}(u)\right|,$$

where we have used the same abuse of notation to denote by $\mu^y(u)$ the Lebesgue density of μ^y evaluated at $u \in \mathsf{X}$. We will refer to these techniques as *Normalizing Flows*.
The *crux* of this method is to find such a diffeomorphism T. This is, in general not a trivial task and often one needs to instead look for some optimal $T = T_\theta$ over a set of parametric diffeomorphisms $\mathcal{T}_\theta$, satisfying some chosen concept of optimality (we will be more precise about this briefly). Furthermore, if the target distribution μ^y has, loosely speaking, very complicated structure, a simple T_θ (such as a scale or shift) will not work, thus, one typically constructs T_θ as a composition of L simpler diffeomorphisms:

$$T_\theta = T_\theta^L \circ T_\theta^{L-1} \circ \ldots \circ T_\theta^1,$$

which induce

$$z_k = T_\theta^{k-1}(z_{k-1}), \quad k = 1, \dots, L,$$

and

$$\mu^y(u) \approx \nu_\theta = \nu(z) \prod_{k=1}^{L} \left(\left| \det J_{T_\theta^k}(z_{k-1}) \right|^{-1} \right).$$

This poses a clear issue from a computational perspective; the complexity associated to computing the determinant of a $K \times K$ matrix is, in general, $O(K^3)$, thus the total cost of evaluating $\mu^y(u)$ is of $O(LK^3)$. One can circumvent this issue by setting $\mathcal{T}_\theta$ as the set of diffeomorphisms in X parametrized by θ, such that the determinant of their Jacobian has a smaller complexity than $O(LK^3)$. We now proceed to briefly review three appraches to this.

Optimal Triangular Transformations (OTT)

A first approach presented by Marzouk et. al., [110, 124, 125] is to consider $\mathcal{T}_\theta$ as the space of triangular transformations parametrized by θ, i.e., diffeomorphisms of the form

$$T_\theta(u) = \begin{pmatrix} f_\theta^1(u_1) \\ f_\theta^2(u_1, u_2) \\ \vdots \\ f_\theta^K(u_1, u_2, \dots, u_K) \end{pmatrix}, \quad \forall u \in \mathsf{X}, \tag{2.18}$$

where, for any $i = 1, \dots, K$, f_θ^i is the i^{th} component of T_θ, parametrized by $\theta \in \Theta$. Notice that the Jacobian of such a transformation is lower triangular, and as such, its determinant can be computed in K operations. Furthermore, it is known (see, e.g., [165]) that, whenever $\mu^y \ll \nu$, then there exists a unique transformation of the form (2.18) satisfying $\mu^y = T_\sharp \nu$. Such a diffeomorphism is known as the *Knothe-Rosenblatt rearrangement* [165]. There is some flexibility in the choice of $f_\theta^i : \mathbb{R}^i \to \mathbb{R}$, $i = 1, 2, \dots, K$; as stated in [110], this family of functions can be, e.g., multivariate polynomial, or radial basis functions. Having defined T_θ, the OTT approach then proceeds by obtaining $T_\theta = \arg\min_{\mathcal{T}_\theta} \mathrm{d_{KL}}(T_{\theta,\sharp}\nu, \mu^y)$. Alternatively, assuming that one has access to some samples $\{u_n\}_{n=0}^N \sim \mu^y$, one can instead create a diffeomorphism $S_\theta : \mathsf{X} \to \mathsf{X}$ where $S_{\theta,\sharp}\mu^y = \nu$ by minimizing $\mathrm{d_{KL}}(S_{\theta,\sharp}\mu^y, \nu)$ (i.e., the reverse direction of the KL divergence) over $\mathcal{T}_\theta$. Once such a diffeomorphism S_θ has been created, one can then generate samples from $\nu_\theta \approx \mu^y$ by sampling from $S_\theta^{-1}\nu$. This approach has been proposed in [125] in the context of a delayed-rejection Metropolis-Hastings [21] to accelerate the solution of a BIP using MCMC methods (c.f. Chapter 3).

A similar approach to OTT has been developed by the machine learning community in the context of *generative models* [55, 123]. Contrary to the OTT approach, the following two methods require

a (potentially large) set of samples $\{u_n\}_{n=0}^N \sim \mu^y$ obtained a priori. Nevertheless, they are still useful in the context of BIPs. For the next two methods we will consider the case where $T = T_\theta$ is a deep neural network parameterized by θ with L layers. Typically, works in the machine-learning community (see, e.g., the review [123]) solve the optimization over $\mathcal{T}_\theta$ using *maximum-likelihood* [55].

Autorregressive flows

Autoregressive Flows (AF) model the (joint) density $\nu_\theta \approx \mu^y(u)$ as the product of conditional densities $\prod_i \nu_\theta(u_i \mid u_{1:i-1})$. A common example in literature is when the conditional densities are parametrized as Gaussians:

$$\begin{aligned} \nu_\theta(u_i \mid u_{1:i-1}) &= \mathcal{N}(u_i \mid m_i, \exp(\sigma_i)^2), \\ \text{where} \qquad m_i &= T_{m_i}(u_{1:i-1}) \\ \text{and} \qquad \sigma_i &= T_{\sigma_i}(u_{1:i-1}), \end{aligned}$$

where $(m_i, \sigma_i) := \theta_i$. In the above equations, the mean and standard deviations of each conditional distribution are computed using (parameterized) functions of all previous variables. The above can alternatively be written as:

$$u_i = m_i(u_{1:i-1}) + \exp(\sigma_i(u_{1:i-1}))z_i \qquad i = 1, \dots, K$$

This last equation shows how the auto-regressive model can be viewed as a transformation f from the random variables $z \in \mathbb{R}^K$ to the data $u \in \mathbb{R}^K$.
Clearly, in this case, u_i depends only on the components of z that are lower than or equal to i but not any of the higher ones. This is a type of triangular transport map [149] such as the ones used in [126].
AFs tend to be quite *expressive* (i.e., are able to represent a wide class of functions [123]),however, there is a caveat associated to these methods: sampling (i.e., generating u from z) is slow, since this process needs to be done sequentially, i.e., one must first obtain u_1, then u_2, and so on up to u_K. On the flip-side, determining z from u is relatively faster; each of the above equations can be solved for z_i at the same time, resulting in

$$z_i = \frac{u_i - T_{m_i}}{\exp(T_{\sigma_i})} \qquad i = 0, \dots, K-1.$$

This inverse pass (obtaining z from u) is what is used in the likelihood calculations used to train the model. To summarize this approach, it is computationally expensive to generate samples u from it, however, evaluating the density of the approximating distribution is relatively cheap. In summary: Samples slowly but trains (i.e., evaluates density) relatively quickly.

In practice, one constructs several layers of AFs together with a permutation layer. This is done so that there is some *mixing* between the components.

RealNVP and NICE

These are ideas presented in [43, 44] aimed at reducing the sampling cost of AFs. The RealNVP method [44] considers a reduced version of AF for which:

$$\begin{aligned} u_i &= z_i && i = 1, \dots, d \\ u_i &= m_i + \exp(\sigma_i) z_i && i = d+1, \dots K \end{aligned}$$

where

$$\begin{aligned} m_i &= T_{m_i}(z_1, \dots, z_d) \\ \sigma_i &= T_{\sigma_i}(z_1, \dots, z_d) \end{aligned}$$

Hence, the transformation leaves the first d dimensions of z unchanged, while the reaming $K - d$ are transformed by a shift m and scalar term $\exp(\sigma)$, construed in such a way that m and $\exp(\sigma)$ are some given parametric functions, depending only on the first d components of z. Note that, in this case, both the forward and backward pass of the flow can be done fully in parallel. Its predecessor, [43], omits the scale term $\exp(\sigma_i) z_i$ altogether. Once again, one stacks several layers of RNVP together with a permutation in order to improve expressibility. There is, of course, a catch: such a simple form means the flow typically needs a higher number of diffeomorphisms (i.e., a higher K value) to be able to describe *complicated* distributions [123].

2.3.3 Sequential methods

Sequential methods (also known as *filtering* or *particle* methods), approach the solution to BIP by building knowledge on the posterior μ^y and/or u sequentially. More precisely, let $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$, $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}})$ be separable Banach spaces with associated Borel σ-algebra $\mathcal{B}(\mathsf{X})$, $\mathcal{B}(\mathsf{Y})$. Furthermore, assume there exists a sequence of probability measures ν_i, $i = 1, 2, \dots I$ on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, approximating μ^y, with the property that (i) $\nu_0 = \mu_{\mathsf{pr}}$, (ii) $\mu^y = \nu_I$, and (iii) $\nu_i \simeq \nu_j \; \forall i, j = 1, \dots, I$. Notice then that the posterior of interest can be written as

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u) \propto \frac{\mathrm{d}\nu_1}{\mathrm{d}\mu_{\mathsf{pr}}}(u) \times \frac{\mathrm{d}\nu_2}{\mathrm{d}\nu_1}(u) \times \cdots \times \frac{\mathrm{d}\nu_I}{\mathrm{d}\nu_{I-1}}(u). \tag{2.19}$$

Given some μ^y-integrable quantity of interest QoI, One can then approximate $\mathbb{E}_{\mu^y}[\mathsf{QoI}]$ using a Monte Carlo quadrature with N samples by first sampling $\{u_n\}_{n=1}^N \overset{iid}{\sim} \nu_0$, and then applying *importance sampling* [3] sequentially, with biasing function given by each of the Radon-Nikodym

derivatives in (2.19). It is often the case that posterior measures concentrate around a small region of the prior, and as such, μ_{pr} is not necessarily a good approximation to μ^y, which might in turn have undesirable effects in the change of measure being carried out with the importance sampling. In order to avoid this, one typically needs to apply a ν_i-invariant Markov transition kernel (c.f. Definitions 3.1.2 and 3.1.4) at the i^{th} step to the empirical measure of the ν_i-distributed samples (also called *particles*). Before applying this one step of the Markov transition kernel, each particle $\{u_n\}_{n=1}^N$ is re-sampled with weight $w_i : \mathsf{X} \to [0, 1]$

$$w_i(u_n) = \frac{\frac{\mathrm{d}\nu_i}{\mathrm{d}\nu_{i-1}}(u_n)}{\sum_{m=1}^N \frac{\mathrm{d}\nu_i}{\mathrm{d}\nu_{i-1}}(u_m)}.$$

The crux of this method relies then on the construction of the approximating measures ν_i. These approximations can be, e.g., based on temperatures [14] (as presented in Chapters 1 and 4), discretization parameters for the underlying mathematical model $\mathcal{F}$ (as discussed in Chapters 1, 5 and 6), [13], or both [96], where the previously discussed methodology is used in combination with Multi-level Monte Carlo ideas [60]. Similar hierarchy-exploiting ideas have been presented by [5, 29, 72, 79] in the context of filtering problems for partially observed diffusions.

3 Markov chain Monte Carlo

In this chapter, we review MCMC methods from their theory to their implementation. More precisely, we begin this chapter by recalling some basic concepts on Markov kernels, the workhorse of MCMC, and then proceed to present the convergence theory of these methods. We then conclude this chapter with a review (of an arbitrary selection of) some common MCMC techniques. Similarly to Chapter 2, the material presented in this chapter covers a wide variety of topics, which, for the sake of brevity, makes such a presentation necessarily short. For a more in-depth discussion of the topics presented in this chapter, we refer the interested reader to, e.g., the monograph of Meyn and Tweedie [113], for a thorough presentation of classical results in the theory of Markov chains; to the book [21], for a detailed introduction to some MCMC methods and their applications, and to the doctoral dissertations [142] and [153], together with the survey [32], for modern results regarding the convergence and implementation of MCMC methods in function spaces.
We remark that, for the most part, this is a review chapter where we recall some well-known results and methods in the MCMC literature, and that almost no new material is discussed, with the exception of Theorem 3.3.2, which is taken from the appendix of our work [108], and which presents a bound for the non-asymptotic mean-square error of an ergodic estimator obtained using non-reversible Markov chains. Additionally, Lemma 3.4.1 (a slight generalization of Theorem 1 in [160]), and the *ν-MALA algorithm* (a variation of the ∞-MALA of [12] and the pCN algorithm of [130]), are also, to the best of the author's knowledge, new (albeit rather incremental) results.

3.1 Markov Chain Monte Carlo

Let $(\mathsf{X}, \mathcal{B}(\mathsf{X}), \mu)$ be a probability measure space, and let $\mathsf{QoI} : \mathsf{X} \to \mathbb{R}$ be an μ-integrable function that we will call *quantity of interest*. A central task in this work is to compute expectations of the quantity of interest with respect to a reference probability measure, written as:

$$\mu(\mathsf{QoI}) := \mathbb{E}_{\mu}[\mathsf{QoI}] := \int_{\mathsf{X}} \mathsf{QoI}(u)\, \mu(\mathrm{d}u). \tag{3.1}$$

Ultimately, one of the goals of this thesis is to construct and analyze efficient algorithms for estimating expectations of the form (3.1) using MCMC techniques.
For the purposes of this work, we will consider μ to be the posterior measure μ^y, or some hierarchical approximation of it (e.g., a tempered version of μ^y, or a posterior arising from a coarse approximation of the forward mapping operator $\mathcal{F}$). Usually, it is not possible to sample directly from μ using so-called *direct* methods (e.g., via simple transformations of random variables, in-

version of the cumulative distribution function, etc). Instead, one such way of sampling from μ is to use MCMC methods, which, at their core, create a Markov chain $\{u^n,\ n \in \mathbb{N}_0\}$ whose invariant probability measure (c.f. Definition 3.1.4) is μ. Once such a chain has been obtained up to a certain iteration N, one can approximate $\mathbb{E}_\mu[\mathsf{QoI}]$ with the usual ergodic estimator, i.e.,

$$\mathbb{E}_\mu[\mathsf{QoI}] \approx \frac{1}{N}\sum_{n=0}^{N} \mathsf{QoI}(u^n).$$

We formalize these concepts in the following.

Definition 3.1.1 (Markov chain): *Let μ^0 be a probability measure on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, and consider an ordered sequence of random variables $\{u^n,\ n \in \mathbb{N}_0\}$ taking values in X. We say that $\{u^n,\ n \in \mathbb{N}_0\}$ is a Markov chain if (i) $u^0 \sim \mu^0$ and (ii) it fulfills the Markov property; meaning, that for any $i \geq 1$, it holds*

$$\mathbb{P}\left(u^{i+1} \in A | u^0 = \tilde{u}^0, \dots, u^i = \tilde{u}^i\right) = \mathbb{P}\left(u^{i+1} \in A | u^i = \tilde{u}^i\right), \quad A \in \mathcal{B}(\mathsf{X}), \tag{3.2}$$

where, for any $j \in \mathbb{N}$, we denoted by $\tilde{u}^j$ the realization of the random variable u^j.

Equation (3.2) motivates the definition of *Markov transition kernel* [143]:

Definition 3.1.2 (Markov kernel): *A* Markov kernel *(some times referred to as Markov Transition Kernel) on a Banach space $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$ is a function $p : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \to [0,1]$ such that*

1. *For each A in $\mathcal{B}(\mathsf{X})$, the mapping $\mathsf{X} \ni u \mapsto p(u, A)$, is a $\mathcal{B}(\mathsf{X})$-measurable real-valued function.*

2. *For each u in X, the mapping $\mathcal{B}(\mathsf{X}) \ni A \mapsto p(u, A)$, is a probability measure on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$.*

Loosely speaking, $p(u, A)$ can be interpreted as the (conditional) probability of moving to a set $A \in \mathcal{B}(\mathsf{X})$ given that the chain is in a current state $u \in \mathsf{X}$. Similarly, we can define the *n-step Markov transition kernel* given by the recursion:

$$p^n(u, A) := \int_{\mathsf{X}} p^{n-1}(z, A) p(u, \mathrm{d}z), \quad p^1(u, A) = p(u, A), \quad \forall A \in \mathcal{B}(\mathsf{X}). \tag{3.3}$$

The Markov operator [138] associated to a Markov transition kernel is defined as follows:

Definition 3.1.3 (Markov operator): *Let $p : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \mapsto [0,1]$ be a Markov kernel on a Banach space X, let $f : \mathsf{X} \mapsto \mathbb{R}$ be a measurable function on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, and let $\mu \in \mathcal{M}(\mathsf{X})$. We denote by P the* Markov operator *(sometimes we will refer to it as Markov transition operator), which*

acts to the left on measures, $\mu \mapsto \mu P \in \mathcal{M}(\mathsf{X})$*, and to the right on functions,* $f \mapsto Pf$*, measurable on* $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$*, such that*

$$(\mu P)(A) = \int_{\mathsf{X}} p(u, A)\mu(\mathrm{d}u), \quad \forall A \in \mathcal{B}(\mathsf{X}),$$
$$(Pf)(u) = \int_{\mathsf{X}} f(z)p(u, \mathrm{d}z), \quad \forall u \in \mathsf{X}.$$

Similarly, for any $n \in \mathbb{N}$, we denote the n-step Markov transition operator associated to (3.3) by P^n, which clearly satisfies $P^{n+1} = P^n P$. Throughout this work, we will make the distinction between Markov kernel, denoted by lower case letters, and Markov operator, written with an upper case letter. We begin with the definition of invariant measure.

Definition 3.1.4 (μ-invariance): *We say that a Markov operator* P *is* μ-invariant *if* $\mu P = \mu$*, i.e., if it holds that*

$$(\mu P)(A) = \int_{\mathsf{X}} p(u, A)\mu(\mathrm{d}u) = \mu(A), \quad \forall A \in \mathcal{B}(\mathsf{X}).$$

Let $r \in [0, \infty]$. Given a μ-invariant Markov operator $P : L_r(\mathsf{X}, \mu) \to L_r(\mathsf{X}, \mu)$, we define its norm by

$$\|P\|_{L_r \to L_r} := \sup_{\|f\|_{L_r}=1} \|Pf\|_{L_r}, \quad f \in L_r,$$

with $L_r(\mathsf{X}, \mu)$ defined as in (2.5). Of particular importance is the operator norm in the space L_r^0, which induces the L_r*-spectral gap* $\gamma_r[P]$ defined by

$$\gamma_r[P] := 1 - \|P\|_{L_r^0 \to L_r^0},$$

It will be shown shortly that this quantity plays a crucial role in the convergence of Markov chains. Given a Markov operator $P : L_r(\mathsf{X}, \mu) \to L_r(\mathsf{X}, \mu)$, we denote by $P^* : L_{r'}(\mathsf{X}, \mu) \to L_{r'}(\mathsf{X}, \mu)$ its *adjoint operator*, where $\frac{1}{r'} + \frac{1}{r} = 1$. Letting $f : \mathsf{X} \to \mathbb{R}$ be a μ-integrable function, and denoting $\hat{\mu} : L_r(\mathsf{X}, \mu) \to L_r(\mathsf{X}, \mu)$ the "averaging operator" that associates to f the constant function $\hat{\mu} f := \int_{\mathsf{X}} f(u)\mu(\mathrm{d}u)$, it can be shown (see, e.g., [143, page 42]) that

$$\|P - \hat{\mu}\|_{L_r \to L_r} = \|P^* - \hat{\mu}\|_{L_{r'} \to L_{r'}}.$$

Moreover, we define the so-called *pseudo-spectral gap* ([127]) of a Markov operator $P : L_2(\mathsf{X}, \mu) \to L_2(\mathsf{X}, \mu)$ as follows:

$$\gamma_{\mathrm{ps}}[P] := \max_{k \geq 1} \left\{ \gamma_2[(P^*)^k P^k]/k \right\}, \ k \in \mathbb{N}. \tag{3.4}$$

Given a μ-invariant operator P, it follows from [143, Lemma 3.9] that for any measure $\nu \in \mathcal{M}_r(\mathsf{X}, \mu)$

$$\frac{\mathrm{d}(\nu P)}{\mathrm{d}\mu} = P^* \left(\frac{\mathrm{d}\nu}{\mathrm{d}\mu} \right)$$

A related concept to invariance is that of reversibility:

Definition 3.1.5 (Reversibility): *A Markov kernel $p : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \mapsto [0, 1]$ is said to be reversible (or μ-reversible) with respect to a measure $\mu \in \mathcal{M}(\mathsf{X})$ if*

$$\int_B p(u, A)\mu(\mathrm{d}u) = \int_A p(u, B)\mu(\mathrm{d}u), \quad \forall A, B \in \mathcal{B}(\mathsf{X}). \tag{3.5}$$

which is sometimes written in the short-hand form

$$p(u, \mathrm{d}v)\mu(\mathrm{d}u) = p(v, \mathrm{d}u)\mu(\mathrm{d}v). \tag{3.6}$$

It is straightforward to verify that if a Markov kernel is reversible with respect to a probability measure μ, then its associated Markov operator P has μ as an invariant measure. Indeed, for any set $A \in \mathcal{B}(\mathsf{X})$,

$$(\mu P)(A) = \int_{\mathsf{X}} p(u, A)\mu(\mathrm{d}u) \overset{\text{(by reversibility)}}{=} \int_A p(u, \mathsf{X})\mu(\mathrm{d}u) = \int_A \mu(\mathrm{d}u) = \mu(A),$$

where the second-to-last equality comes from the fact that for any $u \in \mathsf{X}$, $\int_{\mathsf{X}} p(u, \mathrm{d}z) = 1$ (since $p(u, \cdot)$ is a probability measure on X). The reverse is not true, in general. A reversible Markov operator $P : L_2(\mathsf{X}, \mu) \to L_2(\mathsf{X}, \mu)$ (resp. $\mathcal{M}_2(\mathsf{X}, \mu) \to \mathcal{M}_2(\mathsf{X}, \mu)$) is known to be self-adjoint; indeed, for any $f, g \in L_2(\mathsf{X}, \mu)$, one has that

$$\begin{aligned} \langle Pf, g \rangle_{L_2} &= \int_{\mathsf{X}} (Pf)(u) g(u) \mu(\mathrm{d}u) = \int_{\mathsf{X}} \int_{\mathsf{X}} p(u, \mathrm{d}v) f(v) g(u) \mu(\mathrm{d}u) \\ &= \int_{\mathsf{X}} \int_{\mathsf{X}} p(v, \mathrm{d}u) f(v) g(u) \mu(\mathrm{d}v) = \langle f, Pg \rangle \text{ (by reversibility.)} \end{aligned}$$

Similarly, for any $\nu, \pi \in \mathcal{M}_2(\mathsf{X}, \mu)$,

$$\begin{aligned}
\langle \nu P, \pi \rangle_{\mathcal{M}_2} &= \int_{\mathsf{X}} \frac{\mathrm{d}(\nu P)}{\mathrm{d}\mu}(u) \frac{\mathrm{d}\pi}{\mathrm{d}\mu}(u) \mu(\mathrm{d}u) = \int_{\mathsf{X}} \frac{\mathrm{d}\pi}{\mathrm{d}\mu}(u) \left(\int_{\mathsf{X}} p(v, \mathrm{d}u) \nu(\mathrm{d}v) \right) \\
&= \int_{\mathsf{X}} \int_{\mathsf{X}} \frac{\mathrm{d}\pi}{\mathrm{d}\mu}(u) p(v, \mathrm{d}u) \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(v) \mu(\mathrm{d}v) \\
&= \int_{\mathsf{X}} \int_{\mathsf{X}} \frac{\mathrm{d}\pi}{\mathrm{d}\mu}(u) p(u, \mathrm{d}v) \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(v) \mu(\mathrm{d}u) \quad \text{(by reversibility)} \\
&= \int_{\mathsf{X}} \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(v) \left(\int_{\mathsf{X}} p(u, \mathrm{d}v) \pi(\mathrm{d}u) \right) = \int_{\mathsf{X}} \frac{\mathrm{d}\nu}{\mathrm{d}\mu}(u) \frac{\mathrm{d}(\pi P)}{\mathrm{d}\mu}(u) \mu(\mathrm{d}u) = \langle \nu, \pi P \rangle_{\mathcal{M}_2}.
\end{aligned}$$

This self-adjointness plays an important role in the construction of MCMC methods, as shown in [127, 143]. On the one hand, it is known (see, e.g., [15, 89, 116]) that some non-reversible chains converge faster to their invariant measure. On the other hand, under some technical conditions, one can obtain sharper error bounds when computing ergodic estimators with samples obtained from reversible chains (see e.g., [143] and Theorems 3.3.2 and 3.3.1).
It is known (see, e.g, [143, Lemma 3.8]) that for any $r \in [1, \infty]$, Markov operators $P : L_r(\mathsf{X}, \mu) \to L_r(\mathsf{X}, \mu)$ with invariant measure μ induce a weak contraction, i.e., for any $f \in L_r(\mathsf{X}, \mu)$ it follows that

$$\|Pf\|_{L_r} \leq \|f\|_{L_r}, \text{ and } \|P\|_{L_r \to L_r} \leq 1. \tag{3.7}$$

Furthermore, notice that for the particular case where $f = 1$, one has $Pf = \int_{\mathsf{X}} 1 p(u, \mathrm{d}v) = p(u, \mathsf{X}) = 1$, i.e., the function $f = 1$ is an eigen-function of the operator P associated to the eigenvalue $\lambda = 1$ and $\|P\|_{L_r} = 1$.

3.2 Convergence

It is usually the case that $\mu^0 \neq \mu$, i.e., the Markov chain is not started from stationarity. This motivates the convergence study of $\mu^0 P^n \to \mu$. We begin by defining a notion of convergence for Markov chains.

Definition 3.2.1 (Geometric ergodicity): *Let $r \in [1, \infty]$. Given a μ-invariant Markov operator P and a probability measure $\mu^0 \in \mathcal{M}_r(\mathsf{X}, \mu)$, we say that the Markov chain $\{u^n,\ n \in \mathbb{N}\}$ generated by P with $u^0 \sim \mu^0$ is* M_r-geometrically ergodic *if there exists $\rho \in (0, 1)$ and a finite $\mathcal{M}_{\mu^0} \in \mathbb{R}_+$ such that*

$$\left\| \mu^0 P^n - \mu \right\|_{\mathcal{M}_r} \leq M_{\mu^0} \rho^n, \quad n \in \mathbb{N}. \tag{3.8}$$

Alternatively, given a bounded function $M : \mathsf{X} \to \mathbb{R}_+$*, we say that the Markov chain* $\{u^n, n \in \mathbb{N}\}$ *generated by* P *is* M_r-geometrically ergodic *if there exits* $\rho \in (0,1)$ *such that*

$$\|p^n(u,\cdot) - \mu(\cdot)\|_{\mathcal{M}_r} \leq M(u)\rho^n, \quad \forall n \in \mathbb{N}, \text{ for } \mu\text{-a.e } u \in \mathsf{X}. \tag{3.9}$$

We say that a chain is uniformly ergodic *if either* M_{μ^0} *in* (3.8) *is independent of the initial measure* μ^0 *or if* $M(u)$ *in* (3.9) *is uniformly bounded.*

In practice, one typically runs the chain $\{u^n\}_{n=0}^{N+n_b}$ for $N + n_b$ iterations, where the first n_b samples are discarded to reduce the bias associated to not starting at the invariant distribution (this is the so-called *burn-in period*). However, it is difficult, in general, to quantify an appropriate value (or choice) of n_b (see, e.g., [83] and [143]).
There are two closely related approaches for studying the convergence of Markov chains in general state spaces ([65, 113, 143]), namely:

1. **Spectral methods**. A first approach is to examine the spectral properties of the operator P. More precisely, if the Markov transition operator $P : L_r(\mathsf{X},\mu) \to L_r(\mathsf{X},\mu)$ has a positive L_r-spectral gap, then, it is relatively straightforward to show that P generates an $M_{r'}$-geometrically ergodic chain, where $\frac{1}{r} + \frac{1}{r'} = 1$. Investigating convergence of Markov chains in terms of the spectral properties of P (in particular, the existence of an L_2-spectral gap) can be traced back to the work [98], which relies upon the so-called *conductance* arguments presented in [27] (which has been reprinted in [28]). A closely related approach is presented in the work [65], where the authors consider the contractive properties of P on a Wasserstein metric in order to show the existence of an L_2-spectral gap (and hence, geometric ergodicity) for the preconditioned Crank-Nicolson (pCN) algorithm in function spaces (c.f Section 3.4). Spectral arguments have also been used in the analysis of hierarchical methods, in particular in the convergence analysis of a type of parallel tempering presented in [171]. The convergence analysis for our *Generalized parallel tempering* method ([95]) presented in Chapter 4 also relies upon these arguments. Furthermore, spectral methods can be used to obtain rigorous non-asymptotic error bounds on the Mean Squared Error (MSE) of ergodic estimators; the work [143] presents one such bound under the additional assumption of reversibility (c.f Theorem 3.3.1). A similar bound that does not require the extra assumption of reversibility (at the cost of being less sharp) is presented in Theorem 3.3.2 (taken from our work [108, Appendix]).

2. **Splitting methods**. An independently developed (and perhaps more classical) approach for studying the convergence of Markov chains is based on *renewal theory, splitting, and coupling arguments* as in [82, 83, 99, 113, 120, 139, 140, 159]. In this case, one requires the chain to satisfy certain conditions (c.f. Definitions 3.2.2 through 3.2.8), under which one can study the convergence theory of the Markov chain by splitting its trajectory into independent blocks [120], and then using the coupling inequality (c.f. Equation (3.12)) to bound its convergence in terms of the total variation distance. This approach has been

used to study the convergence of classical MCMC algorithms, such as the Metropolis-Hastings and the Gibbs sampler [140, 138], and will be used to study the convergence of our Multi-level Markov chain Monte Carlo algorithms presented in Chapters 5 and 6. It is worth mentioning that, although these coupling arguments were originally motivated in the theoretical study of convergence of Markov chains, recent works, such as [53, 70, 76, 136], have proposed practical coupling algorithms for Markov chains, that can be used in the context of *unbiased estimation*. These coupling techniques will play a significant role for the methods presented in Chapter 6.

We now present a summary of these two approaches. The following two results are of central importance for the first (i.e., spectral) approach:

Lemma 3.2.1 (Spectral gap implies geometric ergodicity): *For any $r' \in [1, \infty]$, let $P : L_{r'}(\mathsf{X}, \mu) \to L_{r'}(\mathsf{X}, \mu)$ be a Markov operator with a positive $L_{r'}$-spectral gap ; that is, $1 - \|P\|_{L^0_{r'} \to L^0_{r'}} > 0$. Then, the chain generated by P is $\mathcal{M}_r$-geometrically ergodic, where $\frac{1}{r} + \frac{1}{r'} = 1$.*

Proof. This is the proof of [143, Proposition 3.17], however, we include it for completeness. Given some initial measure $\mu^0 \in \mathcal{M}_r(\mathsf{X}, \mu)$, and setting $(1 - \gamma_r[P]) = \rho$, we have that

$$\left\|\mu^0 P^n - \mu\right\|_{\mathcal{M}_r} = \left\|\left(\mu^0 - \mu\right) P^n\right\|_{\mathcal{M}_r} = \left\|(P^*)^n \frac{\mathrm{d}\left(\mu^0 - \mu\right)}{\mathrm{d}\mu}\right\|_{L_{r'}}.$$

Since the function $f = \frac{\mathrm{d}(\mu^0 - \mu)}{\mathrm{d}\mu} \in L^0_{r'}(\mathsf{X}, \mu)$, we then have that

$$\begin{aligned}\left\|(P^*)^n \frac{\mathrm{d}\left(\mu^0 - \mu\right)}{\mathrm{d}\mu}\right\|_{L_{r'}} &\leq \|(P^*)^n\|_{L^0_{r'} \to L^0_{r'}} \left\|\mu^0 - \mu\right\|_{\mathcal{M}_r} = \|P\|^n_{L^0_{r'} \to L^0_{r'}} \left\|\mu^0 - \mu\right\|_{\mathcal{M}_r} \\ &= \underbrace{(1 - \gamma_{r'}[P])^n}_{\rho^n} \underbrace{\left\|\mu^0 - \mu\right\|_{\mathcal{M}_r}}_{M_{\mu^0}} = M_{\mu^0} \rho^n.\end{aligned}$$

□

The converse of Lemma 3.2.1 is true for the L_2-spectral gap, under the additional assumption of reversibility, as shown in [138, Theorem 2.1].

Lemma 3.2.2 (L_2-Geometric ergodicity and reversibility imply an L_2-spectral gap): *Let P be a μ-reversible Markov transition operator. Then, P has a positive $L_2(\mathsf{X}, \mu)$-spectral gap if and only if P is $L_2(\mathsf{X}, \mu)$-geometrically ergodic.*

Proof. See [138, Theorem 2.1]. □

We now present some definitions and results necessary for the splitting method approach.

Definition 3.2.2 (ψ-Irreducibility): *Given a strictly positive measure ψ on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, we say that a Markov kernel $p : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \to [0, 1]$ is* ψ-irreducible *if for all measurable sets $A \in \mathcal{B}(\mathsf{X})$ with $\psi(A) > 0$ and for all $u \in \mathsf{X}$, there exists a positive integer n, possibly depending on u and A such that*

$$p^n(u, A) > 0.$$

We say that a chain is ψ-irreducible if it is generated by a ψ-irreducible Markov transition kernel.

ψ-Irreducibility is the weakest form of stochastic stability, and can be understood as a statement on the "accessibility " of the state space; loosely speaking, this accessibility can be understood as "how easy it is to reach a set $A \in \mathcal{B}(\mathsf{X})$ from a point $u \in \mathsf{X}$, when using a Markov transition kernel $p(\cdot, \cdot)$".

Definition 3.2.3 (Harris recurrence): *A set $A \in \mathcal{B}(\mathsf{X})$ is called* recurrent *if*

$$\mathbb{P}(\textit{chain visits } A \textit{ infinitely often}) = 1.$$

We say that a Markov chain is Harris Recurrent *if it is ψ-irreducible and every set $A \in \mathcal{B}(\mathsf{X})$ with $\psi(A) > 0$ is recurrent. Similarly, we say that an operator P is Harris recurrent if it induces a Harris recurrent chain.*

Intuitively, one would expect that the previous condition must be satisfied for a Markov chain to converge. Indeed, Meyn and Tweedie [113, Theorem 10.0.1] present the following result.

Theorem 3.2.1: *Let $\{u^n,\ n \in \mathbb{N}\}$ be a Harris recurrent chain generated by a Markov operator P. Then, $\{u^n,\ n \in \mathbb{N}\}$ has a unique (up to constant multiples) invariant measure $\tilde{\mu}$ (notice that $\tilde{\mu}$ is not necessarily a probability measure).*

Furthermore, it is known from Theorem 17.0.1 (i) in [113], that, given a Harris recurrent, μ-invariant Markov chain, the law of large numbers holds for any $g \in L_1(\mathsf{X}, \mu)$. It is not always easy to show that a Markov chain is recurrent, thus, one typically needs to resort to some additional concepts and results in the study of Markov chains.

Definition 3.2.4 (Small set): *Given some positive, finite measure ν on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, we say that a set $S \in \mathcal{B}(\mathsf{X})$ is* (ν, m)-small *if there exists an $m \in \mathbb{N}$ such that*

$$p^m(u, A) \geq \nu(A), \quad \forall u \in S,\ A \in \mathcal{B}(\mathsf{X}). \tag{3.10}$$

We say that S is small *if* (3.10) *holds with $m = 1$.*

Notice that one can replace the right hand side of (3.10) by $\delta\hat{\nu}$, where $\delta := \int_{\mathsf{X}} \nu(\mathrm{d}u)$ and $\hat{\nu} = \nu/\delta$ is the probability measure induced by normalizing ν. The name "small set" is a bit of a misnomer; in practice, a small set S could be arbitrarily large (in fact, it could even be the whole space X for a

given type of chain). Some authors refer to them as *test sets*. A related concept is that of *petite set*, defined next.

Definition 3.2.5 (Petite set): *A set $C \subseteq \mathsf{X}$ is called* petite *(or $(n_0, \delta, \hat{\nu})$-petite), if there exists $\delta > 0$, $a \in (0,1)$, a probability measure $\hat{\nu}$ on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, and a positive integer n_0 such that*

$$(1-a)\sum_{n=1}^{n_0} a^n P^n(u,\cdot) \geq \delta\hat{\nu}(\cdot), \quad \forall u \in C.$$

Notice that petite sets then allow for the covering of the minorization condition in (3.10) by a combination of states (see, e.g., [113] for a thorough discussion on what this implies). We remark that a small set is always petite, however, the reverse is not always true, in general. An important result regarding irreducibility and petite sets is given next.

Theorem 3.2.2: *Given some Markov operator P, let $\tau_A^u := \inf\{n \geq 1 : u^n \in A\}$, where $\{u^n\}$ is an irreducible Markov chain with operator P starting at $u^0 = u$, denote the hitting time of the set $A \in \mathcal{B}(\mathsf{X})$ from the state u. The Markov chain generated by P is Harris recurrent if there exists some petite set $C \in \mathcal{B}(\mathsf{X})$ such that $\mathbb{P}(\tau_C^u < \infty) = 1, \forall u \in C$.*

Proof. See [113, Theorem 8.3.6]. □

Thus, instead of showing Harris recurrence of the chain directly, one typically looks for such a petite set C to which the chain always returns with probability 1.

Definition 3.2.6 (Aperiodicity): *A ψ-irreducible chain $\{u^n\}$ is called* aperiodic*, if there exists a small set S with $\psi(S) > 0$ and $\tilde{n} \in \mathbb{N}$ such that*

$$\inf_{u \in S} p^n(u, S) > 0, \quad \forall n \geq \tilde{n}.$$

Aperiodicity can be verified in light of the following result presented in [113].

Lemma 3.2.3: *Let $\{u^n\}_{n\in\mathbb{N}}$ be a ψ-irreducible Markov chain induced by a Markov transition kernel p. If there exists a $u \in \mathsf{X}$ such that $p(u, \{u\}) > 0$, then, the chain is aperiodic.*

A first result concerning the convergence of Markov chains, as presented by e.g., [141], is that if a Markov operator P is μ-invariant,ψ- irreducible, aperiodic, and Harris recurrent, it then follows that

$$\lim_{n\to\infty} \|p^n(u,\cdot) - \mu(\cdot)\|_{\mathcal{M}_1} = 0, \quad \forall u \in \mathsf{X}.$$

The previous result however, does not quantify the rate at which the Markov chain converges to its target probability measure. We will present the so-called regeneration construction to quantify such a rate. Given a small set S, notice that if one writes $p(u, \cdot)$ as

$$\begin{aligned} p(u,\cdot) &= \delta\hat{\nu}(\cdot) + p(u,\cdot) - \delta\hat{\nu}(\cdot) \\ &= \delta\hat{\nu}(\cdot) + (1-\delta)\underbrace{\frac{p(u,\cdot) - \delta\hat{\nu}(\cdot)}{1-\delta}}_{=:\hat{p}(u,\cdot)} \\ &= \delta\hat{\nu}(\cdot) + (1-\delta)\hat{p}(u,\cdot), \quad u \in S, \end{aligned} \tag{3.11}$$

where the minorization condition in (3.10) guarantees the positivity of $\hat{p}(u, \cdot)$, sampling from the probability measure $p(u, \cdot)$ can be understood as sampling from the mixture (3.11), i.e., with probability $(1 - \delta)$ one samples from the auxiliary kernel $\hat{p}(u, \cdot)$, and otherwise, one samples from $\hat{\nu}(\cdot)$ independently of the current state of the chain. By using arguments from renewal theory, [113, 120] one can show that the chain regenerates (broadly speaking, "forgets about the past") with probability δ, i.e., every time we sample from $\hat{\nu}$. To see this, we begin by defining the concept of *coupling*, together with the so-called *coupling inequality*.

Definition 3.2.7: *Given two measures μ, ν on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, we say that a probability measure Γ on $(\mathsf{X} \times \mathsf{X}, \mathcal{B}(\mathsf{X} \times \mathsf{X}))$ is a* coupling *of μ and ν if $(u, v) \sim \Gamma$, implies $u \sim \mu$ and $v \sim \nu$.*

It is known that whenever X is a Polish space (i.e., any separable, completely metrizable topological space), the following coupling inequality holds [99] for any coupling Γ:

$$\|\mu - \nu\|_{\mathsf{tv}} \leq \mathbb{P}_{\Gamma}(u \neq v), \quad (u, v) \sim \Gamma. \tag{3.12}$$

Now, let P be a μ-invariant, ψ-irreducible, and aperiodic Markov transition operator satisfying a minorization condition of the form (3.10). Furthermore, let S be a small set, define $\hat{S} := S \times S$, and consider two μ-invariant Markov chains $\{u^n, n \in \mathbb{N}\}$, $\{v^n, n \in \mathbb{N}\}$ generated by P with $u^0 \sim \nu$ and $v^0 \sim \mu$ (i.e., $\{v^n,\ n \in \mathbb{N}\}$ is started at stationarity). We generate a coupling Γ of the chains $\{u^n, n \in \mathbb{N}\}, \{v^n,\ n \in \mathbb{N}\}$ by using Algorithm 1.

Algorithm 1 Coupling Construction

procedure COUPLING CONSTRUCTION(μ,ν,P,$\hat{\nu}$,δ,$\hat{S}$).
  Sample $u^0 \sim \nu$ and $v^0 \sim \mu$
  for $n = 0, 1, \dots$ **do**
    if $u^n = v^n$ **then**
      Sample $u^{n+1} \sim p(u^n, \cdot)$ and set $v^{n+1} = u^{n+1}$
    else
      if $(u^n, v^n) \notin \hat{S}$ **then**
        Sample $u^{n+1} \sim p(u^n, \cdot)$, $v^{n+1} \sim p(v^n, \cdot)$ independently
      else
        with probability δ sample $u^{n+1} \sim \hat{\nu}$, and set $v^{n+1} = u^{n+1}$
        Otherwise, sample $u^{n+1} \sim \hat{p}(u^n, \cdot)$, $v^{n+1} \sim \hat{p}(u^n, \cdot)$, independently.
      end if
    end if
  end for
  Output $\{u^n, v^n\}_{n \in \mathbb{N}}$
end procedure

Furthermore, denoting by $T \in \mathbb{N}$ the random time at which coupling occurs, it then follows from the coupling inequality that

$$\left\| p^n(u^0, \cdot) - \mu(\cdot) \right\|_{\mathsf{tv}} \leq \mathbb{P}_\Gamma(v^n \neq u^n) \leq \mathbb{P}_\Gamma(T > n). \tag{3.13}$$

Under the additional (restrictive) assumption that $S = \mathsf{X}$, we have the first convergence theorem.

Theorem 3.2.3 (Small state space implies uniform ergodicity): *Let P be a μ-invariant, ψ-irreducible and aperiodic Markov operator satisfying a minorization condition of the form* (3.10) *with $S = \mathsf{X}$ and $n = 1$. Then, the Markov chain generated by P is $\mathcal{M}_1$-uniformly ergodic.*

Proof. Notice that since $S = \mathsf{X}$, at any given step, the chain can sample from $\hat{\nu}$ (and hence, couple) with probability δ. Thus, the random variable T follows a geometric(δ) distribution, and as such, $\mathbb{P}_\Gamma(T = n) = \delta(1-\delta)^{n-1}$, for which it follows that $\mathbb{P}_\Gamma(T > n) = (1-\delta)^n$. Thus, from the coupling inequality (3.13) it then follows that

$$\left\| p^n(u^0, \cdot) - \mu(\cdot) \right\|_{\mathcal{M}_1(\mathsf{X},\mu)} = 2 \left\| p^n(u^0, \cdot) - \mu(\cdot) \right\|_{\mathsf{tv}} \leq 2(1-\delta)^n.$$

□

Remark 3.2.1: *It is shown in [113, Theorem 16.0.2] that the converse of the previous theorem holds true as well, i.e., a Markov chain is uniformly ergodic if and only if the entire state space is a small set.*

However, this coupling argument can only be applied whenever a minorization condition of the form (3.10) holds, i.e., whenever the chain is at a current state u contained in a small set. In order to show convergence in the case where $S \subset \mathsf{X}$, the chain would then need to *drift* towards this small set. We now formalize this intuition.

Definition 3.2.8 (Drift condition): *A Markov chain induced by a Markov operator P is said to satisfy a* drift condition *if there exist a function $V : \mathsf{X} \to [1, \infty]$, a small set S and positive constants $\lambda \in (0, 1)$, $b \in \mathbb{R}_+$, such that the following holds:*

$$(PV)(u) \leq \lambda V(u) + b\mathbb{1}_{\{u \in S\}}, \quad u \in \mathsf{X}. \tag{3.14}$$

Here, the function $V : \mathsf{X} \to [1, \infty]$ is called a Lyapunov function*, and $\mathbb{1}_{\{u \in S\}}$ is the characteristic function of the set S.*

Notice that $(PV)(u)$ can be understood as $\mathbb{E}_{p(u,\cdot)}\left[V(u^{n+1})|u^n = u\right]$, where the expectation is taken with respect to the measure $p(u, \cdot)$. Defining $\Delta V(u) := (PV)(u) - V(u)$, it is easy to see then that $\Delta V(u) < 0$ whenever the chain is not in the small set, thus making the chain drift, on average, to the regions of X where $V(u)$ is small (i.e., to S). Furthermore, it does so in such a way that for points $u \in \mathsf{X}$ for which $V(u)$ is large, this drift is faster. Intuitively ([83]), this implies that once the chain leaves the small set S, it tends to return rather quickly to it. This motivates the following classical results.

Theorem 3.2.4 (Existence of an invariant measure, and convergence to it): *Let P be a ψ-irreducible and aperiodic Markov operator satisfying a drift condition as in Equation (3.14). Then, it holds that*

1. *There exists a unique invariant probability measure μ for P.*
2. *The chain generated by P is $\mathcal{M}_1$-geometrically ergodic.*
3. *$\mathcal{M}_1$-geometric ergodicity is equivalent to*

$$\|p^n(u, \cdot) - \mu(\cdot)\|_V \leq MV(u)\rho^n, \quad \forall n \geq 0, \mu\text{-a.e.} u \in \mathsf{X}, \tag{3.15}$$

where $\|\mu(\cdot)\|_V := \sup_{|f| \leq V} |\mu(f)|$, $M \in \mathbb{R}_+$, $\rho \in (0, 1)$.

Proof. This is a standard result in the Theory of Markov chains. See, e.g., [113, Theorem 15.0.1] □

It is easy to see that Markov chains for which the bound on the right hand side of (3.15) does not depend on u are *uniformly ergodic*. The following result demonstrates the reverse implication

Theorem 3.2.5 (Bounded Lyapunov function, [113]): *A Markov chain on a general state space X is uniformly ergodic if and only if it satisfies a drift condition of the form (3.14) with a bounded Lyapunov function $V : \mathsf{X} \to [1, V_{\max}]$, $V_{\max} \in \mathbb{R}_+$, and a small set S.*

Proof. See [113, Theorem 16.0.2, implication viii]. □

A further consequence of a drift condition is the existence of a central limit theorem, as shown in Theorem 17.0.1(ii-iv) of [113]

Theorem 3.2.6 (Drift condition and Harris recurrence imply a central limit theorem): *Let P be a μ-invariant, Harris recurrent Markov operator satisfying a drift condition of the form (3.14). Denote by $\{u^n,\ n \in \mathbb{N}\}$ the Markov chain obtained from such an operator with $u^0 \sim \mu^0$, let $f : \mathsf{X} \to \mathbb{R}$ be a function satisfying $f^2 \leq V$, where V is the Lyapunov function of the drift condition, and define $g(u) := f(u) - \int_{\mathsf{X}} f(u)\mu(\mathrm{d}u)$. Then, the constant*

$$\mathbb{V}_\mu[f] := \mathbb{E}_\mu[g^2(u^0)] + 2\sum_{n=1}^{\infty} \mathbb{E}_\mu\left[g(u^0)g(u^n)\right],$$

is well-defined, non-negative, and corresponds to the asymptotic variance

$$\lim_{N\to\infty} \frac{1}{N}\mathbb{E}_\mu\left[\left(\sum_{n=1}^{N} g(u^n)\right)^2\right] = \mathbb{V}_\mu[f].$$

Furthermore, if $\mathbb{V}_\mu[f] > 0$, then the central limit theorem holds for f, i.e.,

$$\sqrt{N}\left(\frac{1}{N}\sum_{n=1}^{N} f\left(u^n\right) - \mathbb{E}_\mu[f]\right) \xrightarrow{\mathcal{D}} \mathcal{N}\left(0, \mathbb{V}_\mu[f]\right), \quad \text{as } N \to \infty,$$

where we used the symbol "$\xrightarrow{\mathcal{D}}$" to denote convergence in distribution.

Proof. See Theorem 17.0.1 in [113]. □

In practice, the asymptotic variance $\mathbb{V}_\mu[f]$ is estimated using, e.g., *batched means* ([54, 57]), *window methods* ([57]), or with *regeneration arguments* as in [93, 92].

3.3 MSE bounds

3.3.1 Non-asymptotic bounds on the MSE: known results

Let $f : \mathsf{X} \to \mathbb{R}$ be a μ-integrable function, and denote by $\{u^n\}_{n=0}^{N+n_b}$ the (finite-length) Markov chain obtain from a geometrically ergodic, μ-invariant, Markov operator P, with $u^0 \sim \mu^0$. Furthermore, denote by $\hat{f}_{N,n_b} = \frac{1}{N}\sum_{n=1}^{N} f(u^{n+n_b})$ the ergodic estimator obtained from the Markov chain generated by P. We define the Mean Squared Error (MSE) of the chain as

$$\mathrm{MSE}(\hat{f}_{N,n_b}; \mu^0) := \mathbb{E}_{\mu^0,P}\left[\left(\hat{f}_{N,n_b} - \mathbb{E}_\mu[f]\right)^2\right], \tag{3.16}$$

where $\mathbb{E}_{\mu^0,P}[\cdot]$ denotes expectation with respect to the Markov chain started from an initial measure μ^0, and induced by the μ-invariant Markov operator P. It is of interest for practical applications to obtain (or at least, to quantify) non-asymptotic error bounds for the MSE (3.16). In the context of this thesis, such will be the case in Chapters 5 and 6, where verifying the cost-tolerance assumptions of ML-MCMC algorithms ([45, 108]) will require a non -asymptotic bound on the MSE of a given estimator in terms of its asymptotic variance (c.f. Assumption T3 in Chapter 5).

Results providing bounds on (3.16) are, to the best of the author's knowledge, rather scarce, with only a handful of results on this topic. In particular, under the assumption of a drift condition on P, with Lyapunov function V, the work [92] presents a bound on the form

$$\mathrm{MSE}(\hat{f}_{N,n_b};\mu^0) \leq \left(\sqrt{\frac{\mathbb{V}_\mu[f]}{N}}\left(1+\frac{c_0(P)}{N}\right)+\frac{c_1(P,f)}{N}\right)^2,$$

where the terms c_i, $i = 0, 1$, depend on the constant δ in the minorization condition (3.10), together with the constant λ and the Lyapunov function V in the drift condition (3.14). Under the same assumption of a drift condition, a similar bound is presented in [93, Theorem 3.1] of the form

$$\mathrm{MSE}(\hat{f}_{N,n_b};\mu^0) \leq N^{-1}\left(\sup_{u\in\mathsf{X}}\frac{|f-\mu(f)|^2}{V(u)}\right)\left(1+\frac{2B\rho}{1-\rho}\right)\left(\mu(V)+\frac{c_2(V,\mu^0,\mu)}{N(1-\rho)}\right).$$

for some $c_2(B, V, \mu, \mu^0) > 0$, with B, V, ρ once again as in (3.15).

The work [143], presents a non-assymptotic bound on the MSE for reversible chains.

Theorem 3.3.1: *Let $f \in L_2(\mathsf{X},\mu)$, be a μ-square integrable function and write $g(u) = f(u) - \int_{\mathsf{X}} f(u)\mu(\mathrm{d}u)$. Let P be a μ-reversible Markov operator and assume the chain generated by P starts from an initial probability measure $\mu^0 \ll \mu$, with $\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu} \in L_\infty(\mathsf{X},\mu)$. In addition, suppose that*

R1. *(L_2-spectral gap) there exists $b \in (0,1)$ such that*

$$\|P\|_{L_2^0(\mathsf{X},\mu)\to L_2^0(\mathsf{X},\mu)} \leq b,$$

R2. *(L_1-exponential convergence) there exists $\tilde{c} \in \mathbb{R}_+$, $a \in (0,1)$ such that*

$$\left\|\mu^0 P^n - \mu\right\|_{\mathcal{M}_1(\mathsf{X},\mu)} := \left\|\frac{\mathrm{d}(\mu^0 P^n)}{\mathrm{d}\mu} - 1\right\|_{L_1(\mathsf{X},\mu)} \leq \tilde{c}a^n,$$

Then, the non-asymptotic MSE is given by

$$\mathsf{E}_{\mu^0,P}\left|\frac{1}{N}\sum_{n=1}^{N} g(u^{n+n_b})\right|^2 \leq \frac{\mathbb{V}_\mu[u]}{N}\left(\frac{2}{(1-b)} + \frac{2\tilde{c}\left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu}-1\right\|_{L_\infty} a^{n_b}}{N(1-a)^2}\right), \tag{3.17}$$

where the first term in the parenthesis is associated with the variance contribution to the MSE, whereas the second term corresponds to the statistical squared bias and is of higher order in N.

Remark 3.3.1: *The additional assumptions R1 and R2 in Theorem 3.3.1 are satisfied for a geometrically ergodic, μ-reversible Markov operator P.*

3.3.2 Non-asymptotic bounds on the MSE: new result for non-reversible chains

Let $P : L_2(\mathsf{X},\mu) \mapsto L_2(\mathsf{X},\mu)$ be a μ-invariant Markov operator for some probability measure μ on $(\mathsf{X},\mathcal{B}(\mathsf{X}))$, and recall the *averaging operator* $\widehat{\mu}f : L_2(\mathsf{X},\mu) \to L_2(\mathsf{X},\mu)$, $f \mapsto \widehat{\mu}(f) = \int_{\mathsf{X}} f(u)\mu(\mathrm{d}u)$. We now present a bound similar to that in (3.17). This bound generalizes that of Theorem 3.3.1 to the case where R1 and R2 do not necessarily hold (e.g., whenever the Markov chain is not reversible), using the pseudo-spectral gap. This bound is an original contribution, first presented in our work [108].

Theorem 3.3.2 (Non-asymptotic bound on the mean square error): *Let $f \in L_2(\mathsf{X},\mu)$, be a μ-square integrable function and write $g(u) = f(u) - \int_{\mathsf{X}} f(u)\mu(\mathrm{d}u)$. Let P be a μ-invariant (but not necessarily μ-reversible) Markov operator with $\gamma_{\mathrm{ps}}[P] > 0$, and assume the chain generated by P starts from an initial probability measure $\mu^0 \ll \mu$, with $\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu} \in L_\infty(\mathsf{X},\mu)$. Then,*

$$\mathrm{MSE}(\hat{f}_{N,n_b};\mu^0) = \mathsf{E}_{\mu^0,P}\left|\frac{1}{N}\sum_{n=1}^{N} g(u^{n+n_b})\right|^2 \leq \frac{\mathbb{V}_\mu[f]}{N}(C_{\mathrm{inv}} + C_{\mathrm{ns}}), \tag{3.18}$$

where $C_{\mathrm{inv}} = \left(1 + \frac{4}{\gamma_{\mathrm{ps}}[P]}\right)$, $C_{\mathrm{ns}} = \left(2\left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu}-1\right\|_{L_\infty}\left(1+\frac{4}{\gamma_{\mathrm{ps}}[P]}\right)\right)$, where $\gamma_{\mathrm{ps}}[P]$ is the pseudo-spectral gap of P, defined in (3.4).

The proof of Theorem 3.3.2 is decomposed into a series of auxiliary results.
We present a first bound of the form (3.18) for chains which are started at stationarity (i.e., whenever $\mu^0 = \mu$). Although this is usually not the case, the following Lemma is useful in the proof of Theorem 3.3.2.

Lemma 3.3.1 (MSE bound starting at stationarity): *Under the same assumptions as in* Theorem 3.3.2 *and with* $\mu^0 = \mu$, *it holds*

$$\mathrm{MSE}(\hat{f}_{N,n_b=0};\mu) := \mathsf{E}_{\mu,P}\left|\frac{1}{N}\sum_{n=1}^{N} g(u^n)\right|^2 \leq \frac{\mathbb{V}_\mu[f]}{N}\left(1+\frac{4}{\gamma_{\mathrm{ps}}[P]}\right).$$

Proof. We follow a similar approach to those presented in [127, Theorem 3.2] and [142, Section 3]. To ease notation, for the remainder of this proof we write $L_q = L_q(\mathsf{X},\mu),\ q\in[1,\infty]$. We can write the MSE of a Markov chain generated by P starting at μ as

$$\mathsf{E}_{\mu,P}\left|\frac{1}{N}\sum_{n=1}^{N} g(u^n)\right|^2 = \frac{1}{N^2}\sum_{n=1}^{N}\mathsf{E}_{\mu,P}[g(u^n)^2] + \frac{2}{N^2}\sum_{j=1}^{N-1}\sum_{i=j+1}^{N}\mathsf{E}_{\mu,P}[g(u^i)g(u^j)]. \quad (3.19)$$

Working on the expectation of the second term on the right hand side we get from the Cauchy-Schwarz inequality that

$$\begin{aligned}\mathsf{E}_{\mu,P}[g(u^i)g(u^j)] &= \langle g, P^{i-j}g\rangle_\mu = \langle g, (P-\widehat{\mu})^{i-j}g\rangle_\mu \\ &\leq \|g\|_{L_2}^2 \left\|(P-\widehat{\mu})^{i-j}\right\|_{L_2\mapsto L_2}.\end{aligned}$$

Notice that for any $k\geq 1$, we have

$$\begin{aligned}\left\|(P-\widehat{\mu})^{i-j}\right\|_{L_2\mapsto L_2} &\leq \left\|(P-\widehat{\mu})^{k}\right\|_{L_2\mapsto L_2}^{\lfloor\frac{i-j}{k}\rfloor} \\ &= \left\|(P^*-\widehat{\mu})^{k}(P-\widehat{\mu})^{k}\right\|_{L_2\mapsto L_2}^{\frac{1}{2}\lfloor\frac{i-j}{k}\rfloor} \end{aligned} \quad (3.20)$$

where $\lfloor\cdot\rfloor$ is the floor function. Now, let k_{ps} be the smallest integer such that

$$k_{\mathrm{ps}}\gamma_{\mathrm{ps}}[P] = \gamma[(P^*)^{k_{\mathrm{ps}}}P^{k_{\mathrm{ps}}}] = 1 - \left\|(P^*-\widehat{\mu})^{k_{\mathrm{ps}}}(P-\widehat{\mu})^{k_{\mathrm{ps}}}\right\|_{L_2\mapsto L_2}, \quad (3.21)$$

which is strictly positive for uniformly ergodic chains (see [139, Section 3.3]). Then, from (3.19), (3.20), and (3.21), we obtain the following:

$$\frac{2}{N^2}\sum_{j=1}^{N-1}\sum_{i=j+1}^{N}\mathsf{E}_{\mu,P}[g(u^i)g(u^j)] \leq \frac{2}{N^2}\sum_{j=1}^{N-1}\sum_{i=j+1}^{N}\|g\|_{L_2}^2\left(1-k_{\mathrm{ps}}\gamma_{\mathrm{ps}}[P]\right)^{\frac{1}{2}\lfloor\frac{i-j}{k_{\mathrm{ps}}}\rfloor}.$$

For notational simplicity we write $\varrho = (1 - k_{\text{ps}}\gamma_{\text{ps}}[P])$. We then have the following:

$$\begin{aligned}\frac{2}{N^2}\sum_{j=1}^{N-1}\sum_{i=j+1}^{N}\|g\|_{L_2}^2\,\varrho^{\frac{1}{2}\lfloor\frac{i-j}{k_{\text{ps}}}\rfloor} &\leq \frac{2\,\|g\|_{L_2}^2}{N}\sum_{m=0}^{\infty}\varrho^{\frac{1}{2}\lfloor\frac{m}{k_{\text{ps}}}\rfloor} \leq \frac{2\,\|g\|_{L_2}^2\,k_{\text{ps}}}{N}\sum_{m=0}^{\infty}\varrho^{\frac{1}{2}m} \\ &= \frac{2\,\|g\|_{L_2}^2\,k_{\text{ps}}}{N}\frac{1}{1-\varrho^{1/2}} = \frac{2\,\|g\|_{L_2}^2\,k_{\text{ps}}}{N}\frac{1+\varrho^{\frac{1}{2}}}{1-\varrho} \\ &\leq \frac{4\,\|g\|_{L_2}^2}{N\gamma_{\text{ps}}[P]},\end{aligned}$$

where the second inequality comes from the definition of the floor function $\lfloor\cdot\rfloor$.
We shift our attention to the first term in (3.19). Using Hölder's inequality with $q = \infty,\ q' = 1$, and the fact that P is a weak contraction in $L_q(\mathsf{X}, \mu)$, for any $q \in [1, \infty]$, we obtain the following:

$$\frac{1}{N^2}\sum_{n=1}^{N}\mathsf{E}_{\mu,P}[g(u^n)^2] = \frac{1}{N^2}\sum_{n=1}^{N}\langle 1, P^n g^2\rangle_\mu \leq \frac{1}{N^2}\sum_{n=1}^{N}\left\|P^n g^2\right\|_{L_1} \leq \frac{\left\|g^2\right\|_{L_1}}{N}.$$

Lastly,

$$\left\|g^2\right\|_{L_1} = \int_{\mathsf{X}}|g^2(u)|\mu(\mathrm{d}u) = \int_{\mathsf{X}}g^2(u)\mu(\mathrm{d}u) = \|g\|_{L_2}^2.$$

Hence, we obtain the following bound:

$$\frac{1}{N^2}\sum_{n=1}^{N}\mathsf{E}_{\mu,P}[g(u^n)^2] \leq \frac{\|g\|_{L_2}^2}{N}. \tag{3.22}$$

Thus, from (3.19) and (3.22), with the observation that $\|g\|_{L_2}^2 = \int_{\mathsf{X}}(f(u) - \widehat{\mu}(f))^2\mu(\mathrm{d}u) = \mathbb{V}_\mu[f]$, we finally obtain,

$$\begin{aligned}\mathsf{E}_{\mu,P}\left|\frac{1}{N}\sum_{n=1}^{N}g(u^n)\right|^2 &\leq \frac{\|g\|_{L_2}^2}{N}\left(1 + \frac{4}{\gamma_{\text{ps}}[P]}\right) \\ &= \frac{\mathbb{V}_\mu[f]}{N}\left(1 + \frac{4}{\gamma_{\text{ps}}[P]}\right).\end{aligned}$$

□

The previous result should be compared to [127, Theorem 3.2] and [142, Theorem 5]. We consider the more general case where the chain is not started from stationarity, i.e., when $u^0 \sim \mu^0$, where $\mu^0 \ll \mu$ is a probability measure on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$. We recall the following result from [143].

Lemma 3.3.2: *Denote by $n_b \in \mathbb{N}$ the burn-in period and let $\{u^n\}_{n\in\mathbb{N}}$ be a Markov chain generated by P starting from an initial measure μ^0 and invariant probability measure μ, with $\mu^0 \ll \mu$. Under the same assumptions as in Theorem 3.3.2, it holds that:*

$$\mathsf{E}_{\mu^0,P}\left|\frac{1}{N}\sum_{n=1}^{N} g(u^{n+n_b})\right|^2 = \mathsf{E}_{\mu,P}\left|\frac{1}{N}\sum_{n=1}^{N} g(u^n)\right|^2 + \frac{1}{N^2}\sum_{j=1}^{N}\mathcal{H}^{j+n_b}(g^2) + \frac{2}{N^2}\sum_{j=1}^{N-1}\sum_{k=j+1}^{N}\mathcal{H}^{j+n_b}(gP^{k-j}g), \tag{3.23}$$

where

$$\mathcal{H}^i(h) = \left\langle (P^i - \widehat{\mu})h, \left(\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu} - 1\right)\right\rangle_\mu, \quad i \in \mathbb{N},\ h \in L_2(\mathsf{X},\mu),$$

Proof. See [143, Proposition 3.29]. □

We can now prove Theorem 3.3.2.

Proof of Theorem 3.3.2. Once again, for the remainder of this proof we write $L_q = L_q(\mathsf{X},\mu)$, $q \in [1,\infty]$. From Lemma 3.3.2 we get

$$\mathcal{H}^{j+n_b}(g^2) = \left\langle (P^{j+n_b} - \widehat{\mu})g^2, \left(\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu} - 1\right)\right\rangle_\mu, \tag{3.24}$$

$$\mathcal{H}^{j+n_b}(gP^{k-j}g) = \left\langle (P^{j+n_b} - \widehat{\mu})(gP^{k-j}g), \left(\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu} - 1\right)\right\rangle_\mu. \tag{3.25}$$

Using Hölder's inequality with $q' = \infty$, $q = 1$ on the right hand side of (3.24) gives

$$\begin{aligned}\mathcal{H}^{j+n_b}(g^2) &\le \left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu} - 1\right\|_{L_\infty} \left\|(P^{j+n_b} - \widehat{\mu})g^2\right\|_{L_1} \\ &\le \left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu} - 1\right\|_{L_\infty} \left\|(P^{j+n_b} - \widehat{\mu})\right\|_{L_1\mapsto L_1} \left\|g^2\right\|_{L_1},\end{aligned}$$

where the last inequality comes from the definition of operator norm. Moreover, since the Markov operators are weak contractions, we have that $\left\|(P^{j+n_b} - \widehat{\mu})\right\|_{L_1\mapsto L_1} \le 2, \forall j \in \mathbb{N}$, which gives the bound

$$\mathcal{H}^{j+n_b}(g^2) \le 2\left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu} - 1\right\|_{L_\infty} \|g\|_{L_2}^2.$$

Summing over j results in

$$\frac{1}{N^2}\sum_{j=1}^{N}\mathcal{H}^{j+n_b}(g^2) \leq \frac{2\left\|g\right\|_{L_2}^2}{N}\left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu}-1\right\|_{L_\infty}. \tag{3.26}$$

Following similar procedure for (3.25) we obtain

$$\mathcal{H}^{j+n_b}(gP^{k-j}g) \leq 2\left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu}-1\right\|_{L_\infty}\left\|g(P^{k-j})g\right\|_{L_1}.$$

Furthermore, from Hölder's inequality (with $q'=q=2$) and the fact that $\widehat{\mu}(g)=0$,

$$\begin{aligned}\left\|g(P^{k-j})g\right\|_{L_1} &\leq \left\|g\right\|_{L_2}\left\|P^{k-j}g\right\|_{L_2} = \left\|g\right\|_{L_2}\left\|(P-\widehat{\mu})^{k-j}g\right\|_{L_2}\\ &\leq \left\|g\right\|_{L_2}^2\left\|(P-\widehat{\mu})^{k-j}\right\|_{L_2\mapsto L_2} \leq \left\|g\right\|_{L_2}^2\left(1-k_{\mathrm{ps}}\gamma_{\mathrm{ps}}[P]\right)^{\frac{1}{2}\lfloor\frac{k-j}{k_{\mathrm{ps}}}\rfloor},\end{aligned}$$

where the last inequality follows from the same pseudo-spectral gap argument used in the proof of Lemma 3.3.1. Adding over j and k produces

$$\frac{2}{N^2}\sum_{j=1}^{N-1}\sum_{k=j+1}^{N}\mathcal{H}^{j+n_b}(gP^{k-j}g) \leq \frac{8\left\|g\right\|_{L_2}^2}{N\gamma_{\mathrm{ps}}}\left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu}-1\right\|_{L_\infty}. \tag{3.27}$$

Notice then that Equations (3.26) and (3.27), provide a bound on the second and third term in Lemma 3.3.2. Lastly, combining these results with Lemma 3.3.1 and once again observing that $\left\|g\right\|_{L_2}^2 = \mathbb{V}_\mu[f]$, provides the desired result

$$(3.23) \leq \frac{\mathbb{V}_\mu[f]}{N}\left(1+\frac{2}{\gamma_{\mathrm{ps}}[P]}\right)+\frac{\mathbb{V}_\mu[f]}{N}\left(2\left\|\frac{\mathrm{d}\mu^0}{\mathrm{d}\mu}-1\right\|_{L_\infty}\left(1+\frac{4}{\gamma_{\mathrm{ps}}[P]}\right)\right).$$

□

3.4 Review of common techniques and algorithms

Perhaps the best known MCMC technique is the *Metropolis-Hastings* (MH) algorithm [68, 112]. Loosely speaking, this algorithm constructs a Markov chain by iteratively proposing a (possibly state-dependent) candidate state, and accepting or rejecting it as the new state of the chain, in such a way that the resulting Markov transition kernel associated to this process is invariant with respect to a desired probability measure. More formally, given a state $u \in \mathsf{X}$, a target probability measure μ, and an auxiliary Markov transition kernel $Q(u,\cdot)$, denote by $h(\mathrm{d}u,\mathrm{d}v) = Q(u,\mathrm{d}v)\mu(\mathrm{d}u)$.

Furthermore, defining $h^{\mathsf{T}}(\mathrm{d}u, \mathrm{d}v)$ as $h(\mathrm{d}v, \mathrm{d}u)$ and assuming that $h^{\mathsf{T}}(\mathrm{d}u, \mathrm{d}v) \ll h(\mathrm{d}u, \mathrm{d}v)$, one can define the *Metropolis-Hastings acceptance probability* as

$$\alpha(u, v) = \min\left\{1, \frac{\mathrm{d}h^{\mathsf{T}}}{\mathrm{d}h}(u, v)\right\}. \tag{3.28}$$

Here, $\alpha(u, v)$ corresponds to the probability of accepting a proposed state v (potentially depending on the current state u), given that the current state of the chain is u. As it is often the case, the absolute continuity of h^{T} with respect to h is relatively straightforward to show in the case where X is a finite-dimensional space. One needs to be more careful to show this absolute continuity whenever X is an infinite-dimensional space. The following slight extension of [160, Theorem 2] (c.f. Remark 3.4.1) provides a way of constructing such measures.

Lemma 3.4.1 (Extension of Theorem 1 in [160]): *Consider a Metropolis-Hastings algorithm with target measure $\mu(\mathrm{d}u)$ and proposal kernel $Q(u, \mathrm{d}v)$. Assume there exists a reference measure $\nu(\mathrm{d}u)$ and a reference kernel $Q_{ref}(u, \mathrm{d}v)$, such that $\mu \ll \nu$ and $Q(u, \mathrm{d}v)\mu(\mathrm{d}u) = h(\mathrm{d}u, \mathrm{d}v) \ll h_{ref}(\mathrm{d}u, \mathrm{d}v) := Q_{ref}(u, \mathrm{d}v)\nu(\mathrm{d}u)$. Then, there exists a $\mathcal{B}(\mathsf{X})$-measurable function $f : \mathsf{X} \to \mathbb{R}_+$ such that $\mu(\mathrm{d}u) = f(u)\nu(\mathrm{d}v)$ and a $\mathcal{B}(\mathsf{X}) \otimes \mathcal{B}(\mathsf{X})$-measurable function $\tilde{g} : \mathsf{X} \times \mathsf{X} \to \mathbb{R}_+$ such that $\frac{\mathrm{d}h}{\mathrm{d}h_{ref}}(u, v) = \tilde{g}(u, v)$. Furthermore, if in addition it holds that*

1. *f and $\tilde{g}$ are positive h_{ref}—a.s., and*
2. *Q_{ref} is ν-reversible as in (3.6) i.e., $\nu(\mathrm{d}u)Q_{ref}(u, \mathrm{d}v) = \nu(\mathrm{d}v)Q_{ref}(v, \mathrm{d}u)$,*

then, there exists a $\mathcal{B}(\mathsf{X}) \otimes \mathcal{B}(\mathsf{X})$-measurable function $g : \mathsf{X} \times \mathsf{X} \to \mathbb{R}_+$ such that $\frac{\mathrm{d}Q(u,\cdot)}{\mathrm{d}Q_{ref}(u,\cdot)}(v) = g(u, v)$. Furthermore, $h^{\mathsf{T}}(\mathrm{d}u, \mathrm{d}v) \ll h(\mathrm{d}u, \mathrm{d}v)$, with

$$\frac{\mathrm{d}h^{\mathsf{T}}}{\mathrm{d}h}(u, v) = \frac{f(v)}{f(u)}\frac{g(v, u)}{g(u, v)}.$$

Thus, the Metropolis-Hastings acceptance probability of the form (3.28) is well-defined.

Proof. On the one hand, since $h(\mathrm{d}u, \mathrm{d}v) \ll h_{\mathrm{ref}}(\mathrm{d}u, \mathrm{d}v)$, one has that

$$f(u)\nu(\mathrm{d}u)Q(u, \mathrm{d}v) = \mu(\mathrm{d}u)Q(u, \mathrm{d}v) = \tilde{g}(u, v)\nu(\mathrm{d}u)Q_{\mathrm{ref}}(u, \mathrm{d}v).$$

Let $A := \{(u, v) \in \mathsf{X}^2 : f(u) = 0\}$, which satisfies $h_{\mathrm{ref}}(A) = 0$. Then, setting $g(u, v) := \tilde{g}(u, v)/f(u)$, we have that

$$\text{and} \qquad \begin{aligned} Q(u, \mathrm{d}v) &= g(u, v)Q_{\mathrm{ref}}(u, \mathrm{d}v), \\ \frac{\mathrm{d}Q(u, \cdot)}{\mathrm{d}Q_{\mathrm{ref}}(u, \cdot)} &= g(u, v), \quad h_{\mathrm{ref}}\text{-a.e.} \end{aligned}$$

On the other hand, since Q_{ref} is ν-reversible, one then has that $\nu(\mathrm{d}u)Q_{\text{ref}}(u,\mathrm{d}v) = \nu(\mathrm{d}v)Q_{\text{ref}}(v,\mathrm{d}u)$. Multiplying both sides of this equation by $f(u)f(v)g(u,v)g(v,u)$ then gives:

$$f(v)g(v,u)\left(f(u)\nu(\mathrm{d}u)g(u,v)Q_{\text{ref}}(u,\mathrm{d}v)\right) = \left(f(v)\nu(\mathrm{d}v)g(v,u)Q_{\text{ref}}(v,\mathrm{d}u)\right)f(u)g(u,v).$$

Since $\mu(\mathrm{d}u) = f(u)\nu(\mathrm{d}u)$ and $Q(u,\mathrm{d}v) = g(u,v)Q_{\text{ref}}(u,\mathrm{d}v)$, one then obtains:

$$f(v)g(v,u)\mu(\mathrm{d}u)Q(u,\mathrm{d}v) = f(u)g(u,v)\mu(\mathrm{d}v)Q(v,\mathrm{d}u).$$

Furthermore, recognizing that $h(\mathrm{d}u,\mathrm{d}v) = \mu(\mathrm{d}u)Q(u,\mathrm{d}v)$, and since f and g are positive h_{ref}−a.e., it then follows that

$$\begin{aligned} h^{\mathsf{T}}(\mathrm{d}u,\mathrm{d}v) = \mu(\mathrm{d}v)Q(v,\mathrm{d}u) &= \frac{f(v)g(v,u)}{f(u)g(u,v)}\mu(\mathrm{d}u)Q(u,\mathrm{d}v) \\ &= \frac{f(v)g(v,u)}{f(u)g(u,v)}h(\mathrm{d}u,\mathrm{d}v), \quad h_{\text{ref}}\text{−a.e.,} \end{aligned}$$

which implies $h^{\mathsf{T}}(\mathrm{d}u,\mathrm{d}v) \ll h(\mathrm{d}u,\mathrm{d}v)$. □

Remark 3.4.1: *This previous result has also appeared in [40], where it is used to theoretically justify the preconditioned Crank-Nicholson algorithm. Our (slight) extension from those in [160] and [40] come from the fact that we state it for arbitrary ν and (ν-reversible) Q_{ref} (provided $\mu \ll \nu$ and $Q(u,\cdot) \ll Q_{ref}(u,\cdot)$ hold); while those works present such a result in terms of the prior and proposal kernel, respectively (i.e., for $\nu = \mu_{\mathsf{pr}}$, $Q_{ref}(u,\cdot) = Q(u,\cdot)$).*

Given an initial state $u^0 \sim \mu^0$, one can define the MH algorithm as in Algorithm 2.

Algorithm 2 Metropolis-Hastings

1: **procedure** METROPOLIS-HASTINGS(N,μ,Q,λ^0).
2: Sample $u^0 \sim \lambda^0$
3: **for** $n = 0,\ldots,N-1$ **do**
4: Sample $v \sim Q(u^n,\cdot)$.
5: Set $u^{n+1} = v$ with probability $\alpha(u^n,v)$ given by (3.28). Set $u^{n+1} = u^n$ otherwise.
6: **end for**
7: Output $\{u^n\}_{n=0}^N$
8: **end procedure**

Step 5 in the previous algorithm is commonly known as the *Metropolization step*. Given a state $u \in \mathsf{X}$, Algorithm 2 induces a Markov transition kernel of the form

$$p(u,A) = \int_A \alpha(u,v)Q(u,\mathrm{d}v) + \delta_u(A)\int_{\mathsf{X}}(1-\alpha(u,v))Q(u,\mathrm{d}v), \quad A \in \mathcal{B}(\mathsf{X}).$$

Remark 3.4.2 (Aperiodicity of Metropolized Algorithms): *It follows from Lemma 3.2.3 that Metropolized MCMC algorithms (i.e., those including an acceptance-rejection step, c.f. Section 3.4) are aperiodic if* $\alpha^*(u) = \int_{\mathsf{X}}(1 - \alpha(u, v))Q(u, \mathrm{d}v) > 0, \forall u \in \mathsf{X}$.

3.4.1 Construction of Q

A proper choice of Q is critical for the performance of the MH algorithm. We now present some common choices in the MCMC literature [3, 13, 129, 144], that we will justify as applications of Lemma 3.4.1. Throughout the rest of this section we will focus specifically to the setting of BIPs and will take $\mu(\mathrm{d}u) = \mu^y(\mathrm{d}u) \propto \exp(-\Phi(u; y))\mu_{\mathsf{pr}}(\mathrm{d}u)$.

Independent Metropolis Hastings (IMH)

The main idea behind this method is to choose a "transition kernel" $Q(u, \mathrm{d}v)$ which is independent of u, i.e., $Q(u, \mathrm{d}v) = Q(\mathrm{d}v)$ (i.e., $Q(\mathrm{d}v)$ is perhaps better understood as "just" a probability measure independent of the current state of the chain u). This method is attractive in the sense that, intuitively (and rather loosely speaking) one would expect that if one uses a proposal $Q(\mathrm{d}u)$ that well-approximates $\mu(\mathrm{d}u)$, then the algorithm would be quite efficient. In general however, it is, of course, not a trivial task to obtain such a measure Q. We present some common choices for it.

Prior-based IMH:

A first, natural, candidate for $Q(\mathrm{d}u)$ in the context of a BIP is the prior measure $\mu_{\mathsf{pr}}(\mathrm{d}u)$. We set $Q(\mathrm{d}v) = Q_{\mathrm{ref}}(\mathrm{d}v) = \mu_{\mathsf{pr}}(\mathrm{d}v)$, for which the assumptions of Lemma 3.4.1 are (trivially) satisfied, and as such one obtains:

$$\frac{\mathrm{d}h^{\mathsf{T}}}{\mathrm{d}h}(u, v) = \exp(\Phi(u; y) - \Phi(v; y)).$$

Laplace approximation -based IMH:

Suppose that $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$, for some self-adjoint, positive definite and trace-class operator $\mathcal{C}$. Suppose furthermore that one has constructed a Gaussian measure $Q(\mathrm{d}v) = \mathcal{N}(m, \mathcal{H})(\mathrm{d}v)$ approximating the posterior measure μ^y by using e.g., the optimization methods presented in Section 2.3.2, in such a way that $Q(\mathrm{d}v) \ll \mu_{\mathsf{pr}}(\mathrm{d}v)$ with $Q(\mathrm{d}v) \propto \exp(-\psi(v; \theta))\mu_{\mathsf{pr}}(\mathrm{d}v)$ (where $\theta = (m, \mathcal{H})$, with $m \in \mathsf{X}$ and $\mathcal{H}$ a self-adjoint, positive-definite, trace-class operator and ψ is a quadratic function). Taking once again the prior as a reference probability measure for both the posterior and the proposal (i.e., $\nu = Q_{\mathrm{ref}} = \mu_{\mathsf{pr}}$) one then has that

$$\frac{\mathrm{d}h^{\mathsf{T}}}{\mathrm{d}h}(u, v) = \frac{\exp(-\Phi(v; y) + \psi(v; \theta))}{\exp(-\Phi(u; y) + \psi(u; \theta))}.$$

Theorem 3.4.1 (Convergence of IMH [3]): *The Metropolis-Hastings chain obtained using the IMH algorithm is uniformly exponentially ergodic if and only if* $\sup_{u\in\mathsf{X}} \frac{f(u)}{g(u)} < \infty$, *with* g *and* f *defined as in Lemma 3.4.1. Otherwise the algorithm fails to be exponentially ergodic in the sense of Definition 3.2.1.*

Remark 3.4.3: *In the case where* $\sup_{u\in\mathsf{X}} \frac{f(u)}{g(u)} = +\infty$, *The previous Theorem does not necessarily preclude the IMH algorithm to have* slower *types of convergence, such as polynomial ergodicity.*

This type of sampler is at the core of multi-level MCMC techniques, such as the one presented in [45], and the one discussed in Chapter 5. We reiterate, however, that is not always easy to find or construct efficient independent kernels $Q(\mathrm{d}u)$. We now proceed to discuss a family of widely used methods that allow for extra flexibility.

Diffusion-based methods

Let $(\mathsf{X}, \langle\cdot,\cdot\rangle_{\mathsf{X}})$ be a separable Hilbert space with Borel σ-algebra $\mathcal{B}(\mathsf{X})$, let $\Phi(\cdot; y) : \mathsf{X} \to \mathbb{R}$ be a Fréchet-differentiable function, and denote by $D\Phi$ its (Fréchet) derivative. One way of constructing proposal kernels is based on discretizing the following over-damped Langevin SDE :

$$\mathrm{d}u_t = -b\mathcal{C}\left(\mathcal{K}^{-1}u + aD\Phi(u; y)\right)\mathrm{d}t + \sqrt{2}\mathcal{C}^{1/2}\mathrm{d}W_t, \quad a, b \in \{0, 1\}, \tag{3.29}$$

where $\mathcal{K}, \mathcal{C} : \mathsf{X} \to \mathsf{X}$ are self-adjoint, positive-definite and trace-class covariance operators, and W_t is a cylindrical Wiener process. We now investigate how one can use (3.29) to construct transition kernels for the MH algorithm.

Random Walk Metropolis:

As a first case, let X be a finite-dimensional space and set $b = a = 0$ in (3.29). A simple Euler-Maruyama (EM) discretization of (3.29) gives:

$$\begin{aligned} &\frac{u_{n+1} - u_n}{\tau} = \sqrt{\frac{2}{\tau}}\xi, \quad \xi \sim \mathcal{N}(0, \mathcal{C}),\ n = 1, 2, \dots. \\ &\implies u_{n+1} = u_n + \xi, \quad \xi \sim \mathcal{N}(0, 2\tau\mathcal{C}), \end{aligned}$$

where $\tau > 0$ denotes the discretization step of the EM scheme. Clearly, this induces the kernel $Q(u, \mathrm{d}v) = \mathcal{N}(u, 2\tau\mathcal{C})$. When used as a proposal for the MH algorithm, this results in the well-known *Random Walk Metropolis* algorithm, where the proposal for the $(n+1)^{\text{th}}$ step of the MH algorithm is a Gaussian centered at the current state u_n. This method is arguably one of the simplest and most common variants of MH. With a slight abuse of notation, we write $\mu^y(\cdot)$ (resp.

$Q(u^n, \cdot)$) as the Lebesgue density of the posterior (resp. proposal). By symmetry of the Gaussian density, we then obtain

$$\alpha(u, v) = \min\left\{1, \frac{\mu^y(v)Q(v,u)}{\mu^y(u)Q(u,v)}\right\} = \min\left\{1, \frac{\mu^y(v)}{\mu^y(u)}\right\}.$$

It is known that, for the finite-dimensional setting, the RWM algorithm is geometrically ergodic under relatively mild assumptions [3], however, it is shown in [65] that its L_2-spectral gap decays to 0 as the dimensionality of the state-space grows. In the case where $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$, this dependence of the dimensionality can be avoided by a small modification of such an algorithm, which we present next.

Preconditioned Crank-Nicolson [32, 130, 144, 153]

Alternatively, setting $b = 1, a = 0$ and $\mathcal{K} = \mathcal{C}$, a Crank-Nicolson discretization of (3.29) using a time step $0 < \tau < 2$ gives

$$\frac{u_{n+1} - u_n}{\tau} = -\left(\frac{u_{n+1} + u_n}{2}\right) + \sqrt{\frac{2}{\tau}}\xi, \quad \xi \sim \mathcal{N}(0, \mathcal{C}),\ n = 1, 2, \ldots,$$

which gives

$$\begin{aligned} u_{n+1} &= \frac{2-\tau}{2+\tau}u_n + \frac{\sqrt{8\tau}}{2+\tau}\xi, \quad \xi \sim \mathcal{N}(0, \mathcal{C}),\ n = 1, 2, \ldots \\ &= \sqrt{1-\rho^2}u_n + \rho\xi, \quad \xi \sim \mathcal{N}(0, \mathcal{C}),\ n = 1, 2, \ldots \end{aligned}$$

which clearly induces the kernel $Q(u_n, \cdot) = \mathcal{N}(\sqrt{1-\rho^2}u_n, \rho^2\mathcal{C})$. Assuming that $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$, it can be shown ([153]) that such a kernel is μ_{pr}-reversible; to that end, we define $h_0(\mathrm{d}u, \mathrm{d}v) := Q(u, \mathrm{d}v)\mu_{\mathsf{pr}}(\mathrm{d}u)$, as the process of sampling $u \sim \mu_{\mathsf{pr}}$ and $v|u \sim Q(u, \cdot)$, write $h_0^{\mathsf{T}}(\mathrm{d}u, \mathrm{d}v) = h_0(\mathrm{d}v, \mathrm{d}u)$ and follow the same procedure as [153, Chapter 5]. Sampling $u \sim \mu_{\mathsf{pr}}, z \sim \mu_{\mathsf{pr}}$ independently gives

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u \\ \sqrt{1-\rho^2}u + \rho z \end{pmatrix} = \begin{pmatrix} I & 0 \\ \sqrt{1-\rho^2}I & \rho I \end{pmatrix}\begin{pmatrix} u \\ z \end{pmatrix} \sim h_0(\mathrm{d}u, \mathrm{d}v),$$

which in turn implies (see, e.g., [153, Proposition 2.20]) that

$$h_0(\mathrm{d}u, \mathrm{d}v) = \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \mathcal{C} & \sqrt{1-\rho^2}\mathcal{C} \\ \sqrt{1-\rho^2}\mathcal{C} & \mathcal{C} \end{pmatrix}\right).$$

Similarly, one has that

$$\begin{pmatrix} v \\ u \end{pmatrix} = \begin{pmatrix} \sqrt{1-\rho^2} I & \rho I \\ I & 0 \end{pmatrix} \begin{pmatrix} u \\ z \end{pmatrix} \sim h_0^{\intercal}(\mathrm{d}u, \mathrm{d}v) = \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \mathcal{C} & \sqrt{1-\rho^2}\mathcal{C} \\ \sqrt{1-\rho^2}\mathcal{C} & \mathcal{C} \end{pmatrix} \right)$$

and as such $h_0^{\intercal}(\mathrm{d}u, \mathrm{d}v) = h_0(\mathrm{d}u, \mathrm{d}v)$. Thus, by setting $\nu(\mathrm{d}u) = \mu_{\mathsf{pr}}$ and $Q_{\mathrm{ref}}(u, \mathrm{d}v) = Q(u, \cdot) = (\sqrt{1-\rho^2}u, \rho^2\mathcal{C})$ in the notation of Lemma 3.4.1, one obtains the well-known *pre-conditioned Crank-Nicolson* variant of the MH algorithm, which has an acceptance probability given by:

$$\alpha(u, v) = \min\left\{1, \exp\left(\Phi(u; y) - \Phi(v; y)\right)\right\}. \tag{3.30}$$

It is shown by Hairer et. Al. [65] that the pCN algorithm converges with an L_2-spectral gap that is independent of the dimensionality of the state space.
There are two closely related –yet independently developed, extensions of the pCN algorithm by [130] and [153, 144].

ν-pCN [130]

The first one, presented by Pinski et. Al. in [130], assumes that $A_y := \{u \in \mathsf{X} : \Phi(u; y) = \infty\}$ has μ_{pr}-measure equal to zero $\forall y$, and sets once again $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$ and defines $\nu = \mathcal{N}(m_{\mathrm{KL}}, \mathcal{H}_{\mathrm{KL}})$ as a reference measure, where m_{KL} and $\mathcal{H}_{\mathrm{KL}}$ are the mean and covariance operator of a Gaussian measure minimizing the Kullback-Liebler (KL) divergence $d_{\mathrm{KL}}(\mu^y, \nu)$ between μ^y and ν, constructed in such a way that $\mu^y \simeq \nu$ (notice that this implies that $\mu^y \simeq \mu_{\mathsf{pr}}$, see [129]), with

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\nu}(u) = \frac{\mathrm{d}\mu}{\mathrm{d}\mu_{\mathsf{pr}}}(u)\frac{\mathrm{d}\mu_{\mathsf{pr}}}{\mathrm{d}\nu}(u) = \exp\left(-\Phi(u; y) + \Phi_\nu(u)\right) =: F(u; y), \quad \text{with} \tag{3.31}$$
$$\Phi_\nu(u) := -\left\langle u - m_{\mathrm{KL}}, m_{\mathrm{KL}}\right\rangle_{\mathcal{C}} + \frac{1}{2}\left\langle u - m_{\mathrm{KL}}, \left(\mathcal{H}_{\mathrm{KL}}^{-1} - \mathcal{C}^{-1}\right)(u - m_{\mathrm{KL}})\right\rangle - \frac{1}{2}\left\|m_{\mathrm{KL}}\right\|_{\mathcal{C}}^2,$$

where the mean and covariance operator of ν are such that $m_{\mathrm{KL}} \in \mathrm{Im}(\mathcal{C}^{1/2})$, and $\mathrm{Im}(\mathcal{C}^{1/2}) = \mathrm{Im}(\mathcal{H}_{\mathrm{KL}}^{1/2})$. Intuitively, by optimizing over a class of Gaussian measures such that $\nu \ll \mu_{\mathsf{pr}}$, it follows that $\nu \simeq \mu_{\mathsf{pr}}$ as a consequence of the Cameron-Martin theorem (c.f. Theorem 2.1.3); see the work of [129] for more details. Given a state $u \in \mathsf{X}$, ν induces a proposal kernel of the form $Q_{\mathrm{KL}}(u, \cdot) = \mathcal{N}\left(m_{\mathrm{KL}} + \sqrt{1-\rho^2}(u - m_{\mathrm{KL}}), \rho^2\mathcal{H}_{\mathrm{KL}}\right)$, which can be shown to be ν-reversible following a similar procedure as before. Denote $h_0(\mathrm{d}u, \mathrm{d}v) = Q_{\mathrm{KL}}(u, \mathrm{d}v)\nu(\mathrm{d}u)$, and

let $z_1, z_2 \sim \mathcal{N}(0, \mathcal{H}_{\mathrm{KL}})$, which implies $z_i + m_{\mathrm{KL}} \sim \nu$ $i = 1, 2$. Following a similar procedure as before then gives

$$\begin{aligned}\begin{pmatrix} u \\ v \end{pmatrix} &= \begin{pmatrix} m_{\mathrm{KL}} \\ m_{\mathrm{KL}} \end{pmatrix} + \begin{pmatrix} z_1 \\ \sqrt{1-\rho^2} z_1 + \rho z_2 \end{pmatrix} \\ &= \begin{pmatrix} m_{\mathrm{KL}} \\ m_{\mathrm{KL}} \end{pmatrix} + \begin{pmatrix} I & 0 \\ \sqrt{1-\rho^2} I & \rho I \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \sim h_0(\mathrm{d}u, \mathrm{d}v),\end{aligned}$$

which implies that

$$h_0(\mathrm{d}u, \mathrm{d}v) = \mathcal{N}\left(\begin{pmatrix} m_{\mathrm{KL}} \\ m_{\mathrm{KL}} \end{pmatrix}, \begin{pmatrix} \mathcal{H}_{\mathrm{KL}} & \sqrt{1-\rho^2}\mathcal{H}_{\mathrm{KL}} \\ \sqrt{1-\rho^2}\mathcal{H}_{\mathrm{KL}} & \mathcal{H}_{\mathrm{KL}} \end{pmatrix} \right).$$

Proceeding just as in the case for the pCN algorithm, a similar computation shows that $(v, u)^{\mathsf{T}} \sim h_0(\mathrm{d}u, \mathrm{d}v)$, which implies $h_0(\mathrm{d}u, \mathrm{d}v) = h_0^{\mathsf{T}}(\mathrm{d}u, \mathrm{d}v)$, meaning that the kernel $Q_{\mathrm{KL}}(\cdot, \cdot)$ is ν-reversible. Setting $Q_{\mathrm{ref}} = Q_{\mathrm{KL}}$, It then follows from Lemma 3.4.1 that the MH algorithm induced by using a ν-reversible proposal kernel Q_{KL} is well-defined and its acceptance probability is given by

$$\alpha(u, v) = \min\{1, \exp(F(v; y) - F(u; y))\}.$$

Intuitively, $F(u; y)$ would tend to be smaller that $\Phi(u; y)$, at least for regions of high probability with respect to μ^y. Thus, for some fixed ρ, this extension of the pCN algorithm would tend to accept more often that the standard pCN, thus providing a faster mixing. We will refer to this method as *ν-pCN*.

Generalized pCN [?, 130, 144, 153]

A second extension to the pCN algorithm is presented by Sprungk and Rudolf in [153, 144], in the spirit of the operator weighted proposals work of Law [97], by considering Gaussian proposals whose covariance resemble that of the target measure μ^y. Let $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$, and $G : \mathsf{X} \to \mathsf{X}$ be a bounded, self-adjoint, and positive linear operator. Furthermore, define the following bounded linear operators on X:

$$H_G := \mathcal{C}^{1/2} G \mathcal{C}^{1/2}, \quad C_G := \mathcal{C}^{1/2}(I + H_G)^{-1}\mathcal{C}^{1/2}, \quad A_G := \mathcal{C}^{1/2}\sqrt{I - \rho^2(I + H_G)^{-1}}\mathcal{C}^{-1/2}.$$

The *generalized preconditioned Crank-Nicolson* algorithm of [153, 144], is then defined by using the μ_{pr}-reversible kernel $Q_{\mathrm{gpCN}}(u, \cdot) = \mathcal{N}(A_G u, \rho^2 C_G)$ (see [153, p. 318]) in the MH algorithm. Setting $Q_{\mathrm{gpCN}}(u, \cdot) = Q_{\mathrm{ref}}(u, \cdot)$, it follows from Lemma 3.4.1 that the Metropolis-Hastings acceptance ratio is well defined and it is of the same form as Equation (3.30).

In the case where μ^y is induced by a Bayesian inverse problem with (finite-dimensional) additive Gaussian noise of the form $\mu_{\text{noise}} = \mathcal{N}(0, \sigma^2_{\text{noise}} I)$, [144] suggests to set $G = (\sigma^{-2}_{\text{noise}} \mathcal{L}\mathcal{L}^{\mathsf{T}})$, with $\mathcal{L} := D\Phi(u_{\text{map}})$, where $u_{\text{map}} \in \mathsf{X}$ is the *maximum a posteriori point* of μ^y. In practice, $D\Phi(u_{\text{map}})$ can be efficiently computed using the adjoint state method, as in [24, 23, 164]. Notice that both of these extensions to the pCN algorithm are closely related to the Laplace-approximation and variational methods described in Section 2.3, however, given that they are being used in the context of a *Metropolized* algorithm, the samples obtained follow the desired target distribution instead of just an approximation of it.

Infinite-dimensional Metropolis-adjusted Langevin Algorithm (∞-MALA) [12]

Alternatively, Assume that $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$ and that $\forall y \in \mathsf{Y}, \mathcal{C}D\Phi(u; y) \in \mathsf{Im}(\mathcal{C}^{1/2})$, μ_{pr}-a.s. Setting $b = a = 1, \mathcal{K} = \mathcal{C}$ in (3.29), a semi-implicit Euler scheme yields the discretized model

$$u_{n+1} = \sqrt{1-\rho^2}u_n - \rho\sqrt{\frac{\tau}{2}}CD\Phi(u_n; y) + \rho\xi, \quad \xi \sim \mathcal{N}(0, \mathcal{C}),$$

which induces the proposal kernel

$$Q_{\text{MALA}}(u, \cdot) = \mathcal{N}\left(\sqrt{1-\rho^2}u - \rho\sqrt{\frac{\tau}{2}}CD\Phi(u; y), \rho^2\mathcal{C}\right).$$

Setting $Q_{\text{ref}}(u, \cdot) = \mathcal{N}\left(\sqrt{1-\rho^2}u, \rho^2\mathcal{C}\right)$ (which is known to be μ_{pr}-reversible, from the discussion on pCN) it follows from the Cameron-Martin theorem (Theorem 2.1.3) that $Q_{\text{ref}}(u, \cdot) \simeq Q_{\text{MALA}}(u, \cdot)$, with

$$\frac{\mathrm{d}Q_{\text{MALA}}(u, \cdot)}{\mathrm{d}Q_{\text{ref}}(u, \cdot)}(v) = \exp\left(-\frac{\rho^2\tau}{4}\|\mathcal{C}D\Phi(u; y)\|^2_{\mathcal{C}} - \left\langle \rho\sqrt{\frac{\tau}{2}}CD\Phi(u; y), v - \sqrt{1-\rho^2}u\right\rangle_{\mathcal{C}}\right) = g(u, v). \tag{3.32}$$

Thus, setting once again $\nu = \mu_{\mathsf{pr}}$, and since Q_{ref} is μ_{pr}-reversible it then follows from Lemma 3.4.1 that the infinite-dimensional MALA algorithm is well-defined in function spaces, and its Metropolis-Hastings acceptance probability is given by

$$\alpha(u, v) = \min\left\{1, \exp\left(\Phi(u; y) - \Phi(v; y)\right)\frac{g(v, u)}{g(u, v)}\right\},$$

with $g(\cdot, \cdot)$ as in (3.32).

ν-MALA:

One can, of course, combine the ideas behind ν-pCN and ∞-MALA as follows. Once again, assume $\forall y \in \mathsf{Y}, \mathcal{C}D\Phi(u;y) \in \mathsf{Im}(\mathcal{C}^{1/2})$, μ_{pr}-a.s., let $\mu_{\mathsf{pr}} = \mathcal{N}(0,\mathcal{C})$, take $\nu = \mathcal{N}(m_{\mathrm{KL}}, \mathcal{H}_{\mathrm{KL}}) \simeq \mu_{\mathsf{pr}}$ as in the ν-pCN method, with $m_{\mathrm{KL}} \in \mathrm{Im}(\mathcal{C}^{1/2})$ and $\mathrm{Im}(\mathcal{C}^{1/2}) = \mathrm{Im}(\mathcal{H}_{\mathrm{KL}}^{1/2})$ and define the transition kernel $Q_{\nu\text{-MALA}} : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \to [0,1]$:

$$Q_{\nu\text{-MALA}}(u,\cdot) := \mathcal{N}\left(m_{\mathrm{KL}} + \sqrt{1-\rho^2}(u_n - m_{\mathrm{KL}}) - \rho\sqrt{\frac{\tau}{2}}\mathcal{C}D\Phi(u;y), \rho^2\mathcal{H}_{\mathrm{KL}}\right).$$

Setting $Q_{\mathrm{ref}} = Q_{\mathrm{KL}}$ (known to be ν-reversible), it is a consequence of the Cameron-Martin theorem that $Q_{\mathrm{KL}} \simeq Q_{\nu\text{-MALA}}$, with

$$\frac{\mathrm{d}Q_{\nu\text{-MALA}}(u,\cdot)}{\mathrm{d}Q_{\mathrm{KL}}(u,\cdot)}(v) = \exp\left(-\frac{\rho^2\tau}{4}\|\mathcal{C}D\Phi(u;y)\|_{\mathcal{H}_{\mathrm{KL}}}^2 - \left\langle \rho\sqrt{\frac{\tau}{2}}\mathcal{C}D\Phi(u;y), v - m_{\mathrm{KL}} + \sqrt{1-\rho^2}(u_n - m_{\mathrm{KL}})\right\rangle_{\mathcal{H}_{\mathrm{KL}}}\right) = g'(u,v).$$

Once again, the conditions of Lemma 3.4.1 are satisfied, implying that the MH algorithm induced by taking $Q_{\nu\text{-MALA}}(u,\cdot)$ as a proposal kernel is well-defined, with acceptance probability given by

$$\alpha(u,v) = \min\left\{1, \exp\left(F(u;y) - F(v;y)\right)\frac{g'(v,u)}{g'(u,v)}\right\},$$

with F defined as in (3.31).

Hamiltonian Monte Carlo (HMC)

A common shortfall of diffusion-based proposals in the MH algorithm is the potentially slow exploration rate of the state space. One idea borrowed from physics, which, can be applied to most problems with continuous state space, is to introduce a "fictitious" Hamiltonian dynamics and "fictitious" momentum variables. We begin by describing such a method in the finite-dimensional case.

Let $\mathsf{X} = \mathbb{R}^m$. We recall that a Hamiltonian dynamical system is characterized by a *Hamiltonian function* $H : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$, $H = H(u,w)$, that is conserved during dynamics. Here $u \in \mathsf{X}$ denotes the position vector and w denotes the momentum vector. The Hamiltonian dynamics is governed by the equations

$$\frac{\mathrm{d}u_i}{\mathrm{d}t} = \frac{\partial H}{\partial w_i} \tag{3.33}$$

$$\frac{\mathrm{d}w_i}{\mathrm{d}t} = -\frac{\partial H}{\partial u_i} \tag{3.34}$$

for $i = 1, \dots, m$. In general, the above equation can be understood as a conservation of the total energy of a system in time.

Hamiltonian Monte Carlo takes inspiration from the previous physical system in order to construct a Markov Chain Monte Carlo algorithm with a given invariant density $\mu^y(u)$ on the position variables u. To do so, we introduce the *potential energy* $U(u) = -\log \mu^y(u)$, a *kinetic energy* $K(w) = \frac{1}{2} w^\mathsf{T} M^{-1} w$, for some mass matrix $M \in \mathbb{R}^{m \times m}$, and the Hamiltonian $H(u, w) = U(u) + K(w)$. Having introduced these functions, we can then simulate a Markov chain in which each iteration re-samples the momentum, evolves the Hamiltonian system for a certain time, and then does a Metropolis-type acceptance-rejection step on the new position vector. More concretely, we consider the so-called Gibbs distribution, given by

$$G(u, w) = \frac{1}{Z} \exp(-H(u, w)) = \frac{1}{\tilde{Z}} \exp(-U(u)) \frac{1}{\sqrt{2\pi |\det M|}} \exp(-K(w))$$

where Z is the (unknown) normalizing constant, $\frac{1}{\tilde{Z}} \exp(-U(u))$ is the probability density we are interested and $\frac{1}{\sqrt{2\pi |\det M|}} \exp(-K(w))$ is the density of a multivariate Gaussian distribution centered at 0 with covariance M. Given the state u^n at iteration n, the idea of the algorithm is then to sample a momentum vector w^n, and compute, for each iteration, $H(u^n, w^n)$. The Hamiltonian system is then evolved starting from $u(0) = u^n, w(0) = w^n$, on a time interval $[0, T]$ using equations (3.33), and (3.34) for some arbitrary final time T, to obtain $(u(T), w(T))$, where, in general, $u(T) \neq q(0)$. This state is then taken as the proposal state in a Metropolis-Hastings step to generate the new state u^{n+1}. For many problems of modern relevance, it is not possible to compute the dynamics exactly and numerical discretization is needed. A convenient time discretization scheme is the *Verlet's method*: the time interval $[0, T]$ is divided into N_t intervals of size $\epsilon > 0$ and for each particle i the position q_i and momemtum p_i are updated as follows

$$\begin{aligned} w(t + \epsilon/2) &= w_i(t) - \frac{\epsilon}{2} \nabla U(u(t)) \\ u(t + \epsilon) &= u_i(t) + \epsilon M^{-1} w \left(t + \frac{\epsilon}{2}\right) \\ w(t + \epsilon) &= w_i \left(t + \frac{\epsilon}{2}\right) - \frac{\epsilon}{2} \nabla U(u(t + \epsilon)). \end{aligned}$$

The main steps of the Hamiltonian Monte Carlo algorithm using Verlet's method are outlined in Algorithm 3. There, N is the length of the chain, ϵ the time step in Verlet's method, and T the final integration time.
Notice that, similar to the random-walk Metropolis, this algorithm depends on few parameters, namely, ϵ, T , and M, which should be properly tuned. Furthermore, it is worth noting the equivalence between MALA and the HMC algorithm with a 1-step evolution.

Infinite-dimensional HMC [12]

Algorithm 3 Hamiltonian Monte Carlo

procedure HAMILTONIAN MONTE CARLO(N,μ^y,M,λ^0).
  Sample $u^0 \sim \lambda^0$
  for $n = 0, \dots, N-1$ **do**
    Sample new values for the momentum variables, $w^n \sim \mathcal{N}(0, M)$
    Given the current state (u^n, w^n), propose a new state (u^*, w^*) by evolving the Hamiltonian system (3.33), (3.34) using Verlet's method.
    Set $u^{n+1} = u^*$ with probability α, where

$$\alpha = \min\left[1, \exp\left(-U(u^*) + U(u^n) - K(w^*) + K(w^n)\right)\right]$$

  end for
  Output $\{u^n\}_{n=0}^N$
end procedure

Similarly, the work of [12] extends the HMC algorithm in function space in the case where $\mu_{\text{pr}} = \mathcal{N}(0, \mathcal{C})$, with $\mathcal{C}$ a positive, self-adjoint and trace-class operator. In this case, the infinite dimensional HMC (∞-HMC) algorithm behaves as Algorithm 3, with the modification that $w \sim \mu_{\text{pr}}$ and that the acceptance probability $\alpha(u, u^*)$ is given by $\min\{1, \exp(-\Delta H(\boldsymbol{u}, \boldsymbol{w}))\}$, where $\boldsymbol{u} := (u = u_0, u_1, \dots, u_{\lceil T/\epsilon \rceil - 1}, u_{\lceil T/\epsilon \rceil} = u^*)$ and $\boldsymbol{w} := (w = w_0, w_1, \dots, w_{\lceil T/\epsilon \rceil})$ are the intermediate values of u and w over the temporal evolution of the Hamiltonian system, and

$$\begin{aligned}\Delta H(\boldsymbol{u}, \boldsymbol{w}) := \Phi(u^*; y) - \Phi(u; y) - \frac{\epsilon^2}{8}\left\{\left\|\mathcal{C}^{1/2} D\Phi(u^*; y)\right\|_{\mathsf{X}}^2 - \left\|\mathcal{C}^{1/2} D\Phi(u; y)\right\|_{\mathsf{X}}^2\right\} \\ - \frac{\epsilon}{2} \sum_{i=0}^{\lceil T/\epsilon \rceil - 1} \left(\langle w_i, D\Phi(u_i; y)\rangle_{\mathsf{X}} + \langle w_{i+1}, D\Phi(u_{i+1}; y)\rangle_{\mathsf{X}}\right)\end{aligned} \tag{3.35}$$

(see [12] for derivation of (3.35)). Further extensions of the infinite dimensional MALA and HMC methods which exploit the local geometry of $\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\text{pr}}}(u)$, are presented in the works of [12] and [91].

4 Generalized parallel tempering on Bayesian inverse problems

This Chapter is essentially the same as Publication Latz, J., Madrigal-Cianci, J.P., Nobile, F. et al. *Generalized parallel tempering on Bayesian inverse problems*. Stat Comput 31, 67 (2021). [95]. Small modification have been made with respect to such an article in order to avoid repeating concepts and definitions already presented in previous chapters, and notation has been modified with respect to the published article in order to make it consistent with the notation used throughout this thesis. We have also removed some material that has already been discussed in the state of the art. Here, we present our first hierarchical approach, which introduces and exploits a sequence of *tempered* distributions approximating the posterior of interest, as presented in Chapter 1. More precisely, we present two generalizations of the Parallel Tempering algorithm in the context of discrete-time Markov chain Monte Carlo methods for Bayesian inverse problems. These generalizations use state-dependent swapping rates, inspired by the so-called continuous time Infinite Swapping algorithm presented in [47]. We analyze the reversibility and ergodicity properties of our generalized PT algorithms. Numerical results on sampling from different target distributions show that the proposed methods significantly improve sampling efficiency over more traditional sampling algorithms such as Random Walk Metropolis, preconditioned Crank-Nicolson, and (standard) Parallel Tempering.

4.1 Introduction

Modern computational facilities and recent advances in computational techniques have made the use of Markov Chain Monte Carlo (MCMC) methods feasible for some large-scale Bayesian inverse problems (BIP), where the goal is to characterize the posterior distribution of a set of parameters u of a computational model which describes some physical phenomena, conditioned on some (usually indirectly) measured data y. However, some computational difficulties are prone to arise when dealing with *difficult to explore* posteriors, i.e., posterior distributions that are multi-modal, or that concentrate around a non-linear, lower-dimensional manifold, since some of the more commonly-used Markov transition kernels in MCMC algorithms, such as random walk Metropolis (RWM) or preconditioned Crank-Nicholson (pCN), are not well-suited in such situations. This in turn can make the computational time needed to properly *explore* these complicated target distributions arbitrarily long. Some recent works address these issues by employing Markov transitions kernels that use geometric information [12]; however, this requires efficient computation of the gradient

of the posterior density, which might not always be feasible, particularly when the underlying computational model is a so-called "black-box". (this is new) One such way of alleviating these issues is with *tempering strategies*, such as the ones in [42, 52, 96, 114, 167]. In particular, we will focus on *parallel-tempering techniques* [52, 90, 114], as described in Chapter 1, and whose main idea we will recall next, for convenience. In short, parallel tempering algorithms simultaneously run K independent MCMC chains, where each chain is invariant with respect to a *flattened* (referred to as *tempered*) version of the posterior of interest μ^y, while, at the same time, proposing to swap states between any two chains every so often. Such a swap is then accepted using the standard Metropolis-Hastings (MH) acceptance-rejection rule. Intuitively, chains with a larger smoothing parameter (referred to as *temperature*) will be able to better explore the parameter space. Thus, by proposing to exchange states between chains that target posteriors at different temperatures, it is possible for the chain of interest (i.e., the one targeting μ^y) to mix faster, and to avoid the undesirable behavior of some MCMC samplers, such as the diffusion-based methods, presented in Chapter 3, of getting "stuck" in a mode. Moreover, the fact that such an exchange of states is accepted with the typical MH acceptance-rejection rule, will guarantee that the chain targeting μ remains invariant with respect to such probability measure [52]. A improvement over the PT approach in the context of (inherently time-continuous) molecular dynamics is presented in the so-called *Infinite Swapping (IS)* algorithm [49, 133]; a continuous-time Markov process which considers the limit where states between chains are swapped infinitely often. It is shown in [49] that such an approach can in turn be understood as a swap of dynamics, i.e., kernel and temperature (as opposed to states) between chains. We remark that once such a change in dynamics is considered, it is not possible to distinguish particles belonging to different chains. However, since the stationary distribution of each chain is known, importance sampling can be employed to compute posterior expectations with respect to the target measure of interest. Infinite Swapping has been successfully applied in the context of computational molecular dynamics and rare event simulation [47, 103, 133]; however, to the best of our knowledge, a (discrete-time) equivalent to such method has not been implemented in the context of Bayesian inverse problems.

In light of this, the current work aims at importing such ideas to the BIP setting, by presenting them in a discrete-time Metropolis-Hastings Markov chain Monte Carlo context. We will refer to these algorithms as *Generalized Parallel Tempering* (GPT). We emphasize, however, that these methods are *not* a time discretization of the continuous-time Infinite Swapping presented in [49], but, in fact, a discrete-time Markov process inspired by the ideas presented therein with suitably defined state-dependent probabilities of swapping states or dynamics. We now summarize the main contributions of this chapter.

We begin by presenting a generalized framework for discrete time PT in the context of MCMC for BIP, and then proceed to propose, analyze and implement two novel state-dependent PT algorithms inspired by the ideas presented in [49].

Furthermore, we prove that our GPT methods have the right invariant measure, by showing reversibility of the generated Markov chains, and prove their ergodicity. Finally, we implement the proposed GPT algorithms for an array of Bayesian inverse problems, comparing their efficiency

to that of an un-tempered, (single temperature), version of the underlying MCMC algorithm, and standard PT. For the base method to sample at the cold temperature level, we use Random Walk Metropolis (RWM) (Sections 4.5.3-4.5.6) or preconditioned Crank-Nicolson (Section 4.5.7), however, we emphasize that our methods can be used together with any other, more efficient base sampler. Experimental results show improvements in terms of computational efficiency of GPT over un-tempered RWM and standard PT, thus making the proposed methods attractive from a computational perspective. From an implementation perspective, the swapping component of our proposed methods is rejection-free, thus effectively eliminating some tuning parameters on the PT algorithm, such as swapping frequency.
We remark that a PT algorithm with state-dependent swapping probabilities has been proposed in [90], however, such a work only considers pairwise swapping of chains and a different construction of the swapping probabilities, resulting in a less-efficient sampler, at least for the BIPs addressed in this work.
Our ergodicity result relies on an L_2 spectral gap analysis. It is known [143] that when a Markov chain is both reversible and has a positive L_2-spectral gap, one can in turn provide non-asymptotic error bounds on the mean square error of an ergodic estimator of the chain. Our bounds on the L_2-spectral gap, however, are far from being sharp and could possibly be improved using e.g., domain decomposition ideas as in [171]. Such analysis is left for a future work.
The rest of this Chapter is organized as follows. Section 4.2 is devoted to the introduction of some additional notation. In Section 4.3 we provide a brief review of (standard) PT (Section 4.3.2), and introduce the two versions of the GPT algorithm in Sections 4.3.3 and 4.3.4, respectively. In fact, we present a general framework that accommodates both the standard PT algorithms and our generalized versions. In Section 4.4, we present the main theoretical results of the current Chapter (Theorems 4.4.1 and 4.4.2). The proof of these Theorems is given by a series of Propositions in Section 4.4.1. Lastly, we illustrate our methods on various numerical experiments in Section 4.5.

4.2 Problem setting

4.2.1 Notation

Let $(\mathsf{X}_i, \|\cdot\|)$ be a separable Banach space with associated Borel σ-algebra $\mathcal{B}(\mathsf{X}_i)$, $i = 1, 2$ and let μ_i, ν_i be probability measures on $(\mathsf{X}_i, \mathcal{B}(\mathsf{X}_i))$, with $\mu_i \ll \nu_i$, and denote by $\pi_i : \mathsf{X} \to \mathbb{R}_+$ the corresponding Radon-Nikodym derivative $\pi_i(u) = \frac{\mathrm{d}\mu_i}{\mathrm{d}\nu_i}(u)$. The *product* of these two measures is defined by

$$\begin{aligned}\boldsymbol{\mu}(A) &= (\mu_1 \times \mu_2)(A) \\ &= \iint_A \pi_1(u_1)\pi_2(u_2)\nu_1(\mathrm{d}u_1)\nu_2(\mathrm{d}u_2),\end{aligned}$$

for all $A \in \mathcal{B}(\mathsf{X}_1 \times \mathsf{X}_2)$. Joint measures on $(\mathsf{X}_1 \times \mathsf{X}_2, \mathcal{B}(\mathsf{X}_1, \times\mathsf{X}_2))$ will always be written in boldface, hereafter.

Let $P_k,\ k = 1, 2$, be Markov transition operators associated to kernels $p_k : \mathsf{X}_k \times \mathcal{B}(\mathsf{X}_k) \mapsto [0, 1]$ (c.f 3.1.2). We define the *tensor product Markov operator* $\mathbf{P} := P_1 \otimes P_2$ as the Markov operator associated with the product measure $\mathbf{p}(\boldsymbol{u}, \cdot) = p_1(u_1, \cdot) \times p_2(u_2, \cdot),\ \boldsymbol{u} = (u_1, u_2) \in \mathsf{X}_1 \times \mathsf{X}_2$. In particular, $\boldsymbol{\nu}\mathbf{P}$ is the measure on $(\mathsf{X}_1 \times \mathsf{X}_2, \mathcal{B}(\mathsf{X}_1 \times \mathsf{X}_2))$ that satisfies

$$(\boldsymbol{\nu}\mathbf{P})(A_1 \times A_2) = \iint_{\mathsf{X}_1 \times \mathsf{X}_2} p_1(u_1, A_1) p_2(u_2, A_2) \boldsymbol{\nu}(\mathrm{d}u_1, \mathrm{d}u_2),$$

for all $A_1 \in \mathcal{B}(\mathsf{X}_1)$ and $A_2 \in \mathcal{B}(\mathsf{X}_2)$. Moreover, $(\mathbf{P}f) : \mathsf{X}_1 \times \mathsf{X}_2 \to \mathbb{R}$ is the function given by

$$(\mathbf{P}f)(\boldsymbol{u}) = \iint_{\mathsf{X}_1 \times \mathsf{X}_2} f(z_1, z_2) p_1(u_1, \mathrm{d}z_1) p_2(u_2, \mathrm{d}z_2),$$

for an appropriate $f : \mathsf{X}_1 \times \mathsf{X}_2 \to \mathbb{R}$, $\mathcal{B}(\mathsf{X}_1 \times \mathsf{X}_2)$-measurable.
Recall that a Markov operator P (resp. $\mathbf{P}$) is *invariant* with respect to a measure ν (resp. $\boldsymbol{\nu}$) if $\nu P = \nu$ (resp. $\boldsymbol{\nu}\mathbf{P} = \boldsymbol{\nu}$). For two given ν-invariant Markov operators P_1, P_2, we say that $P_1 P_2$ is a *composition* of Markov operators, not to be confused with $P_1 \otimes P_2$. Furthermore, given a composition of K ν-invariant Markov operators $P_c := P_1 P_2 \dots P_K$, we say that P_c is *palindromic* if $P_1 = P_K, P_2 = P_{K-1}, ..., P_k = P_{K-k+1},\ k = 1, 2 \dots, K$. It is known (see, e.g., [21, Section 1.12.17]) that a palindromic, ν-invariant Markov operator P_c has an associated Markov transition kernel p_c which is ν-reversible.

4.2.2 Tempering

Denote by $p : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \to [0, 1]$ the Markov transition kernel induced by the Metropolis-Hastings algorithm (c.f. Section 3.4.1), using a diffusion-based proposal kernel $q_{\text{prop}}(u, \cdot)$, such as the *random walk Metropolis* or *preconditioned Crank Nicolson* algorithms, where, given some suitable covariance operator $\Sigma : \mathsf{X} \to \mathsf{X}$, $q_{\text{prop}}(u^n, \cdot) = \mathcal{N}(u^n, \rho\Sigma)$ or $q_{\text{prop}}(u^n, \cdot) = \mathcal{N}(\sqrt{1-\rho^2}u^n, \rho^2\Sigma)$, $0 < \rho < 1$, respectively. This type of proposals are widely used in practice, however, they tend to present some undesirable behaviors when sampling from certain *difficult* measures, which are, for example, concentrated over a manifold or are multi-modal [52]. In the first case, in order to avoid a large rejection rate, the "step-size" ρ of the proposal kernel must be quite small, which will in turn produce highly-correlated samples. In the second case, chains generated by these *localized* kernels tend to get stuck in one of the modes. In either of these cases, very long chains are required to properly explore the parameter space.
One way of overcoming such difficulties is to introduce tempering. As in Chapter 2, write

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\text{pr}}}(u) = \frac{e^{-\Phi(u;y)}}{Z} =: \pi^y(u),$$

and, given a set of K temperatures $\{T_k\}_{k=1}^K$, such that $1 = T_1 < T_2 \cdots < T_K$, define $\mu_k^y \ll \mu_{\mathsf{pr}}$ such that

$$\frac{\mathrm{d}\mu_k^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u) := \frac{e^{-\Phi(u;y)/T_k}}{Z_k} =: \pi_k^y(u), \tag{4.1}$$

where $Z_k := \int_{\mathsf{X}} e^{-\Phi(u;y)/T_k} \mu_{\mathsf{pr}}(\mathrm{d}u)$, and with $\Phi(u;y)$ the potential function defined in Theorem 2.2.1. In the case where $T_K = \infty$, we set $\mu_K^y = \mu_{\mathsf{pr}}$. Notice that μ_1^y corresponds to the target posterior measure.

Remark 4.2.1 (On notation): *Notice that we have used the inverse notation with respect to Chapter 1, setting $\mu^y = \mu_1^y$ instead of $\mu^y = \mu_K^y$. Furthermore, notice that we are not including any discretization accuracy in our formulation, i.e., we are not using $\Phi_{\mathsf{L}}(u;y)$ to denote $\Phi(u;y)$ evaluated at an accuracy level L, and assume that all models are evaluated at the same discretization accuracy. We hope this is not a cause of confusion to the reader.*

We say that for $k = 2, \ldots, K$, each measure μ_k^y is a *tempered* version of μ_1^y. In general, the $1/T_k$ term in (4.1) serves as a "smoothing"[1] factor, which in turn makes μ_k^y easier to explore as $T_k \to \infty$. In the "standard " parallel tempering MCMC algorithm [52], one samples from all posterior measures μ_k^y simultaneously. Here, we first use a μ_k^y-reversible Markov transition kernel p_k on each chain, and then, we propose to exchange states between chains at two consecutive temperatures, i.e., chains targeting $\mu_k^y, \mu_{k-1}^y,\ k \in \{2, \ldots, K\}$. Such a proposed swap is then accepted or rejected with a standard Metropolis-Hastings acceptance rejection step. This procedure is presented in Algorithm 4. Notice that such an algorithm does a systematic sweep across temperatures going *from hot-to-cold*. Alternatively, one could construct such an algorithm going *from cold-to-hot*; i.e., swapping chains at temperatures T_k and T_{k+1}, $k = \{1, 2, \ldots, K-1\}$. Our numerical examples in Section 5.6 implement Algorithm 4 in such a way that the order of the swapping (i.e., either *from hot-to-cold* or *from cold-to-hot*) is alternated at every iteration. We remark that such an algorithm can be modified to, for example, propose to swap states every N_s steps of the chain, or to swaps states between two chains μ_i^y, μ_j^y, with i, j chosen randomly and uniformly from the index set $\{1, 2, \ldots, K\}$. In the next section we present the generalized PT algorithms which swap states according to a random permutation of the indices drawn from a state dependent probability.

4.3 Generalizing Parallel Tempering

Infinite Swapping was initially developed in the context of continuous-time MCMC algorithms, which were used for molecular dynamics simulations. In continuous-time PT, the swapping of the states is controlled by a Poisson process on the set $\{1, \ldots, K\}$. Infinite Swapping is the limiting algorithm obtained by letting the waiting times of this Poisson process go to zero. Hence, we swap the states of the chain infinitely often over a finite time interval. We refer to [49] for a thorough

[1] Here, smoothing is to be understood in the sense that it *flattens* the density.

Algorithm 4 Standard PT.

function STANDARD PT(N, $\{p_k\}_{k=1}^K$, $\{\pi_k^y\}_{k=1}^K$, μ_{pr})
  Sample $u_k^1 \sim \mu_{\mathsf{pr}},\ k = 1, \dots, K$
  # Do one step of MH on each chain
  for $n = 1, 2, \dots, N-1$ **do**
    for $k = 1, \dots, K$ **do**
      Sample $u_k^{n+1} \sim p_k(u_k^n, \cdot)$
    end for
    # Swap states
    for $k = K, K-1, \dots, 2$ **do**
      Swap states u_k^{n+1} and u_{k-1}^{n+1} with probability

$$\alpha_{\text{swap}} = \min\left\{1, \frac{\pi_k^y(u_{k-1}^{n+1})\pi_{k-1}^y(u_k^{n+1})}{\pi_k^y(u_k^{n+1})\pi_{k-1}^y(u_{k-1}^{n+1})}\right\}$$

    end for
  end for
  Output $\{u_1^n\}_{n=1}^N$
end function

introduction and review of Infinite Swapping in continuous-time. In Section 5 of the same article, the idea to use Infinite Swapping in time-discrete Markov chains was briefly discussed. Inspired by this discussion, we present two Generalizations of the (discrete-time) Parallel Tempering strategies. To that end, we propose to either (i) swap states in the chains at every iteration of the algorithm in such a way that the swap is accepted with probability one, which we will refer to as the *Unweighted Generalized Parallel Tempering (UGPT)*, or (ii), swap dynamics (i.e., swap kernels and temperatures between chains) at every step of the algorithm. In this case, importance sampling must also be used when computing posterior expectations since this in turn provides a Markov chain whose invariant measure is not the desired one. We refer to this approach as *Weighted Generalized Parallel Tempering (WGPT)*. We begin by introducing a common framework to both PT and the two versions of GPT.

Let $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$ be a separable Banach space with associated Borel σ-algebra $\mathcal{B}(\mathsf{X})$. Let us define the K-fold product space $\mathsf{X}^K := \bigtimes_{k=1}^K \mathsf{X}$, with associated product σ-algebra $\mathcal{B}^K := \bigotimes_{k=1}^K \mathcal{B}(\mathsf{X})$, as well as the product measure on $(\mathsf{X}^K, \mathcal{B}^K)$

$$\boldsymbol{\mu^y} := \bigtimes_{k=1}^K \mu_k^y, \tag{4.2}$$

where μ_k^y, $k = 1, \dots, K$ are the tempered measures with temperatures $1 = T_1 < T_2 < T_3 < \cdots < T_K \leq \infty$ introduced in the previous section. Similarly, we define the product prior measure $\boldsymbol{\mu}_{\mathsf{pr}} := \bigtimes_{k=1}^{K} \mu_{\mathsf{pr}}$. Notice that $\boldsymbol{\mu}^{\boldsymbol{y}}$ has a density $\boldsymbol{\pi}^{\boldsymbol{y}}(\boldsymbol{u})$ with respect to $\boldsymbol{\mu}_{\text{prior}}$ given by

$$\boldsymbol{\pi}^{\boldsymbol{y}}(\boldsymbol{u}) = \prod_{k=1}^{K} \pi_k^y(u_k), \quad \boldsymbol{u} := (u_1, \dots, u_K) \in \mathsf{X}^K,$$

with $\pi_i^y(u)$ given as in (4.1). The idea behind the tempering methods presented herein is to sample from $\boldsymbol{\mu}^{\boldsymbol{y}}$ (as opposed to solely sampling from μ_1^y) by creating a Markov chain obtained from the successive application of two $\boldsymbol{\mu}^{\boldsymbol{y}}$-invariant Markov kernels $\mathbf{p}$ and $\mathbf{q}$, to some initial distribution $\boldsymbol{\nu}$, usually chosen to be the prior $\boldsymbol{\mu}_{\mathsf{pr}}$. Each kernel acts as follows. Given the current state $\boldsymbol{u}^n = (u_1^n, \dots, u_K^n)$, the kernel $\mathbf{p}$, which we will call the *standard MCMC kernel*, proposes a new, intermediate state $\tilde{\boldsymbol{u}}^{n+1} = (\tilde{u}_1^{n+1}, \dots, \tilde{u}_K^{n+1})$, possibly following the Metropolis-Hastings algorithm (or any other algorithm that generates a $\boldsymbol{\mu}$-invariant Markov operator). The Markov kernel $\mathbf{p}$ is a product kernel, meaning that each component $\tilde{u}_k^n$, $k = 1 \dots, K$, is generated independently of the others. Then, the *swapping kernel* $\mathbf{q}$ proposes a new state $\boldsymbol{u}^{n+1} = (u_1^{n+1}, \dots, u_K^{n+1})$ by introducing an "interaction" between the components of $\tilde{\boldsymbol{u}}^{(n+1)}$. This interaction step can be achieved, e.g., in the case of PT, by proposing to swap two components at two consecutive temperatures, i.e., components k and $k+1$, and accepting this swap with a certain probability given by the usual Metropolis-Hastings acceptance-rejection rule. In general, the swapping kernel is applied every N_s steps of the chain, for some $N_s \geq 1$. We will devote the following subsection to the construction of the swapping kernel $\mathbf{q}$.

4.3.1 The swapping kernel $\mathbf{q}$

Define $\mathscr{S}_K$ as the collection of all the bijective maps from $\{1, 2, \dots, K\}$ to itself, i.e., the set of all $K!$ possible permutations of $\mathrm{id} := \{1, \dots, K\}$. Let $\sigma \in \mathscr{S}_K$ be a permutation, and define the swapped state $\boldsymbol{u}_\sigma := (u_{\sigma(1)}, \dots, u_{\sigma(K)})$, and the inverse permutation $\sigma^{-1} \in \mathscr{S}_K$ such that $\sigma \circ \sigma^{-1} = \sigma^{-1} \circ \sigma = \mathrm{id}$. In addition, let $S_K \subseteq \mathscr{S}_K$ be any subset of $\mathscr{S}_K$ closed with respect to inversion, i.e., $\sigma \in S_K \implies \sigma^{-1} \in S_K$. We denote the cardinality of S_K by $|S_K|$.

Example 4.3.1: *As a simple example, consider a Standard PT as in Algorithm 4 with $K = 4$. In this case, we attempt to swap two contiguous temperatures T_i and T_{i+1}, $i = 1, 2, 3$. Thus, S_K is the set of permutations $\{\sigma_{1,2}, \sigma_{2,3}, \sigma_{3,4}\}$ with:*

$$\begin{aligned} \sigma_{1,2} &= (2, 1, 3, 4), \\ \sigma_{2,3} &= (1, 3, 2, 4), \\ \sigma_{3,4} &= (1, 2, 4, 3). \end{aligned}$$

Notice that each permutation is its own inverse; for example:

$$\sigma_{1,2}(\sigma_{1,2}) = \sigma_{1,2}((2,1,3,4)) = (1,2,3,4) = \mathrm{id}.$$

To define the swapping kernel $\mathbf{q}$, we first need to define the swapping ratio and swapping acceptance probability.

Definition 4.3.1 (Swapping ratio): *We say that a function* $r : \mathsf{X}^K \times S_K \mapsto [0,1]$ *is a* swapping ratio *if it satisfies the following two conditions:*

1. $\forall \boldsymbol{u} \in \mathsf{X}^K$, $r(\boldsymbol{u}, \cdot)$ *is a probability mass function on* S_K.

2. $\forall \sigma \in S_K$, $r(\cdot, \sigma)$ *is measurable on* $(\mathsf{X}^K, \mathcal{B}^K)$.

Definition 4.3.2 (Swapping acceptance probability): *Let* $\boldsymbol{u} \in \mathsf{X}^K$ *and* $\sigma,\ \sigma^{-1} \in S_K$. *We call* swapping acceptance probability *the function* $\alpha_{\text{swap}} : \mathsf{X}^K \times S_K \mapsto [0,1]$ *defined as*

$$\alpha_{\text{swap}}(\boldsymbol{u}, \sigma) = \begin{cases} \min\left\{1, \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma) r(\boldsymbol{u}_\sigma, \sigma^{-1})}{\boldsymbol{\pi}^y(\boldsymbol{u}) r(\boldsymbol{u}, \sigma)}\right\}, & \text{if } r(\boldsymbol{u}, \sigma) > 0, \\ 0 & \text{if } r(\boldsymbol{u}, \sigma) = 0. \end{cases}$$

We can now define the swapping kernel $\mathbf{q}$.

Definition 4.3.3 (Swapping kernel): *Given a swapping ratio* $r : \mathsf{X}^K \times S_K \mapsto [0,1]$ *and its associated swapping acceptance probability* $\alpha_{\text{swap}} : \mathsf{X}^K \times S_K \mapsto [0,1]$, *we define the* swapping Markov kernel $\mathbf{q} : \mathsf{X}^K \times \mathcal{B}^K \mapsto [0,1]$ *as*

$$\begin{aligned} \mathbf{q}(\boldsymbol{u}, B) = \sum_{\sigma \in S_K} r(\boldsymbol{u}, \sigma) \big[& (1 - \alpha_{\text{swap}}(\boldsymbol{u}, \sigma)) \delta_{\boldsymbol{u}}(B) \\ & + \alpha_{\text{swap}}(\boldsymbol{u}, \sigma) \delta_{\boldsymbol{u}_\sigma}(B) \big], \qquad \boldsymbol{u} \in \mathsf{X}^K,\ B \in \mathcal{B}^K, \end{aligned} \tag{4.3}$$

where $\delta_{\boldsymbol{u}}(B)$ *denotes the Dirac measure in* $\boldsymbol{u}$, *i.e.,* $\delta_{\boldsymbol{u}}(B) = 1$ *if* $\boldsymbol{u} \in B$ *and* 0 *otherwise.*

The swapping mechanism should be understood in the following way: given a current state of the chain $\boldsymbol{u} \in \mathsf{X}^K$, the swapping kernel samples a permutation σ from S_K with probability $r(\boldsymbol{u}, \sigma)$ and generates $\boldsymbol{u}_\sigma$. This permuted state is then accepted as the new state of the chain with probability $\alpha_{\text{swap}}(\boldsymbol{u}, \sigma)$. Notice that the swapping kernel follows a Metropolis-Hastings-like procedure with "proposal" distribution $r(\boldsymbol{u}, \sigma)$ and acceptance probability $\alpha_{\text{swap}}(\boldsymbol{u}, \sigma)$. Moreover, as detailed in the next proposition, such a kernel is reversible with respect to $\boldsymbol{\mu}$, since it is a Metropolis-Hastings type kernel.

Proposition 4.3.1: *The Markov kernel* $\mathbf{q}$ *defined in (4.3) is reversible with respect to the product measure* $\boldsymbol{\mu}$ *defined in (4.2).*

Proof. Let $A, B \in \mathcal{B}^K$. We want to show that

$$\int_A q(\boldsymbol{u}, B)\boldsymbol{\mu}(\mathrm{d}\boldsymbol{u}) = \int_B q(\boldsymbol{u}, A)\boldsymbol{\mu}(\mathrm{d}\boldsymbol{u}).$$

Thus,

$$\int_A q(\boldsymbol{u}, B)\boldsymbol{\mu}(\mathrm{d}\boldsymbol{u}) = \underbrace{\sum_{\sigma\in S_K}\int_A r(\boldsymbol{u},\sigma)\alpha_{\text{swap}}(\boldsymbol{u},\sigma)\delta_{\boldsymbol{u}_\sigma}(B)\boldsymbol{\pi}^y(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})}_{I}$$
$$+ \underbrace{\sum_{\sigma\in S_K}\int_A r(\boldsymbol{u},\sigma)\left(1-\alpha_{\text{swap}}(\boldsymbol{u},\sigma)\right)\delta_{\boldsymbol{u}}(B)\boldsymbol{\pi}^y(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})}_{II}.$$

Let $A_\sigma := \{\boldsymbol{z} \in \mathsf{X}^K : \boldsymbol{z}_{\sigma^{-1}} \in A\}$, and, for notational simplicity, write $\min\{a, b\} = \{a \wedge b\}$, $a, b \in \mathbb{R}$. From I, we have:

$$\begin{aligned} I &= \sum_{\sigma\in S_K}\int_A \left\{1 \wedge \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)r(\boldsymbol{u}_\sigma,\sigma^{-1})}{\boldsymbol{\pi}^y(\boldsymbol{u})r(\boldsymbol{u},\sigma)}\right\} r(\boldsymbol{u},\sigma)\boldsymbol{\pi}^y(\boldsymbol{u})\delta_{\boldsymbol{u}_\sigma}(B)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) \\ &= \sum_{\sigma\in S_K}\int_A \left\{1 \wedge \frac{\boldsymbol{\pi}^y(\boldsymbol{u})r(\boldsymbol{u},\sigma)}{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)r(\boldsymbol{u}_\sigma,\sigma^{-1})}\right\} r(\boldsymbol{u}_\sigma,\sigma^{-1})\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)\delta_{\boldsymbol{u}_\sigma}(B)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}). \end{aligned}$$

Then, noticing that $\boldsymbol{\mu}_{\mathsf{pr}}$ is permutation invariant, we get

$$\begin{aligned} I &= \sum_{\sigma\in S_K}\int_{A_\sigma} \left\{1 \wedge \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma^{-1}})r(\boldsymbol{u}_{\sigma^{-1}},\sigma)}{\boldsymbol{\pi}^y(\boldsymbol{u})r(\boldsymbol{u},\sigma^{-1})}\right\} \\ &\qquad \times r(\boldsymbol{u},\sigma^{-1})\boldsymbol{\pi}^y(\boldsymbol{u})\delta_{\boldsymbol{u}}(B)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) \\ &= \sum_{\sigma\in S_K}\int_{A_\sigma\cap B} \left\{1 \wedge \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma^{-1}})r(\boldsymbol{u}_{\sigma^{-1}},\sigma)}{\boldsymbol{\pi}^y(\boldsymbol{u})r(\boldsymbol{u},\sigma^{-1})}\right\} \\ &\qquad \times r(\boldsymbol{u},\sigma^{-1})\boldsymbol{\pi}^y(\boldsymbol{u})\delta_{\boldsymbol{u}}(B)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) \\ &= \sum_{\sigma\in S_K}\int_{B} \left\{1 \wedge \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma^{-1}})r(\boldsymbol{u}_{\sigma^{-1}},\sigma)}{\boldsymbol{\pi}^y(\boldsymbol{u})r(\boldsymbol{u},\sigma^{-1})}\right\} \\ &\qquad \times r(\boldsymbol{u},\sigma^{-1})\boldsymbol{\pi}^y(\boldsymbol{u})\delta_{\boldsymbol{u}}(A_\sigma)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) \\ &= \sum_{\sigma\in S_K}\int_{B} \left\{1 \wedge \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma^{-1}})r(\boldsymbol{u}_{\sigma^{-1}},\sigma)}{\boldsymbol{\pi}^y(\boldsymbol{u})r(\boldsymbol{u},\sigma^{-1})}\right\} \\ &\qquad \times r(\boldsymbol{u},\sigma^{-1})\boldsymbol{\pi}^y(\boldsymbol{u})\delta_{\boldsymbol{u}_{\sigma^{-1}}}(A)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) \end{aligned}$$

$$= \sum_{\sigma \in S_K} \int_B r(\boldsymbol{u}, \sigma^{-1}) \boldsymbol{\pi}^y(\boldsymbol{u}) \alpha_{\text{swap}}(\boldsymbol{u}, \sigma^{-1}) \delta_{\boldsymbol{u}_{\sigma^{-1}}}(A) \boldsymbol{\mu}_{\text{pr}}(\mathrm{d}\boldsymbol{u})$$
$$= \sum_{\sigma \in S_K} \int_B r(\boldsymbol{u}, \sigma) \boldsymbol{\pi}^y(\boldsymbol{u}) \alpha_{\text{swap}}(\boldsymbol{u}, \sigma) \delta_{\boldsymbol{u}_\sigma}(A) \boldsymbol{\mu}_{\text{pr}}(\mathrm{d}\boldsymbol{u}).$$

For the second term II we simply have

$$II = \sum_{\sigma \in S_K} \int_A r(\boldsymbol{u}, \sigma)(1 - \alpha_{\text{swap}}(\boldsymbol{u}, \sigma)) \delta_{\boldsymbol{u}}(B) \boldsymbol{\pi}^y(\boldsymbol{u}) \boldsymbol{\mu}_{\text{pr}}(\mathrm{d}\boldsymbol{u})$$
$$= \sum_{\sigma \in S_K} \int_{A \cap B} r(\boldsymbol{u}, \sigma)(1 - \alpha_{\text{swap}}(\boldsymbol{u}, \sigma)) \delta_{\boldsymbol{u}}(B) \boldsymbol{\pi}^y(\boldsymbol{u}) \boldsymbol{\mu}_{\text{pr}}(\mathrm{d}\boldsymbol{u})$$
$$= \sum_{\sigma \in S_K} \int_B r(\boldsymbol{u}, \sigma)(1 - \alpha_{\text{swap}}(\boldsymbol{u}, \sigma)) \delta_{\boldsymbol{u}}(A) \boldsymbol{\pi}^y(\boldsymbol{u}) \boldsymbol{\mu}_{\text{pr}}(\mathrm{d}\boldsymbol{u}).$$

□

This generic form of the swapping kernel provides the foundation for both PT and GPT. We describe these algorithms in the following subsections.

4.3.2 The Parallel Tempering case

We first show how a PT algorithm that only swaps states between the i^{th} and j^{th} components of the chain can be cast in the general framework presented above. To that end, let $\sigma_{i,j}$ be the permutation of $(1, 2, \ldots, K)$, which only permutes the i^{th} and j^{th} components, while leaving the other components invariant (i.e., such that $\sigma(i) = j, \sigma(j) = i$, and $\sigma(k) = k, k \neq i, k \neq j$). We can take $S_K = \{\sigma_{i,j},\ i, j = 1, \ldots, K\}$ and define the PT swapping ratio between components i and j by $r_{i,j}^{(\text{PT})} : \mathsf{X}^K \times S_K \mapsto [0, 1]$ as

$$r_{i,j}^{(\text{PT})}(\boldsymbol{u}, \sigma) := \begin{cases} 1 & \text{if } \sigma = \sigma_{i,j}, \\ 0 & \text{otherwise.} \end{cases}$$

Notice that this implies that $r_{i,j}^{(\text{PT})}(\boldsymbol{u}_\sigma, \sigma^{-1}) = r_{i,j}^{(\text{PT})}(\boldsymbol{u}, \sigma)$ since $\sigma_{i,j}^{-1} = \sigma_{i,j}$ and $r_{i,j}^{(\text{PT})}$ does not depend on $\boldsymbol{u}$, which in turn leads to the swapping acceptance probability $\alpha_{\text{swap}}^{(\text{PT})} : \mathsf{X}^K \times S_K \mapsto [0, 1]$ defined as:

$$\alpha_{\text{swap}}^{(\text{PT})}(\boldsymbol{u}, \sigma_{i,j}) := \min\left\{1, \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma_{i,j}})}{\boldsymbol{\pi}^y(\boldsymbol{u})}\right\},$$
$$\alpha_{\text{swap}}^{(\text{PT})}(\boldsymbol{u}, \sigma) = 0,\ \sigma \neq \sigma_{i,j}.$$

Thus, we can define the swapping kernel for the Parallel Tempering algorithm that swaps components i and j as follows:

Definition 4.3.4 (Pairwise Parallel Tempering swapping kernel): *Let $\boldsymbol{u} \in \mathsf{X}^K$, $\sigma_{i,j} \in S_K$. We define the* Parallel Tempering swapping kernel*, which proposes to swap states between the i^{th} and j^{th} chains as $\mathbf{q}_{i,j}^{(\text{PT})} : \mathsf{X}^K \times \mathcal{B}^K \mapsto [0,1]$ given by*

$$\begin{aligned}
\mathbf{q}_{i,j}^{(\text{PT})}(\boldsymbol{u}, B) &= \sum_{\sigma \in S_K} r_{i,j}^{(\text{PT})}(\boldsymbol{u}, \sigma) \Big((1 - \alpha_{\text{swap}}^{(\text{PT})}(\boldsymbol{u}, \sigma)) \delta_{\boldsymbol{u}}(B) \\
&+ \alpha_{\text{swap}}^{(\text{PT})}(\boldsymbol{u}, \sigma) \delta_{\boldsymbol{u}_\sigma}(B) \Big) \\
&= \left(1 - \min \left\{ 1, \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma_{i,j}})}{\boldsymbol{\pi}^y(\boldsymbol{u})} \right\} \delta_{\boldsymbol{u}}(B) \right) \\
&+ \min \left\{ 1, \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma_{i,j}})}{\boldsymbol{\pi}^y(\boldsymbol{u})} \right\} \delta_{\boldsymbol{u}_{\sigma_{i,j}}}(B), \quad \forall B \in \mathcal{B}^K.
\end{aligned}$$

In practice, however, the PT algorithm considers various sequential swaps between chains, which can be understood by applying the composition of kernels $\mathbf{q}_{i,j}^{(\text{PT})} \mathbf{q}_{k,\ell}^{(\text{PT})} \dots$ at every swapping step. In its most common form [21, 52, 114], the PT algorithm, hereafter referred to as standard PT (which on a slight abuse of notation we will denote by PT), proposes to swap states between chains at two consecutive temperatures. Its swapping kernel $\mathbf{q}^{(\text{PT})} : \mathsf{X}^K \times \mathcal{B}^K \mapsto [0,1]$ is given by

$$\mathbf{q}^{(\text{PT})} := \mathbf{q}_{1,2}^{(\text{PT})} \mathbf{q}_{2,3}^{(\text{PT})} \cdots \mathbf{q}_{K-1,K}^{(\text{PT})}.$$

Moreover, the algorithm described in [52], proposes to swap states every $N_s \geq 1$ steps of MCMC. The complete kernel for the PT kernel is then given by [21, 52, 114]

$$\mathbf{p}^{(\text{PT})} := \mathbf{q}_{1,2}^{(\text{PT})} \mathbf{q}_{2,3}^{(\text{PT})} \cdots \mathbf{q}_{K-1,K}^{(\text{PT})} \mathbf{p}^{N_s}, \tag{4.4}$$

where $\mathbf{p}$ is a standard reversible Markov transition kernel used to evolve the individual chains independently.

Remark 4.3.1: *Although the kernel $\mathbf{p}$ as well as each of the $\mathbf{q}_{i,i+1}$ are $\boldsymbol{\mu}$-reversible, notice that (4.4) does not have a palindromic structure, and as such it is not necessarily $\boldsymbol{\mu}$-reversible. One way of making the PT algorithm reversible with respect to $\boldsymbol{\mu}$ is to consider the palindromic form*

$$\begin{aligned}
&\mathbf{p}^{(\text{RPT})} := \\
&\left(\mathbf{q}_{1,2}^{(\text{PT})} \mathbf{q}_{2,3}^{(\text{PT})} \cdots \mathbf{q}_{K-1,K}^{(\text{PT})} \right) \mathbf{p}^{N_s} \left(\mathbf{q}_{K,K-1}^{(\text{PT})} \cdots \mathbf{q}_{3,2}^{(\text{PT})} \mathbf{q}_{2,1}^{(\text{PT})} \right),
\end{aligned}$$

where RPT stands for Reversible Parallel Tempering. In practice, there is not much difference between $\mathbf{p}^{(\text{RPT})}$ and $\mathbf{p}^{(\text{PT})}$, however, under the additional assumption of geometric ergodicity of the chain (c.f

Section 4.4) having a reversible kernel is useful to compute explicit error bounds on the non-asymptotic mean square error of an ergodic estimator [143].

4.3.3 Unweighted Generalized Parallel Tempering

The idea behind the Unweighted Generalized Parallel Tempering algorithm is to generalize PT so that (i) $N_s = 1$ provides a proper mixing of the chains, (ii) the algorithm is reversible with respect to $\boldsymbol{\mu}$, and (iii) the algorithm considers arbitrary sets S_K of swaps (always closed w.r.t inversion), instead of only pairwise swaps. We begin by constructing a kernel of the form (4.3). Let $r^{(\mathrm{UW})} : \mathsf{X}^K \times S_K \mapsto [0,1]$ be a function defined as

$$r^{(\mathrm{UW})}(\boldsymbol{u}, \sigma) := \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)}{\sum_{\sigma' \in S_K} \boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma'})}, \quad \boldsymbol{u} \in \mathsf{X}^K, \sigma \in S_K. \tag{4.5}$$

Clearly, (4.5) is a swapping ratio according to Definition 4.3.1. As such, given some state $\boldsymbol{u} \in \mathsf{X}^K$, $r^{(\mathrm{UW})}(\boldsymbol{u}, \sigma)$ assigns a state-dependent probability to each of the $|S_K|$ possible permutations in S_K. A permutation $\sigma \in S_K$ is then accepted with probability $\alpha^{(\mathrm{UW})}_{\mathrm{swap}}(\boldsymbol{u}, \sigma)$, given by

$$\alpha^{(\mathrm{UW})}_{\mathrm{swap}}(\boldsymbol{u}, \sigma) := \min\left\{1, \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma) r^{(\mathrm{UW})}(\boldsymbol{u}_\sigma, \sigma^{-1})}{\boldsymbol{\pi}^y(\boldsymbol{u}) r^{(\mathrm{UW})}(\boldsymbol{u}, \sigma)}\right\}. \tag{4.6}$$

Thus, we can define the swapping kernel for the UGPT algorithm, which takes the form of (4.3), with the particular choice of $r(\boldsymbol{u}, \sigma) = r^{(\mathrm{UW})}(\boldsymbol{u}, \sigma)$ and

$$\alpha_{\mathrm{swap}}(\boldsymbol{u}, \sigma) = \alpha^{(\mathrm{UW})}_{\mathrm{swap}}(\boldsymbol{u}, \sigma).$$

Notice that $\alpha^{(\mathrm{UW})}_{\mathrm{swap}}(\boldsymbol{u}, \sigma) = 1, \forall \sigma \in S_K$. Indeed, if we further examine Equation (4.6), we see that

$$\begin{aligned}\frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma) r^{(\mathrm{UW})}(\boldsymbol{u}_\sigma, \sigma^{-1})}{\boldsymbol{\pi}^y(\boldsymbol{u}) r^{(\mathrm{UW})}(\boldsymbol{u}, \sigma)} &= \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)}{\boldsymbol{\pi}^y(\boldsymbol{u})} \cdot \frac{\boldsymbol{\pi}^y(\boldsymbol{u})}{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)} \cdot \frac{\sum_{\sigma'} \boldsymbol{\pi}^y(\boldsymbol{u}_{\sigma'})}{\sum_{\hat{\sigma}} \boldsymbol{\pi}^y(\boldsymbol{u}_{\hat{\sigma}})} \\ &= \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)}{\boldsymbol{\pi}^y(\boldsymbol{u})} \cdot \frac{\boldsymbol{\pi}^y(\boldsymbol{u})}{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)} = 1.\end{aligned}$$

In practice, this means that the proposed permuted state is always accepted with probability 1. The expression of the UGPT kernel then simplifies as follows.

Definition 4.3.5 (unweighted swapping kernel): *The* unweighted swapping kernel $\mathbf{q}^{(\mathrm{UW})} : \mathsf{X}^K \times \mathcal{B}^K \mapsto [0,1]$ *is defined as*

$$\mathbf{q}^{(\mathrm{UW})}(\boldsymbol{u}, B) = \sum_{\sigma \in S_K} r^{(\mathrm{UW})}(\boldsymbol{u}, \sigma) \delta_{\boldsymbol{u}_\sigma}(B),$$

$\forall \boldsymbol{u} \in \mathsf{X}^K,\ B \in \mathcal{B}^K$. Applying this swapping kernel successively with the kernel $\mathbf{p} = p_1 \times p_2 \times \ldots p_K$ in the order $\mathbf{q}^{(\mathrm{UW})}\mathbf{p}\mathbf{q}^{(\mathrm{UW})} =: \mathbf{p}^{(\mathrm{UW})}$ gives what we call *Unweighted Generalized Parallel Tempering kernel* $\mathbf{p}^{(\mathrm{UW})}$. Lastly, we write the UGPT in operator form as

$$\mathbf{P}^{(\mathrm{UW})} := \mathbf{Q}^{(\mathrm{UW})}\mathbf{P}\mathbf{Q}^{(\mathrm{UW})},$$

where $\mathbf{P}$ and $\mathbf{Q}^{(\mathrm{UW})}$ are the Markov operators corresponding to the kernels $\mathbf{p}$ and $\mathbf{q}^{(\mathrm{UW})}$, respectively. We now investigate the reversibility of the UGPT kernel. We start with a rather straightforward result.

Proposition 4.3.2: *Suppose that, for any $k = 1, 2, \ldots, K$, p_k is μ_k-reversible. Then, $\mathbf{p} = p_1 \times \cdots \times p_K$ is reversible with respect to $\boldsymbol{\mu}$.*

Proof. We prove reversibility by confirming that equation (3.5) holds true. To that end, let $\boldsymbol{u} \in \mathsf{X}^K,\ A, B \in \mathcal{B}^K$, where A and B tensorize, i.e., $A := \prod_{k=1}^K A_k$ and $B := \prod_{k=1}^K B_k$, with $A_1, \ldots, A_K, B_1, \ldots, B_K \in \mathcal{B}(\mathsf{X})$. Then,

$$\begin{aligned}
\int_A \boldsymbol{\pi}^y(\boldsymbol{u})\mathbf{p}(\boldsymbol{u}, B)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) &= \prod_{k=1}^K \int_{A_k} \pi^y(u_k)p(u_k, B_k)\mu_{\mathsf{pr}}(\mathrm{d}u_k) \\
&= \prod_{k=1}^K \int_{B_k} \pi^y(u_k)p(u_k, A_k)\mu_{\mathsf{pr}}(\mathrm{d}u_k) \\
&= \int_B \boldsymbol{\pi}^y(\boldsymbol{u})\mathbf{p}(\boldsymbol{u}, A)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}).
\end{aligned}$$

Showing that the previous equality holds for sets A, B that tensorize is indeed sufficient to show that the claim holds for any $A, B \in \mathcal{B}^K$. This follows from Carathéodory's Extension Theorem applied as in the proof of uniqueness of product measures; see [2, §1.3.10, 2.6.3], for details. □

We can now prove the reversibility of the chain generated by $\mathbf{p}^{(\mathrm{UW})}$.

Proposition 4.3.3 (Reversibility of the UGPT chain): *Suppose that, for any $k = 1, 2, \ldots, K$, p_k is μ_k-reversible. Then, the Markov chain generated by $\mathbf{p}^{(\mathrm{UW})}$ is $\boldsymbol{\mu}$-reversible.*

Proof. It follows from Proposition 4.3.1 and 4.3.2 that the kernels $\mathbf{q}^{(\mathrm{UW})}$ and $\mathbf{p}$ are $\boldsymbol{\mu}$-reversible. Furthermore, since $\mathbf{p}^{(\mathrm{UW})}$ is a *palindromic* composition of kernels, each of which is reversible with respect to $\boldsymbol{\mu}$, then, $\mathbf{p}^{(\mathrm{UW})}$ is reversible with respect to $\boldsymbol{\mu}$ [21].

□

The UGPT algorithm proceeds by iteratively applying the kernel $\mathbf{p}^{(\mathrm{UW})}$ to a predefined initial state. In particular, states are updated using the procedure outlined in Algorithm 5.

Remark 4.3.2: *In practice, one does not need to perform $|S_K|$ posterior evaluations when computing $r^{(\mathrm{UW})}(\boldsymbol{u}^n, \cdot)$, rather "just" K of them. Indeed, since $\pi_j^y(u_k^n) \propto \pi^y(u_k)^{T_j}$, $k, j = 1, 2, \ldots, K$,*

Algorithm 5 Unweighted Generalized Parallel Tempering.

function GENERALIZED PARALLEL TEMPERING($\mathbf{p}, N, \boldsymbol{\nu}$)
  Sample $\boldsymbol{u}^{(1)} \sim \boldsymbol{\nu}$
  for $n = 1, 2, \dots, N-1$ **do**
    # First swapping kernel
    Sample $\boldsymbol{\theta}_\sigma^{(n)} \sim \mathbf{q}^{(\mathrm{UW})}(\boldsymbol{u}^{(n)}, \cdot)$
    # Markov transition kernel $\mathbf{p}$
    Sample $\boldsymbol{z}^{(n+1)} \sim \mathbf{p}(\boldsymbol{\theta}_\sigma^{(n)}, \cdot)$ kernel
    # Second swapping kernel
    Sample $\boldsymbol{u}^{(n+1)} \sim \mathbf{q}^{(\mathrm{UW})}(\boldsymbol{z}^{(n+1)}, \cdot)$
  end for
  Output $\{\theta_1^{(n)}\}_{n=1}^N$
end function

we just need to store the values of $\pi^y(u_k^n)$, $k = 1, 2, \dots, K$, *for a fixed* n, *and then permute over the temperature indices.*

Let now $\mathsf{QoI} : \mathsf{X} \mapsto \mathbb{R}$ be a quantity of interest. The posterior mean of QoI, $\mu^y(\mathsf{QoI}) := \mu_1^y(\mathsf{QoI})$ is approximated using $N \in \mathbb{N}$ samples by the following ergodic estimator $\widehat{\mathsf{QoI}}_{(\mathrm{UW})}$:

$$\mu^y(\mathsf{QoI}) \approx \widehat{\mathsf{QoI}}_{(\mathrm{UW})} = \frac{1}{N-b} \sum_{n=b}^{N} \mathsf{QoI}(u_1^{(n)}).$$

A comment on the pairwise state-dependent PT method of [90]

The work [90] presents a similar state-dependent swapping. We will refer to the method presented therein as Pairwise State Dependent Parallel Tempering (PSDPT). Such a method, however, differs from UGPT from the fact that (i) only pairwise swaps are considered and (ii) it is not rejection free. We summarize such a method for the sake of completeness. Let $S_{K,\mathrm{pairwise}}$ denote the group of pairwise permutations of $(1, 2, \dots, K)$. Given a current state $\boldsymbol{u} \in \mathsf{X}^K$, the PSDPT algorithm samples a pairwise permutation $\boldsymbol{u}_{\sigma_{i,j}} \in S_{K,\mathrm{pairwise}}$ with probability $r_{i,j}^{(\mathrm{PSDPT})}(\boldsymbol{u}, \sigma_{i,j})$ given by

$$r_{i,j}^{(\mathrm{PSDPT})}(\boldsymbol{u}, \sigma_{i,j}) := \frac{\exp(-|\Phi(u_i, y) - \Phi(u_j; y)|)}{\sum_{k,l} \exp(-|\Phi(u_k, y) - \Phi(u_l; y)|)},$$

and then accepts this swap with probability

$$\alpha_{\mathrm{swap}}^{(\mathrm{PSDPT})}(\boldsymbol{u}, \sigma_{i,j}) := \min\left\{1, \left(\frac{\pi_1^y(u_i)}{\pi_1^y(u_j)}\right)^{\frac{1}{T_j} - \frac{1}{T_i}}\right\}.$$

This method is attractive from an implementation point of view in the sense that it promotes pairwise swaps that have a similar *energy*, and as such, are *likely* (yet not guaranteed) to get accepted.

In contrast, UGPT *always* accepts the new proposed state, which in turn leads to a larger amount of *global* moves, thus providing a more efficient algorithm. This is verified on the numerical experiments.

4.3.4 Weighted Generalized Parallel Tempering

Following the intuition of the continuous-time Infinite Swapping approach of [49, 133], we propose a second discrete-time algorithm, which we will refer to as *Weighted Generalized Parallel Tempering* (WGPT). The idea behind this method is to swap the dynamics of the process, that is, the Markov kernels and temperatures, instead of swapping the states such that any given swap is accepted with probability 1. We will see that the Markov kernel obtained when swapping the dynamics is not invariant with respect to the product measure of interest $\boldsymbol{\mu}$; therefore, an importance sampling step is needed when computing posterior expectations.
For a given permutation $\sigma \in S_K$, we define the *swapped Markov kernel* $\mathbf{p}_\sigma : \mathsf{X}^K \times \mathcal{B}^K \mapsto [0,1]$ and the *swapped product posterior measure* $\boldsymbol{\mu}_\sigma$ (on the measurable space $(\mathsf{X}^K, \mathcal{B}^K)$) as:

$$\begin{aligned}\mathbf{p}_\sigma(\boldsymbol{u}, \cdot) &= p_{\sigma(1)}(\theta_1, \cdot) \times \cdots \times p_{\sigma(K)}(\theta_K, \cdot),\\ \boldsymbol{\mu}^y_\sigma &:= \mu_{\sigma(1)} \times \cdots \times \mu_{\sigma(K)},\end{aligned}$$

where the swapped posterior measure has a density with respect to $\boldsymbol{\mu}_{\text{prior}}$ given by

$$\boldsymbol{\pi}^{\boldsymbol{y}}_\sigma(\boldsymbol{u}) := \pi^y_{\sigma(1)}(u_1) \times \cdots \times \pi^y_{\sigma(K)}(u_K), \quad u \in \mathsf{X}^K, \sigma \in S_K \tag{4.7}$$

Moreover, we define the swapping weights

$$w_\sigma(\boldsymbol{u}) := \frac{\boldsymbol{\pi}^y_\sigma(\boldsymbol{u})}{\sum_{\sigma' \in S_K} \boldsymbol{\pi}^y_{\sigma'}(\boldsymbol{u})}, \quad \boldsymbol{u} \in \mathsf{X}^K,\ \sigma \in S_K. \tag{4.8}$$

Note that, in general, $\boldsymbol{\pi}^y_\sigma(\boldsymbol{u}) \neq \boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)$ (however $\boldsymbol{\pi}^y_{\sigma^{-1}}(\boldsymbol{u}) = \boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)$), and as such, $w_\sigma(\boldsymbol{u}) \neq r^{(\mathrm{UW})}(\boldsymbol{u}, \sigma)$, with w_σ defined as in (4.8).

Definition 4.3.6: *We define the* Weighted Generalized Parallel Tempering *kernel* $\mathbf{p}^{(\mathrm{W})} : \mathsf{X}^K \times B^K \mapsto [0,1]$ *as the following state-dependent, convex combination of kernels:*

$$\mathbf{p}^{(\mathrm{W})}(\boldsymbol{u}, \cdot) := \sum_{\sigma \in S_K} w_\sigma(\boldsymbol{u}) \mathbf{p}_\sigma(\boldsymbol{u}, \cdot), \quad \boldsymbol{u} \in \mathsf{X}^K,\ \sigma \in S_K.$$

Thus, the WGPT chain is obtained by iteratively applying $\mathbf{p}^{(\mathrm{W})}$. We show in proposition 4.3.4 that the resulting Markov chain has invariant measure

$$\boldsymbol{\mu}^y_{\mathrm{W}} = \frac{1}{|S_K|} \sum_{\sigma \in S_K} \boldsymbol{\mu}^y_\sigma = \tilde{\mu} \times \cdots \times \tilde{\mu},$$

with $\tilde{\mu^y} = \frac{1}{|S_K|}\sum_\sigma \mu^y_\sigma$, i.e., the average with tensorization. Furthermore, $\boldsymbol{\mu}^y_{\mathrm{W}}$ has a density (w.r.t the tensorized prior $\boldsymbol{\mu}_{\mathsf{pr}}$) given by

$$\boldsymbol{\pi}^{\boldsymbol{y}}_{\mathrm{W}}(\boldsymbol{u}) = \frac{1}{|S_K|}\sum_{\sigma\in S_K}\boldsymbol{\pi}^y_\sigma(\boldsymbol{u}), \quad \boldsymbol{u}\in \mathsf{X}^K,$$

and a similar average and then tensorization representation applies to $\boldsymbol{\pi}^y_{\mathrm{W}}$. We now proceed to show that $\mathbf{p}^{(\mathrm{W})}(\boldsymbol{u},\cdot)$ is $\boldsymbol{\mu}^y_{\mathrm{W}}$-reversible (hence $\boldsymbol{\mu}^y_{\mathrm{W}}$-invariant).

Proposition 4.3.4 (Reversibility of the WGPT chain): *Suppose that, for any $k = 1, 2, \ldots, K$ p_k is μ_k-reversible. Then, the Markov chain generated by $\mathbf{p}^{(\mathrm{W})}$ is $\boldsymbol{\mu}^y_{\mathrm{W}}$-reversible.*

Proof. We show reversibility by showing that (3.5) holds true. Thus, for $\boldsymbol{u}\in\mathsf{X}^K$, $A, B\in\mathcal{B}^K$, with $A := A_1\times\cdots\times A_K$, $A_k\in\mathcal{B}(\mathsf{X})$, and with B_k defined in a similar way, we have that:

$$\begin{aligned}
&\int_A \mathbf{p}^{(\mathrm{W})}(\boldsymbol{u},B)\boldsymbol{\pi}^{\boldsymbol{y}}_{\mathrm{W}}(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\\
&= \int_A \left[\sum_{\sigma\in S_K} w_\sigma(\boldsymbol{u})\mathbf{p}_\sigma(\boldsymbol{u},B)\right]\frac{\sum_{\rho\in S_K}\boldsymbol{\pi}^y_\rho(\boldsymbol{u})}{|S_K|}\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\\
&= \int_A \left[\sum_{\sigma\in S_K}\frac{\boldsymbol{\pi}^{\boldsymbol{y}}_\sigma(\boldsymbol{u})}{\sum_{\sigma'\in S_K}\boldsymbol{\pi}^y_{\sigma'}(\boldsymbol{u})}\mathbf{p}_\sigma(\boldsymbol{u},B)\right]\\
&\times\frac{\sum_{\rho\in S_K}\boldsymbol{\pi}^y_\rho(\boldsymbol{u})}{|S_K|}\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\\
&= \frac{1}{|S_K|}\sum_{\sigma\in S_K}\int_A \boldsymbol{\pi}^y_\sigma(\boldsymbol{u})\mathbf{p}_\sigma(\boldsymbol{u},B)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) = I.
\end{aligned}$$

From proposition 4.3.2, and multiplying and dividing by $\sum_{\rho\in S_K}\boldsymbol{\pi}^y_\rho(\boldsymbol{u})$ we obtain

$$\begin{aligned}
I &= \frac{1}{|S_K|}\sum_{\sigma\in S_K}\int_B \boldsymbol{\pi}^y_\sigma(\boldsymbol{u})\mathbf{p}_\sigma(\boldsymbol{u},A)\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) \quad \text{(by Prop. 4.3.2)}\\
&= \frac{1}{|S_K|}\sum_{\sigma\in S_K}\int_B \frac{\boldsymbol{\pi}^y_\sigma(\boldsymbol{u})\mathbf{p}_\sigma(\boldsymbol{u},A)}{\sum_{\sigma'\in S_K}\boldsymbol{\pi}^y_{\sigma'}(\boldsymbol{u})}\sum_{\rho\in S_K}\boldsymbol{\pi}^y_\rho(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\\
&= \sum_{\sigma\in S_K}\int_B w_\sigma(\boldsymbol{u})\mathbf{p}_\sigma(\boldsymbol{u},A)\boldsymbol{\pi}^{\boldsymbol{y}}_{\mathrm{W}}(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\\
&= \int_B \mathbf{p}^{(\mathrm{W})}(\boldsymbol{u},A)\boldsymbol{\pi}^{\boldsymbol{y}}_{\mathrm{W}}(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}).
\end{aligned}$$

where once again, in light of Carathéodory's Extension Theorem, it is sufficient to show that reversibility holds for sets that tensorize. □

We remark that the measure $\boldsymbol{\mu}^y_{\mathbb{W}}$ is not of interest per se. However, we can use importance sampling to compute posterior expectations. Let $\mathsf{QoI}(\boldsymbol{u}) := \mathsf{QoI}(u_1)$ be a $\boldsymbol{\mu}$-integrable quantity of interest. We can write

$$\begin{aligned}\mathbb{E}_{\mu_1}[\mathsf{QoI}] = \mathbb{E}_{\boldsymbol{\mu}}[\mathsf{QoI}(u_1)] &= \mathbb{E}_{\boldsymbol{\mu}_{\mathbb{W}}}\left[\mathsf{QoI}(u_1)\frac{\boldsymbol{\pi}^y(\boldsymbol{u})}{\boldsymbol{\pi}^y_{\mathbb{W}}(\boldsymbol{u})}\right] \\ &= \frac{1}{|S_K|}\sum_{\sigma\in S_K}\mathbb{E}_{\boldsymbol{\mu}_{\mathbb{W}}}\left[\mathsf{QoI}(u_{\sigma(1)})\frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)}{\boldsymbol{\pi}^y_{\mathbb{W}}(\boldsymbol{u}_\sigma)}\right].\end{aligned}$$

The last equality can be justified since $\boldsymbol{\mu}^y_{\mathbb{W}}$ is invariant by permutation of coordinates. Thus, we can define the following (weighted) ergodic estimator $\widehat{\mathsf{QoI}}_{(\mathbb{W})}$ of the posterior mean of a quantity of interest QoI by

$$\begin{aligned}&\mu(\mathsf{QoI}) \approx \\ &\widehat{\mathsf{QoI}}_{(\mathbb{W})} = \frac{1}{|S_K|}\frac{1}{N}\sum_{\sigma\in S_K}\sum_{n=1}^{N}\frac{\boldsymbol{\pi}^y(\boldsymbol{u}^{(n)}_\sigma)}{\boldsymbol{\pi}^y_{\mathbb{W}}(\boldsymbol{u}^{(n)}_\sigma)}\mathsf{QoI}(u^{(n)}_{\sigma(1)}) \\ &= \frac{1}{|S_K|}\frac{1}{N}\sum_{\sigma\in S_K}\sum_{n=1}^{N}\widehat{w}(\boldsymbol{u}^{(n)},\sigma)\mathsf{QoI}(u^{(n)}_{\sigma(1)}),\end{aligned} \tag{4.9}$$

where we have denoted the importance sampling weights by $\widehat{w}(\boldsymbol{u},\sigma) := \frac{\boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)}{\boldsymbol{\pi}^y_{\mathbb{W}}(\boldsymbol{u}_\sigma)} = \frac{\mathrm{d}\boldsymbol{\mu}}{\mathrm{d}\boldsymbol{\mu}^y_{\mathbb{W}}}(\boldsymbol{u}_\sigma)$ and where N is the number of samples in the chain. Notice that $w(\boldsymbol{u},\sigma) = \widehat{w}(\boldsymbol{u},\sigma^{-1})$. As a result, the WGPT algorithm produces an estimator based on NK weighted samples, rather than "just" N, at the same computational cost of UGPT. Thus, the previous estimator evaluates the quantity of interest QoI not only in the points $\mathsf{QoI}(u^{(n)}_1)$, but also in all states of the parallel chains, $\mathsf{QoI}(u^{(n)}_{\sigma(1)})$ for all $\sigma \in S_K$, namely $\mathsf{QoI}(u^{(n)}_k),\ k = 1, 2, \ldots, K$.

Remark 4.3.3: *Although it is known that, in some cases, an importance sampling estimator can be negatively affected by the dimensionality of the parameter space* X *(see e.g., [3, Remark 1.17] or [122, Examples 9.1-9.3]), we argue that this is not the case for our estimator. Indeed, notice that the importance-sampling weights* $\widehat{w}(\boldsymbol{u},\sigma)$ *are always upper bounded by* $|S_K|$*, and do not blow up when the dimension goes to infinity. In Section 4.5.7 we present a numerical example on a high-dimensional problem. The results on that section evidence the robustness of WGPT with respect to the dimension of* u*.*

The Weighted Generalized Parallel Tempering procedure is shown in Algorithm 6. To reiterate, we remark that sampling from $\mathbf{p}_\sigma(\boldsymbol{u}^{(n)},\cdot)$ involves a swap of dynamics, i.e., kernels and temperatures. Just as in Remark 4.3.2, one only needs to evaluate the posterior K times (instead of $|S_K|$) to compute $w_{(\cdot)}(\boldsymbol{u}^n)$.

Algorithm 6 Weighted Generalized Parallel Tempering.

function WEIGHTED GENERALIZED PARALLEL TEMPERING($\mathbf{p}, N, \boldsymbol{\nu}$)
  Sample $\boldsymbol{u}^{(1)} \sim \boldsymbol{\nu}$
  for $n = 1, 2, \ldots, N-1$ **do**
    # Sample permutation σ with probability $w_\sigma(\boldsymbol{u}^n)$
    Sample $\sigma \sim \{w_{\sigma'}(\boldsymbol{u}^n)\}_{\sigma' \in S_K}$
    # Sample state with the swapped Markov kernel
    Sample $\boldsymbol{u}^{(n+1)} \sim \mathbf{p}_\sigma(\boldsymbol{u}^{(n)}, \cdot)$
  end for
  Output $\{\boldsymbol{u}^{(n)}\}_{n=1}^N, \{\{w_{\sigma'}(\boldsymbol{u}^n)\}_{\sigma' \in S_K}\}_{n=1}^N$.
end function

4.4 Ergodicity of Generalized Parallel Tempering

4.4.1 Geometric ergodicity and L_2-spectral gap for GPT

Our path to prove ergodicity of the GPT algorithms will be to show the existence of an L_2-spectral gap. The main results of this section are presented in Theorem 4.4.1 and Theorem 4.4.2, which show the existence of an L_2-spectral gap for both the UGPT and WGPT algorithms, respectively. We begin with the definition of overlap between two probability measures. Such a concept will later be used to bound the spectral gap of the GPT algorithms.

Definition 4.4.1 (Density overlap): *Let μ_k, μ_j be two probability measures on the measurable space $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$, each having respective densities $\pi_k(u), \pi_j(u),\ u \in \mathsf{X}$, with respect to some common reference measure ν_{X} also on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$. We define the* overlap *between $\pi_k(u)$ and $\pi_j(u)$ as*

$$\begin{aligned}\eta_{\nu_{\mathsf{X}}}(\pi_k, \pi_j) :&= \int_{\mathsf{X}} \min\{\pi_k(u), \pi_j(u)\}\nu_{\mathsf{X}}(\mathrm{du}) \\ &= 1 - \frac{1}{2}\left\|\mu_k - \mu_j\right\|_{L_1(\mathsf{X}, \nu_{\mathsf{X}})}.\end{aligned}$$

An analogous definition holds for $\boldsymbol{\pi}_\sigma, \boldsymbol{\pi}_\rho$, with $\rho, \sigma \in S_K$.

Assumption 4.4.1: *For $k = 1, \ldots, K$, let $\mu_k^y \in \mathcal{M}_1(\mathsf{X}, \mu_{\mathsf{pr}})$ be given as in (4.1), $p_k : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \mapsto [0, 1]$ be the Markov kernel associated to the k^{th} dynamics and let $P_k : L_r(\mathsf{X}, \mu_k^y) \mapsto L_r(\mathsf{X}, \mu_k^y)$ be its corresponding μ_k^y invariant Markov operator. In addition, for $\sigma, \rho \in S_K$, define the measures $\boldsymbol{\mu}_\sigma^y, \boldsymbol{\mu}_\rho^y \in \mathcal{M}(\mathsf{X}^K)$ as in Equation (4.2). Throughout this chapter it is assumed that:*

C1. The Markov kernel p_k is μ_k^y-reversible.

C2. The Markov operator P_k has an $L_2(\mathsf{X}, \mu_k^y)$ spectral gap.

C3. For any $\sigma, \rho \in S_K$, $\Lambda_{\sigma,\rho} := \eta_{\boldsymbol{\mu}_{\mathsf{pr}}}(\boldsymbol{\pi}_\sigma^y, \boldsymbol{\pi}_\rho^y) > 0$, with $\boldsymbol{\pi}_\sigma^y, \boldsymbol{\pi}_\rho^y$ defined as in (4.7).

These assumptions are relatively mild. In particular, C1 and C2 are known to hold for many commonly-used Markov transition kernels, such as RWM, Metropolis-adjusted Langevin Algo-

rithm, Hamiltonian Monte Carlo, (generalized) preconditioned Crank-Nicolson, among others, under mild regularity conditions on π^y [3, 65]. Assumption C3 holds true given the construction of the product measures in Section 4.3.

We now present an auxiliary result that we will use to bound the spectral gap of both the Weighted and Unweighted GPT algorithms.

Proposition 4.4.1: *Suppose that Assumption 4.4.1 holds and let* $\mathbf{P} := \bigotimes_{k=1}^{K} P_k : L_2(\mathsf{X}^K, \boldsymbol{\mu}^y) \mapsto L_2(\mathsf{X}^K, \boldsymbol{\mu}^y)$, *with invariant measure* $\boldsymbol{\mu}^y = \mu_1^y \times \cdots \times \mu_K^y$. *Then,* $\mathbf{P}$ *has an* $L_2(\mathsf{X}^K, \boldsymbol{\mu}^y)$*-spectral gap, i.e.,* $\|\mathbf{P}\|_{L_2^0(\mathsf{X}^K, \boldsymbol{\mu}^y) \mapsto L_2^0(\mathsf{X}^K, \boldsymbol{\mu}^y)} < 1$. *Moreover, the Markov chain obtained from* $\mathbf{P}$ *is* L_r *geometrically ergodic, for any* $r \in [1, \infty]$.

Proof. We limit ourselves to the case $K = 2$, since the case for $K > 2$ follows by induction. Denote by $I : L_2(\mathsf{X}, \mu_k^y) \mapsto L_2(\mathsf{X}, \mu_k^y), k = 1, 2$ the identity Markov transition operator, and let $f \in L_2(\mathsf{X}^2, \boldsymbol{\mu}^y)$. Notice that f admits a spectral representation in $L_2(\mathsf{X}^2, \boldsymbol{\mu}^y)$ given by $f(\boldsymbol{u}) = \sum_{k,j} \phi_k(u_1)\psi_j(u_2)c_{k,j}$, with $c_{k,j} \in \mathbb{R}$, and where $\{\phi_k\}_{i \in \mathbb{N}}$ is a complete orthonormal basis (CONB) of $L_2(\mathsf{X}, \mu_1^y)$ and $\{\psi_j\}_{j \in \mathbb{N}}$ is a CONB of $L_2(\mathsf{X}, \mu_2^y)$, so that $\{\phi_k \otimes \psi_j\}_{k,j \in \mathbb{N}}$ is a CONB of $L_2(\mathsf{X}^2, \boldsymbol{\mu}^y)$. Moreover, we assume that $\phi_0 = \psi_0 = 1$, and write, for notational simplicity $\|P_1\| = \|P_1\|_{L_2(\mathsf{X}, \mu_1^y) \mapsto L_2(\mathsf{X}, \mu_1^y)}$, and $\|P_2\| = \|P_2\|_{L_2(\mathsf{X}, \mu_2^y) \mapsto L_2(\mathsf{X}, \mu_2^y)}$. Lastly, denote $f_0 = f - c_{0,0}$, so that $f_0 \in L_2^0(\mathsf{X}^2, \boldsymbol{\mu}^y)$. Notice that

$$\begin{aligned} \|(P_1 \otimes I) f_0\|_{L_2(\mathsf{X}^2, \boldsymbol{\mu}^y)}^2 &= \left\| \sum_{(k,j) \neq (0,0)} (P_1 \phi_k)\, \psi_j c_{k,j} \right\|_{L_2(\mathsf{X}^2, \boldsymbol{\mu}^y)}^2 \\ &= \left\| \sum_{j=0}^{\infty} \left(\sum_{k=1}^{\infty} P_1 \phi_k c_{k,j} \right) \psi_j + \sum_{j=1}^{\infty} c_{0,j} P_1 \phi_0 \psi_j \right\|_{L_2(\mathsf{X}^2, \boldsymbol{\mu}^y)}^2 . \end{aligned} \tag{4.10}$$

Splitting the sum, we get from the orthonormality of the basis that:

$$
\begin{aligned}
(4.10) &= \sum_{j=1}^{\infty} \left\| \sum_{k=1}^{\infty} P_1 \phi_k c_{k,j} + c_{0,j} P_1 \phi_0 \right\|^2_{L_2(\mathsf{X},\mu_1^y)} \\
&+ \left\| \sum_{k=1}^{\infty} P_1 \phi_k c_{k,0} \right\|^2_{L_2(\mathsf{X},\mu_1^y)} \\
&= \sum_{j=1}^{\infty} \left\| P_1 \left(\sum_{k=1}^{\infty} \phi_k c_{k,j} \right) \right\|^2_{L_2(\mathsf{X},\mu_1^y)} + \sum_{j=1}^{\infty} \| c_{0,j} \phi_0 \|^2_{L_2(\mathsf{X},\mu_1^y)} \\
&+ \left\| P_1 \left(\sum_{k=1}^{\infty} \phi_k c_{k,0} \right) \right\|^2_{L_2(\mathsf{X},\mu_1^y)} \\
&\leq \sum_{j=1}^{\infty} \left(\|P_1\|^2 \sum_{k=1}^{\infty} c_{k,j}^2 + c_{0,j}^2 \right) + \|P_1\|^2 \sum_{k=1}^{\infty} c_{k,0}^2 \\
&= \|P_1\|^2 \, \|f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} + (1 - \|P_1\|^2) \sum_{j=1}^{\infty} (c_{0,j})^2 .
\end{aligned}
$$

Proceeding similarly, we can obtain an equivalent bound for $\|(I \otimes P_2) f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)}$. We are now ready to bound $\|\mathbf{P}\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y) \mapsto L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)}$:

$$
\begin{aligned}
&\|\mathbf{P} f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} = \|(P_1 \otimes P_2) f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} \\
&= \|(P_1 \otimes I)(I \otimes P_2) f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} \\
&\leq \|P_1\|^2 \, \|(I \otimes P_2) f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} \\
&+ (1 - \|P_1\|^2) \\
&\times \left(\sum_{j=1}^{\infty} \left((I \otimes P_2) \sum_{(\ell,k) \neq (0,0)} \langle c_{\ell,k} \phi_\ell \psi_k, \phi_0 \psi_j \rangle \right)^2 \right) \\
&= \|P_1\|^2 \, \|(I \otimes P_2) f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} \\
&+ (1 - \|P_1\|^2) \left(\sum_{j=1}^{\infty} \left(\sum_{k=1}^{\infty} \langle c_{0,k} (P_2 \psi_k), \psi_j \rangle \right)^2 \right) \\
&\leq \|P_1\|^2 \, \|(I \otimes P_2) f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} \\
&+ (1 - \|P_1\|^2) \left\| P_2 \left(\sum_{k=1}^{\infty} c_{0,k} \psi_k \right) \right\|^2_{L_2(\mathsf{X},\mu_2^y)}
\end{aligned}
$$

$$
\begin{aligned}
&\leq \|P_1\|^2 \|P_2\|^2 \|f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} \\
&+ \|P_1\|^2 (1 - \|P_2\|^2) \left(\sum_{j=1}^{\infty} c_{j,0}^2 \right) \\
&+ (1 - \|P_1\|^2) \|P_2\|^2 \left(\sum_{k=1}^{\infty} c_{0,k}^2 \right)
\end{aligned}
$$

Assuming without loss of generality that $\|P_1\| \geq \|P_2\|$, we can use the inequality above to bound

$$
\begin{aligned}
\|\mathbf{P} f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} &\leq \|P_1\|^2 \|P_2\|^2 \|f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} \\
&\quad + \|P_1\|^2 (1 - \|P_2\|^2) \underbrace{\left(\sum_{j=1}^{\infty} c_{j,0}^2 + \sum_{k=1}^{\infty} c_{0,k}^2 \right)}_{\leq \|f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)}} \\
&\leq \|P_1\|^2 \|f_0\|^2_{L_2(\mathsf{X}^2,\boldsymbol{\mu}^y)} .
\end{aligned}
$$

Thus, we have that

$$
\|\boldsymbol{P}\|_{L_2^0(\mathsf{X}^2,\boldsymbol{\mu}^y)\mapsto L_2^0(\mathsf{X}^2,\boldsymbol{\mu}^y)} \leq \max_{k=1,2} \{\|P_k\|_{L_2^0(\mathsf{X},\mu_k^y)\mapsto L_2^0(\mathsf{X},\mu_k^y)}\} < 1.
$$

The previous result can easily be extended to $K > 2$. Lastly, $L_r(\mathsf{X}^K, \boldsymbol{\mu}^y)$-geometric ergodicity $\forall r \in [1, \infty]$ follows from Lemma 3.2.1. □

We can use the previous result to prove the geometric ergodicity of the UGPT algorithm:

Theorem 4.4.1 (Ergodicity of UGPT): *Suppose* Assumption 4.4.1 *holds and denote by* $\boldsymbol{\mu}^y$ *the invariant measure of the UGPT Markov operator* $\mathbf{P}^{(\mathrm{UW})}$*. Then,* $\mathbf{P}^{(\mathrm{UW})}$ *has an* $L_2(\mathsf{X}^K, \boldsymbol{\mu}^y)$*-spectral gap. Moreover, the chain generated by* $\mathbf{P}^{(\mathrm{UW})}$ *is* $L_r(\mathsf{X}^K, \boldsymbol{\mu}^y)$*-geometrically ergodic for any* $r \in [1, \infty]$.

Proof. Recall that $\mathbf{P}^{(\mathrm{UW})} := \mathbf{Q}^{(\mathrm{UW})}\mathbf{P}\mathbf{Q}^{(\mathrm{UW})}$. From the definition of operator norm, we have that

$$
\begin{aligned}
&\left\|\mathbf{P}^{(\mathrm{UW})}\right\|_{L_2^0(\mathsf{X}^K,\boldsymbol{\mu}^y)\mapsto L_2^0(\mathsf{X}^K,\boldsymbol{\mu}^y)} \\
&\leq \left\|\mathbf{Q}^{(\mathrm{UW})}\right\|^2_{L_2^0(\mathsf{X}^K,\boldsymbol{\mu}^y)\mapsto L_2^0(\mathsf{X}^K,\boldsymbol{\mu}^y)} \|\mathbf{P}\|_{L_2^0(\mathsf{X}^K,\boldsymbol{\mu}^y)\mapsto L_2^0(\mathsf{X}^K,\boldsymbol{\mu}^y)} \\
&\leq \|\mathbf{P}\|_{L_2^0(\mathsf{X}^K,\boldsymbol{\mu}^y)\mapsto L_2^0(\mathsf{X}^K,\boldsymbol{\mu}^y)} < 1,
\end{aligned}
$$

where the previous line follows from Proposition 4.4.1 and the fact that $\mathbf{Q}^{(\mathrm{UW})}$ is a weak contraction in $L_2(\mathsf{X}^K, \boldsymbol{\mu}^y)$ (see, Equation (3.7)). Lastly, $L_r(\mathsf{X}^K, \boldsymbol{\mu}^y)$-geometric ergodicity $\forall r \in [1, \infty]$ follows from Lemma 3.2.1 and the fact that $\mathbf{P}^{(\mathrm{UW})}$ is $\boldsymbol{\mu}^y$-reversible by Proposition 4.3.3. □

We now turn to proving geometric ergodicity for the WGPT algorithm. We begin with an auxiliary result, lower-bounding the variance of a $\boldsymbol{\mu}_{\mathrm{W}}^{y}$-integrable functional $f \in L_2(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^{y})$.

Proposition 4.4.2: *Let $f \in L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^{y})$ be a $\boldsymbol{\mu}_{\mathrm{W}}^{y}$-integrable function such that $\|f\|_{L_2(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^{y})} = 1$, and denote by $\mathbb{V}_{\boldsymbol{\mu}_{\mathrm{W}}^{y}}[f]$, $\mathbb{V}_{\boldsymbol{\mu}_{\sigma}^{y}}[f]$ the variance of f with respect to $\boldsymbol{\mu}_{\mathrm{W}}^{y}$, $\boldsymbol{\mu}_{\sigma}^{y}$, respectively with $\sigma \in S_K$. In addition, suppose Assumption 4.4.1 holds. Then, it can be shown that*

$$0 < \frac{\Lambda_m}{2 - \Lambda_m} \leq \frac{1}{|S_K|} \sum_{\sigma \in S_K} \mathbb{V}_{\boldsymbol{\mu}_{\sigma}^{y}}[f] \leq \mathbb{V}_{\boldsymbol{\mu}_{\mathrm{W}}^{y}}[f] = 1,$$

with $\Lambda_m = \min_{\sigma,\rho \in S_K} \{\Lambda_{\sigma,\rho}\}$ and $\Lambda_{\sigma,\rho}$ as in Assumption 4.4.1-C3.

Proof. This proof is partially based on the proof of Theorem 1.2 in [106]. Let $\boldsymbol{u}, \boldsymbol{y} \in \mathsf{X}^K$ and define $\bar{f}_\sigma := \boldsymbol{\mu}_\sigma(f)$. The right-most inequality follows from the fact that

$$\begin{aligned}
1 = \mathbb{V}_{\boldsymbol{\mu}_{\mathrm{W}}^{y}}[f] &= \int_{\mathsf{X}^K} f(\boldsymbol{u})^2 \boldsymbol{\mu}_{\mathrm{W}}^{y}(\mathrm{d}\boldsymbol{u}) \\
&= \frac{1}{|S_K|} \sum_{\sigma \in S_K} \int_{\mathsf{X}^K} f^2(\boldsymbol{u}) \boldsymbol{\mu}_\sigma(\mathrm{d}\boldsymbol{u}) \\
&= \frac{1}{|S_K|} \sum_{\sigma \in S_K} \left(\mathbb{V}_{\boldsymbol{\mu}_\sigma}[f] + \bar{f}_\sigma^2 \right) \geq \frac{1}{|S_K|} \sum_{\sigma \in S_K} \mathbb{V}_{\boldsymbol{\mu}_\sigma}[f].
\end{aligned}$$

We follow a procedure similar to the proof of [106, Theorem 1.2] for the lower bound on the variance. We introduce an ordering on $S_K = \{\sigma_1, \sigma_2, \ldots, \sigma_{|S_K|}\}$, define the matrix $C \in \mathbb{R}^{|S_K| \times |S_K|}$ as the matrix with entries

$$C_{ij} = \int_{\mathsf{X}^K} \int_{\mathsf{X}^K} (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \boldsymbol{\mu}_{\sigma_i}(\mathrm{d}\boldsymbol{u}) \boldsymbol{\mu}_{\sigma_j}(\mathrm{d}\boldsymbol{y}),$$

where $C_{jj} = 2\mathbb{V}_{\boldsymbol{\mu}_{\sigma_j}}[f]$ and

$$\begin{aligned}
2 = 2\mathbb{V}_{\boldsymbol{\mu}_{\mathrm{W}}^{y}}[f] = \int_{\mathsf{X}^K} \int_{\mathsf{X}^K} (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 &\left(\frac{1}{|S_K|} \sum_{i=1}^{|S_K|} \boldsymbol{\mu}_{\sigma_i}(\mathrm{d}\boldsymbol{u}) \right) \\
&\times \left(\frac{1}{|S_K|} \sum_{j=1}^{|S_K|} \boldsymbol{\mu}_{\sigma_j}(\mathrm{d}\boldsymbol{y}) \right) \\
&= \sum_{i,j} \frac{1}{|S_K|^2} C_{ij}. \qquad (4.11)
\end{aligned}$$

We thus aim at finding an upper bound of Equation (4.11) in terms of $(|S_K|)^{-1} \sum_{\sigma \in S_K} \mathbb{V}_\sigma[f]$.

By assumption 4.4.1-C3, for any $\sigma_i, \sigma_j \in S_K$ the densities $\boldsymbol{\pi}_{\sigma_i}, \boldsymbol{\pi}_{\sigma_j}$ of $\boldsymbol{\mu}_{\sigma_i}, \boldsymbol{\mu}_{\sigma_j}$ (with respect to $\boldsymbol{\mu}_{\mathsf{pr}}$) have an overlap $\Lambda_{\sigma_i,\sigma_j} > 0$. For brevity, in the following we use the shorthand notation $\Lambda_{i,j}$ for $\Lambda_{\sigma_i,\sigma_j}$ Thus, we can find densities

$$\boldsymbol{\eta}_{ij} := \Lambda_{ij}^{-1} \min_{\boldsymbol{u} \in \mathsf{X}^K} \{\boldsymbol{\pi}_{\sigma_i}(\boldsymbol{u}), \boldsymbol{\pi}_{\sigma_j}(\boldsymbol{u})\}, \boldsymbol{\varphi}_i, \boldsymbol{\psi}_j$$

such that $\boldsymbol{\pi}_{\sigma_i} = \Lambda_{ij}\boldsymbol{\eta}_{ij} + (1 - \Lambda_{ij})\boldsymbol{\varphi}_i$, and $\boldsymbol{\pi}_{\sigma_j} = \Lambda_{ij}\boldsymbol{\eta}_{ij} + (1 - \Lambda_{ij})\boldsymbol{\psi}_j$. Thus, integrating over X^K, we get for the diagonal entries of the C matrix:

$$\begin{aligned}
C_{ii} &= 2\mathbb{V}_{\boldsymbol{\mu}_{\sigma_i}}[f] \\
&= \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \left(\Lambda_{ij}\boldsymbol{\eta}_{ij}(\boldsymbol{u}) + (1 - \Lambda_{ij})\boldsymbol{\varphi}_i(\boldsymbol{u})\right) \\
&\times \left(\Lambda_{ij}\boldsymbol{\eta}_{ij}(\boldsymbol{y}) + (1 - \Lambda_{ij})\boldsymbol{\varphi}_i(\boldsymbol{y})\right) \boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \\
&= \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \Lambda_{ij}^2 \boldsymbol{\eta}_{ij}(\boldsymbol{u})\boldsymbol{\eta}_{ij}(\boldsymbol{y})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \\
&+ \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \Lambda_{ij}(1 - \Lambda_{ij})\boldsymbol{\varphi}_i(\boldsymbol{y})\boldsymbol{\eta}_{ij}(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \\
&+ \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \Lambda_{ij}(1 - \Lambda_{ij})\boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\eta}_{ij}(\boldsymbol{y})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \\
&+ \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 (1 - \Lambda_{ij})^2 \boldsymbol{\varphi}_i(\boldsymbol{y})\boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \\
&= 2\Lambda_{ij}^2 \mathbb{V}_{\boldsymbol{\eta}_{ij}}[f] + 2(1 - \Lambda_{ij})^2 \mathbb{V}_{\boldsymbol{\varphi}_i}[f] + 2\Lambda_{ij}(1 - \Lambda_{ij}) \\
&\times \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \boldsymbol{\eta}_{ij}(\boldsymbol{u})\boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}). \qquad (4.12)
\end{aligned}$$

Notice that equation (4.12) implies that

$$\begin{aligned}
&\iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \boldsymbol{\eta}_{ij}(\boldsymbol{u})\boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \qquad (4.13) \\
&\leq \frac{\mathbb{V}_{\boldsymbol{\mu}_{\sigma_i}}[f] - \Lambda_{ij}^2 \mathbb{V}_{\boldsymbol{\eta}_{ij}}[f]}{\Lambda_{ij}(1 - \Lambda_{ij})}.
\end{aligned}$$

As for the non-diagonal entries of C, we have

$$\begin{aligned}
C_{ij} &= \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \left[\Lambda_{ij}\boldsymbol{\eta}_{ij}(\boldsymbol{u})\right. \\
&+ (1-\Lambda_{ij})\boldsymbol{\varphi}_i(\boldsymbol{u})\left]\right. (\Lambda_{ij}\boldsymbol{\eta}_{ij}(\boldsymbol{y}) \\
&+ (1-\Lambda_{ij})\boldsymbol{\psi}_j(\boldsymbol{y}))\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \\
&= 2\Lambda_{ij}^2 \mathbb{V}_{\boldsymbol{\eta}_{ij}}[f] \\
&+ (1-\Lambda_{ij})^2 \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\psi}_j(\boldsymbol{y})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \\
&+ \Lambda_{ij}(1-\Lambda_{ij}) \iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \\
&\times \left(\boldsymbol{\eta}_{ij}(\boldsymbol{u})\boldsymbol{\psi}_j(\boldsymbol{y}) + \boldsymbol{\eta}_{ij}(\boldsymbol{y})\boldsymbol{\varphi}_i(\boldsymbol{u})\right) \boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}).
\end{aligned} \tag{4.14}$$

We can bound the second term in the previous expression using Cauchy-Schwarz. Let $\boldsymbol{z} \in \mathsf{X}^K$. Then,

$$\begin{aligned}
&\iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\psi}_j(\boldsymbol{y})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \\
&= \iiint (f(\boldsymbol{u}) - f(\boldsymbol{z}) + f(\boldsymbol{z}) - f(\boldsymbol{y}))^2 \boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\psi}_j(\boldsymbol{y})\boldsymbol{\eta}_{ij}(\boldsymbol{z}) \\
&\times \boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{z}) \\
&\leq 2 \iiint \left((f(\boldsymbol{u}) - f(\boldsymbol{z}))^2 + (f(\boldsymbol{z}) - f(\boldsymbol{y}))^2\right) \boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\psi}_j(\boldsymbol{y})\boldsymbol{\eta}_{ij}(\boldsymbol{z}) \\
&\times \boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{z}) \\
&= 2 \iint (f(\boldsymbol{u}) - f(\boldsymbol{z}))^2 \boldsymbol{\varphi}_i(\boldsymbol{u})\boldsymbol{\eta}_{ij}(\boldsymbol{z})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{z}) \\
&+ 2 \iint (f(\boldsymbol{y}) - f(\boldsymbol{z}))^2 \boldsymbol{\psi}_j(\boldsymbol{y})\boldsymbol{\eta}_{ij}(\boldsymbol{z})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y})\boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{z}).
\end{aligned} \tag{4.15}$$

Thus, from equations (4.13), (4.14), and (4.15) we get

$$\begin{aligned}
C_{ij} &\leq 2\Lambda_{ij}^2 \mathbb{V}_{\boldsymbol{\eta}_{ij}}[f] + (2(1-\Lambda_{ij})^2 + \Lambda_{ij}(1-\Lambda_{ij})) \\
&\times \left(\iint (f(\boldsymbol{u}) - f(\boldsymbol{y}))^2 \left(\boldsymbol{\eta}_{ij}(\boldsymbol{u}) \boldsymbol{\psi}_j(\boldsymbol{y}) \right. \right. \\
&\left. \left. + \boldsymbol{\eta}_{ij}(\boldsymbol{y}) \psi_i(\boldsymbol{u}) \right) \boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{u}) \boldsymbol{\mu}_{\mathsf{pr}}(\mathrm{d}\boldsymbol{y}) \right) \\
&= 2\Lambda_{ij}^2 \mathbb{V}_{\boldsymbol{\eta}_{ij}}[f] + (2-\Lambda_{ij})(1-\Lambda_{ij}) \\
&\times \frac{\left(\mathbb{V}_{\boldsymbol{\mu}_{\sigma_i}}[f] - \Lambda_{ij}^2 \mathbb{V}_{\boldsymbol{\eta}_{ij}}[f] + \mathbb{V}_{\boldsymbol{\mu}_{\sigma_j}}[f] - \Lambda_{ij}^2 \mathbb{V}_{\boldsymbol{\eta}_{ij}}[f] \right)}{\Lambda_{ij}(1-\Lambda_{ij})} \\
&= \frac{2-\Lambda_{ij}}{\Lambda_{ij}} \left(V_{\boldsymbol{\mu}_{\sigma_i}}[f] + V_{\boldsymbol{\mu}_{\sigma_j}}[f] \right) - 4\Lambda_{ij}(1-\Lambda_{ij}) \mathbb{V}_{\boldsymbol{\eta}_{ij}}[f] \\
&\leq \frac{2-\Lambda_{ij}}{\Lambda_{ij}} \left(V_{\boldsymbol{\mu}_{\sigma_i}}[f] + V_{\boldsymbol{\mu}_{\sigma_j}}[f] \right),
\end{aligned} \tag{4.16}$$

since $\Lambda_{ij} \in (0,1)\ \forall i,j$. Finally, from equations (4.11) and (4.16) we get that

$$\begin{aligned}
1 = V_{\boldsymbol{\mu}_{\mathrm{W}}^y}[f] &= \frac{1}{2} \sum_{i,j} \frac{1}{|S_K|^2} C_{ij} \\
&\leq \frac{1}{2} \frac{1}{|S_K|^2} \sum_{i,j=1}^{|S_K|} \frac{2-\Lambda_{ij}}{\Lambda_{ij}} \left(V_{\boldsymbol{\mu}_{\sigma_j}}[f] + V_{\boldsymbol{\mu}_{\sigma_j}}[f] \right) \\
&\leq \frac{2-\Lambda_m}{\Lambda_m} \left(\frac{1}{|S_K|} \sum_{i=1}^{|S_K|} \mathbb{V}_{\boldsymbol{\mu}_{\sigma_i}}[f] \right),
\end{aligned}$$

with $\Lambda_m := \min_{i,j=1,2,\ldots,|S_K|} \{\Lambda_{ij}\} > 0$, and $\Lambda_{i,j}$ as in Assumption 4.4.1-C3. Notice that we have used (4.16) for the first inequality, including the case $i = j$, in the previous equation. This in turn yields the lower bound

$$0 < \frac{\Lambda_m}{2-\Lambda_m} \leq \left(\frac{1}{|S_K|} \sum_{i \in S_K} \mathbb{V}_{\boldsymbol{\mu}_i}[f] \right).$$

□

We are finally able to prove the ergodicity of the WGPT algorithm.

Theorem 4.4.2 (Ergodicity of WGPT): *Suppose Assumption 4.4.1 holds for some $r \in [1,\infty]$ and denote by $\boldsymbol{\mu}_{\mathrm{W}}^y$ the invariant measure of the WGPT Markov operator $\mathbf{P}^{(\mathrm{W})}$. Then, $\mathbf{P}^{(\mathrm{W})}$ has an $L_2(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^y)$-spectral gap. Moreover, the chain generated by $\mathbf{P}^{(\mathrm{W})}$ is $L_r(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^y)$ geometrically ergodic for any $r \in [1,\infty]$.*

Proof. Let $\mathcal{L} := \{f \in L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^{\boldsymbol{y}}) : \|f\|_{L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^{\boldsymbol{y}})} = 1\}$, and, for notational clarity, write

$$\|\mathbf{P}_\sigma\|_{L_2^0} := \|\mathbf{P}_\sigma\|_{L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_\sigma^{\boldsymbol{y}}) \mapsto L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_\sigma^{\boldsymbol{y}})}.$$

Then, from the definition of operator norm,

$$\begin{aligned}
&\left\|\mathbf{P}^{(\mathrm{w})}\right\|^2_{L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^{\boldsymbol{y}}) \mapsto L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^{\boldsymbol{y}})} \\
&= \sup_{f \in \mathcal{L}} \left\|\mathbf{P}^{(\mathrm{w})} f\right\|^2_{L_2(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^{\boldsymbol{y}})} \\
&= \sup_{f \in \mathcal{L}} \int_{\mathsf{X}^K} \left| \sum_{\sigma \in S_K} w_\sigma(\boldsymbol{u}) \int_{\mathsf{X}^K} f(\boldsymbol{y}) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right|^2 \boldsymbol{\mu}_{\mathrm{W}}^{\boldsymbol{y}}(\mathrm{d}\boldsymbol{u}) \\
&\leq \sup_{f \in \mathcal{L}} \int_{\mathsf{X}^K} \sum_{\sigma \in S_K} w_\sigma(\boldsymbol{u}) \left| \int_{\mathsf{X}^K} f(\boldsymbol{y}) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right|^2 \boldsymbol{\mu}_{\mathrm{W}}^{\boldsymbol{y}}(\mathrm{d}\boldsymbol{u}) \\
&= \sup_{f \in \mathcal{L}} \frac{1}{|S_K|} \sum_{\sigma \in S_K} \int_{\mathsf{X}^K} \left| \int_{\mathsf{X}^K} f(\boldsymbol{y}) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right|^2 \boldsymbol{\mu}_\sigma^{\boldsymbol{y}}(\mathrm{d}\boldsymbol{u}),
\end{aligned} \tag{4.17}$$

where the second to last line follows from the convexity of $(\cdot)^2$ and the last line follows from the definition of w_σ and $\boldsymbol{\mu}_{\mathrm{W}}^{\boldsymbol{y}}$. Now, let $\bar{f}_\sigma := \boldsymbol{\mu}_\sigma^{\boldsymbol{y}}(f)$. Notice that we have

$$\begin{aligned}
&\int_{\mathsf{X}^K} \left| \int_{\mathsf{X}^K} f(\boldsymbol{y}) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right|^2 \boldsymbol{\mu}_\sigma^{\boldsymbol{y}}(\mathrm{d}\boldsymbol{u}) \\
&= \int_{\mathsf{X}^K} \left| \int_{\mathsf{X}^K} (f(\boldsymbol{y}) - \bar{f}_\sigma + \bar{f}_\sigma) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right|^2 \boldsymbol{\mu}_\sigma^{\boldsymbol{y}}(\mathrm{d}\boldsymbol{u}) \\
&= \int_{\mathsf{X}^K} \left(\left| \int_{\mathsf{X}^K} (f(\boldsymbol{y}) - \bar{f}_\sigma) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right|^2 + \left| \int_{\mathsf{X}^K} \bar{f}_\sigma \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right|^2 \right. \\
&\quad \left. + 2\bar{f}_\sigma \int_{\mathsf{X}^K} (f(\boldsymbol{y}) - \bar{f}_\sigma) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right) \boldsymbol{\mu}_\sigma^{\boldsymbol{y}}(\mathrm{d}\boldsymbol{u}) \\
&= \underbrace{\int_{\mathsf{X}^K} \left(\int_{\mathsf{X}^K} (f(\boldsymbol{y}) - \bar{f}_\sigma) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right)^2 \boldsymbol{\mu}_\sigma^{\boldsymbol{y}}(\mathrm{d}\boldsymbol{u})}_{I} + (\bar{f}_\sigma)^2 \\
&\quad + 2\bar{f}_\sigma \underbrace{\int_{\mathsf{X}^K} \int_{\mathsf{X}^K} (f(\boldsymbol{y}) - \bar{f}_\sigma) \mathbf{p}_\sigma(\mathbf{u}, \mathrm{d}\mathbf{y}) \boldsymbol{\mu}_\sigma^{\mathbf{y}}(\mathrm{d}\mathbf{u})}_{= 0 \text{ by stationarity}}
\end{aligned} \tag{4.18}$$

Thus, multiplying and dividing I by

$$\left(\int_{\mathsf{X}^K} \left(f(\boldsymbol{u}) - \bar{f}_\sigma \right)^2 \boldsymbol{\mu}_\sigma^{\boldsymbol{y}}(\mathrm{d}\boldsymbol{u}) \right),$$

we obtain from the definition of $\|\mathbf{P}_\sigma\|^2_{L_2^0}$ that:

$$
\begin{aligned}
(4.18) &= \left(\frac{\int_{\mathsf{X}^K} \left(\int_{\mathsf{X}^K} (f(\boldsymbol{y}) - \bar{f}_\sigma) \mathbf{p}_\sigma(\boldsymbol{u}, \mathrm{d}\boldsymbol{y}) \right)^2 \boldsymbol{\mu}_\sigma^y(\mathrm{d}\boldsymbol{u})}{\int_{\mathsf{X}^K} \left(f(\boldsymbol{u}) - \bar{f}_\sigma \right)^2 \boldsymbol{\mu}_\sigma^y(\mathrm{d}\boldsymbol{u})} \right) \\
&\quad \times \left(\int_{\mathsf{X}^K} \left(f(\boldsymbol{u}) - \bar{f}_\sigma \right)^2 \boldsymbol{\mu}_\sigma^y(\mathrm{d}\boldsymbol{u}) \right) + (\bar{f}_\sigma)^2 \\
&\leq \|\mathbf{P}_\sigma\|^2_{L_2^0} \left(\int_{\mathsf{X}^K} \left(f(\boldsymbol{u}) - \bar{f}_\sigma \right)^2 \boldsymbol{\mu}_\sigma^y(\mathrm{d}\boldsymbol{u}) \right) + (\bar{f}_\sigma)^2 \\
&= \|\mathbf{P}_\sigma\|^2_{L_2^0} \left(\int_{\mathsf{X}^K} f(\boldsymbol{u})^2 \boldsymbol{\mu}_\sigma^y(\mathrm{d}\boldsymbol{u}) \right) \\
&\quad + \left(1 - \|\mathbf{P}_\sigma\|^2_{L_2^0} \right) (\bar{f}_\sigma)^2 \\
&= \left(\int_{\mathsf{X}^K} f(\boldsymbol{u})^2 \boldsymbol{\mu}_\sigma^y(\mathrm{d}\boldsymbol{u}) \right) - \underbrace{\left(1 - \|\mathbf{P}_\sigma\|^2_{L_2^0} \right)}_{:= \gamma, \text{ with } \gamma \in (0,1)} \\
&\quad \times \left(\int_{\mathsf{X}^K} \left(f(\boldsymbol{u}) - \bar{f}_\sigma \right)^2 \boldsymbol{\mu}_\sigma^y(\mathrm{d}\boldsymbol{u}) \right).
\end{aligned} \tag{4.19}
$$

Replacing Equation (4.19) into Equation (4.17), we get

$$
\begin{aligned}
&\left\| \mathbf{P}^{(\mathrm{W})} \right\|^2_{L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^y) \mapsto L_2^0(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^y)} \\
&\leq \sup_{f \in \mathcal{L}} \left(\int_{\mathsf{X}^K} f(\boldsymbol{u})^2 \boldsymbol{\mu}_{\mathrm{W}}^y(\mathrm{d}\boldsymbol{u}) \right) - \frac{\gamma}{|S_K|} \sum_{\sigma \in S_K} \mathbb{V}_{\boldsymbol{\mu}_\sigma^y}[f] \\
&\leq 1 - \gamma \left(\frac{\Lambda_m}{2 - \Lambda_m} \right) < 1 \quad \text{(by Proposition 4.4.2).}
\end{aligned}
$$

Thus, $\mathbf{P}^{(\mathrm{w})}$ has an $L_2(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^y)$ spectral gap. Once again, $L_r(\mathsf{X}^K, \boldsymbol{\mu}_{\mathrm{W}}^y)$-geometric ergodicity (with $r \in [1, \infty]$) follows from Lemma 3.2.1 and the fact that $\mathbf{P}^{(\mathrm{W})}$ is $\boldsymbol{\mu}_{\mathrm{W}}^y$-reversible by Proposition 4.3.4.

□

Discussion and comparison to similar theoretical result

Theorems 4.4.1 and 4.4.2 state the existence of an L_2-spectral gap, hence L_r-geometric ergodicity for both the UGPT and the WGPT algorithm. Their proof provides also a quantification of the L_2-spectral gap in terms of the L_2-spectral gap of each individual Markov operator P_k. Such a bound is, however, not satisfactory as it does not use any information on the temperature and it just states that the L_2-spectral gap of the UWPT and WGPT chain is not worse that the smallest L_2-spectral gap among the individual chains (without swapping). This result is not sharp, as it can

be evidenced in the numerical section, where a substantial improvement in convergence is achieved by our methods.
Convergence results for the *standard* parallel tempering algorithm have been obtained in the works [114] and [171]. In particular, the work [114] has proved geometric ergodicity for the pairwise parallel tempering algorithm using the standard drift condition construction of [113]. It is unclear from that work which convergence rate is obtained for the whole algorithm. In comparison, our results are given in terms of spectral gaps. On the other hand, the work [171] presents conditions for rapid mixing of a particular type of parallel tempering algorithm, where the transition kernel is to be understood as a convex combination of such kernels, as opposed to our case, where it is to be understood as a tensorization. Their obtained results provide, for their setting, a better convergence rate than the one we obtained for the UGPT. We believe that their result can be extended to the UGPT algorithm, and this will be the focus of future work. On the other hand, the use of the ideas in [171] for the WGPT algorithm seems more problematic.

4.5 Numerical experiments

We now present four academic examples to illustrate the efficiency of both GPT algorithms discussed herein and compare them to the more traditional random walk Metropolis and standard PT algorithms. Notice that we compare the different algorithms in their simplest version that uses random walk Metropolis as a base transition kernel. The only exception is in Section 4.5.7, which presents a high-dimensional BIP for which the preconditioned Crank-Nicolson [32] is used as the base method in all algorithms instead of RWM. More advanced samplers, such as Adaptive metropolis [63, 64], or other transition kernels, could be used as well to replace RWM or pCN. Experiments 4.5.3, 4.5.4 and 4.5.5 were run in a Dell (R) Precision (TM) T3620 workstation with Intel(R) Core(TM) i7-7700 CPU with 32 GB of RAM. Numerical simulations in Section 4.5.3 and 4.5.5 were run on a single thread, while the numerical simulations in Section 4.5.4 were run on an *embarrassingly parallel* fashion over 8 threads using the Message Passing Interface (MPI) and the Python package MPI4py [38]. Lastly, experiments 4.5.6 and 4.5.7 were run on the Fidis cluster of the EPFL. The scripts used to generate the results presented in this section were written in Python 3.6, and can be found in `DOI: 10.5281/zenodo.4736623`

4.5.1 Implementation remarks

In most Bayesian inverse problems, particularly those dealing with large-scale computational models, the computational cost is dominated by the evaluation of the forward operator, which can be, for example, the numerical approximation of a possibly non-linear partial differential equation. In the case where all possible permutations are considered (i.e., $S_K = \mathscr{S}_K$), there are $K!$ possible permutations of the states, the computation of the swapping ratio in the GPT algorithms can become prohibitively expensive if one is to evaluate $K!$ forward models, even for moderate values of K. This problem can be circumvented by storing the values $-\log(\pi^y(u_k^n))$, $k = 1, \dots, K$,

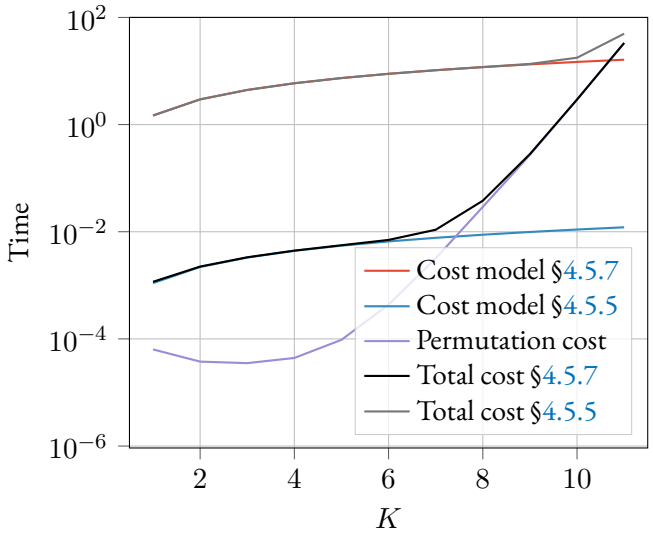

Figure 4.1: Cost per sample vs K for $S_K = \mathscr{S}_K$ for the forward model in Section 4.5.5 and the forward model in 4.5.7.

$n = 1, \dots N$, since the swapping ratio for GPT consists of permutations of these values, divided by the temperature parameters. Thus, "only" K forward model evaluations need to be computed at each step and the swapping ratio can be computed at negligible cost for moderate values of K. There is, however, a clear trade-off between the choice of K (which has a direct impact on the efficiency of the method), and the computational cost associated to (G)PT. Intuitively, a large K would provide a better mixing, however, it requires a larger number of forward model evaluations, which tends to be costly. We remark that such a trade-off between efficiency and number of function evaluations is also present in some advanced MCMC methods, such as Hamiltonian Monte Carlo, where one needs to choose a number of time steps for the time integration (see, e.g., [12]). Furthermore, there is an additional constraint when choosing $S_K = \mathscr{S}_K$, and it is the *permutation cost* associated to computing $r^{(\mathrm{UW})}(\boldsymbol{u}, \sigma)$ and $w_\sigma(\boldsymbol{u})$. In particular, the computation of either of those quantities has a complexity of $K!$ thus, this cost will eventually surpass the cost of evaluating the forward model K times. This is illustrated in Figure 4.1, where we plot the cost per sample of two different posteriors vs K. These posteriors are taken from the numerical examples in Sections 4.5.5 and 4.5.7. The posterior in Section 4.5.5 is rather inexpensive to evaluate, since one can compute the forward map $\mathcal{F}$ analytically (the difficulty associated to sampling from that posterior comes from its high multi-modality). On the contrary, evaluating the posterior in Section 4.5.7 requires numerically approximating the solution to a time-dependent, second-order PDE, and as such, evaluating such a posterior is costly. As we can see for $K \leq 6$, the computational cost in both cases is dominated by the forward model evaluation. Notice that for $K \leq 9$, the cost per sample from posterior (4.27) is dominated by the evaluation of the forward model.

Thus, for high values of K, it is advisable to only consider the union of properly chosen semi-groups A, B of $\mathscr{S}_K$, with $A \cap B \neq \emptyset$, such that A, B generates $\mathscr{S}_K$ (i.e., if the smallest semi-groups that contains A and B is $\mathscr{S}_K$ itself), and $|A \cup B| < |\mathscr{S}_K| = K!$, which is referred to as partial Infinite Swapping in the continuous case [49]. One particular way of choosing A and B is to consider, for example, A to be the set of permutations that only permute the indices associated with relatively low temperatures while leaving the other indices unchanged, and B as the set of permutations for the indices of relatively high temperatures, while leaving the other indices unchanged. Intuitively, swaps between temperatures that are, in a sense, "close" to each other tend to be chosen with a higher probability. We refer the reader to [49, Section 6.2] for a further discussion on this approach in the continuous-time setting. One additional idea would be to consider swapping schemes that, for example, only permute states between μ_i^y and $\mu_{i+1}^y, \mu_{i+2}^y, \ldots, \mu_{i+\ell}^y$ for some user-defined $\ell \geq 1$ and any given $i = 1, 2, \ldots, K-1$. The intuition behind this choice also being that swaps between posteriors that are at close temperatures are more likely to occur than swaps between posteriors with a high temperature difference. We intend to explore this further in depth in future work.

We reiterate that the total number of temperatures K depends heavily on the problem and the computational budget available [47, 163, 172] For the experiments considered in the work we chose $K = 4$ or $K = 5$, which provide an acceptable compromise between acceleration and cost.

Remark 4.5.1: *It was brought to our attention duing the private defense of this Thesis that in the case where $S_K = \bar{S}_K$ (i.e., when all $K!$ permutations of the set $\{1, 2, \ldots, K\}$ are considered), the (state-dependent) normalization term $\tilde{Z}(\boldsymbol{u}) := \sum_{\sigma \in S_K} \boldsymbol{\pi}^y(\boldsymbol{u}_\sigma)$, can be computed with a much lower complexity than $O(K!)$. This can in turn, drastically reduce the computational cost associated to GPT for the case where K is large and all possible permutaitons are considered. Indeed, given a matrix $A \in \mathbb{R}^{K \times K}$ with entries $A_{i,j}$, $i, j = 1, 2, \ldots, K$, we define its permanent $A \mapsto Perm(A)$ as*

$$Perm(A) := \sum_{\sigma \in S_K} \prod_{k=1}^{K} A_{k,\sigma(k)}.$$

Thus, it is easy to see that $\tilde{Z}(\boldsymbol{u})$ is the permanent of the matrix $A(\boldsymbol{u}) \in \mathbb{R}^{K \times K}$ with entries $A_{i,j} = \pi_i^y(u_j)$, $i, j = 1, \ldots, K$. It is shown in [7, 61, 146] that such an operation can be computed with complexity $O(2^{K-1}K)$. We intend to include such algorithms for the efficient cmputation of $\tilde{Z}(\boldsymbol{u})$ in future work.

4.5.2 Experimental setup

We now present an experimental setup common to all the numerical examples presented in the following subsections. In particular, all the experiments presented in this work utilize a *base* method given by either RWM (for experiments 4.5.3 through 4.5.6) or pCN (used in experiment 4.5.7) for the Markov transition kernels p. Furthermore, we take $S_K = \mathscr{S}_K$ for all experiments, where

$K = 5$ for experiment 4.5.5 and $K = 4$ for the other 4 experiments. In addition, we follow the *rule of thumb* of [52] for the choice of temperatures, setting, for each experiment, $T_k = a^{k-1}$, $k = 1, \dots, K$, for some positive constant $a > 1$. The particular choice of a is problem-dependent and it is generally chosen so that μ_K^y becomes sufficiently simple to explore. For each experiment we implement 5 MCMC algorithms to sample from a given posterior $\mu^y = \mu_1^y$, namely, the base (untempered) method (either RWM or pCN), and such a method combined with the standard PT algorithm (PT) with $N_s = 1$, the PSDPT algorithm of [90], and both versions of GPT. For our setting, the tempered algorithms have a cost (in terms of number of likelihood evaluations) that is K times larger than the base method. Thus, to obtain a fair comparison across all algorithms, we run the chain for the base method K times longer. Lastly, given some problem-dependent quantity of interest QoI, we assess the efficiency of our proposed algorithms to compute the posterior expectation of QoI by comparing the mean square error (experiments 4.5.3-4.5.5), for which the exact value of $\mathbb{E}_{\mu^y}[\mathsf{QoI}]$ is known, or the variance (experiments 4.5.6-4.5.7) of the ergodic estimator $\widehat{\mathsf{QoI}}$ obtained over N_{runs} independent runs of each algorithm.

4.5.3 Density concentrated over a quarter circle-shaped manifold

Let μ^y be a probability measure that has density π^y with respect to the uniform Lebesgue measure on the unit square $\boldsymbol{\mu}_{\mathsf{pr}} = \mathcal{U}([0,1]^2)$ given by

$$\pi^y(u) = \frac{1}{Z} \exp\left(-10000(u_1^2 + u_2^2 - 0.8^2)^2\right) \mathbf{1}_{[0,1]^2},$$

where $u = (u_1, u_2)$, Z is the normalization constant, and $\mathbf{1}_{[0,1]^2}$ is the indicator function over the unit square. We remark that this example is not of particular interest *per se*; however, it can be used to illustrate some of the advantages of the algorithms discussed herein. The difficulty of sampling from such a distribution comes from the fact that its density is concentrated over a quarter circle-shaped manifold, as can be seen on the left-most plot in Figure 4.2. This in turn will imply that a single level RWM chain would need to take very small steps in order to properly explore such density.

We aim at estimating $\widehat{\mathsf{QoI}}_i = \mathbb{E}_{\mu^y}[u_i] \approx \widehat{u}_i$, for $i = 1, 2$. For the tempered algorithms (PT, PSDPT, UGPT, and WGPT), we consider $K = 4$ temperatures and choose $T_4 = 5000$, so that the tempered density π_4^y becomes sufficiently simple to explore the target distribution. This gives $T_1 = 1, T_2 = 17.1, T_3 = 292.4, T_4 = 5000$. We compare the quality of our algorithms by examining the variance of the estimators $\widehat{u}_i$, $i = 1, 2$ computed over $N_{\text{runs}} = 100$ independent MCMC runs of each algorithm. For the tempered algorithms, each estimator is obtained by running the inversion experiment for $N = 25,000$ samples per run, discarding the first 20% of the samples (5000) as a burn-in. Accordingly, we run the single-chain random walk Metropolis algorithm for $N_{\text{RWM}} = KN = 100,000$ iterations, per run, and discard the first 20% of the samples obtained with the RWM algorithm (20,000) as a burn-in.

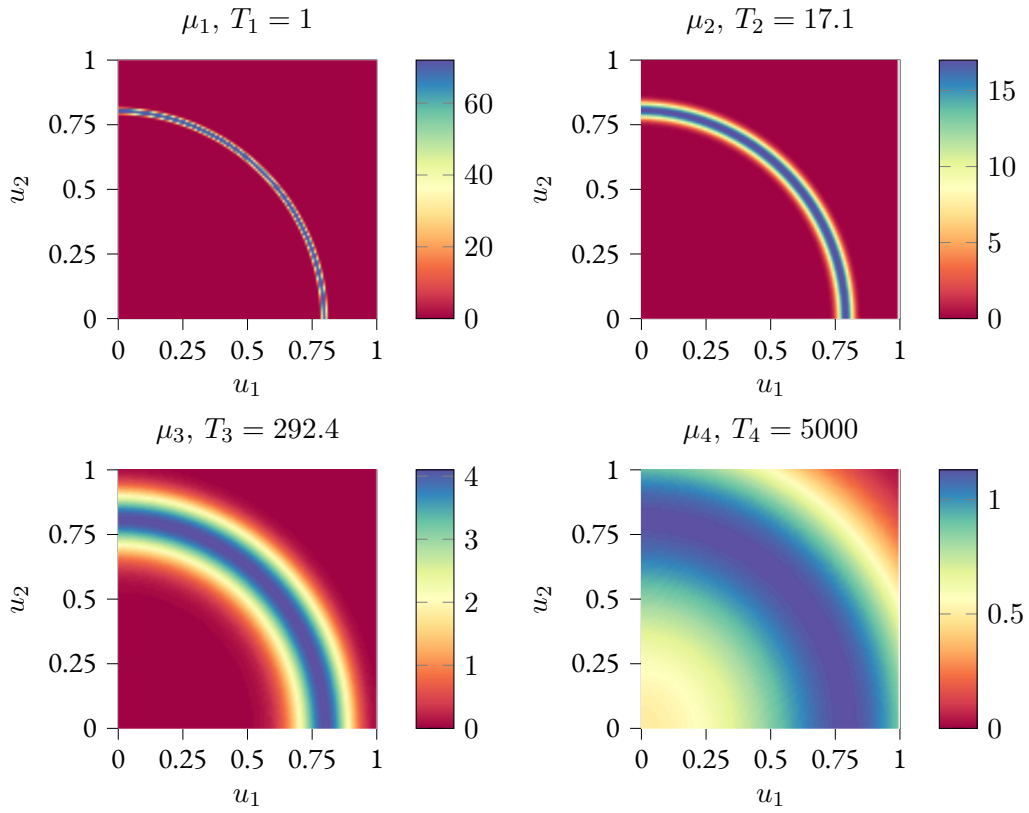

Figure 4.2: Tempered densities (with $T_1 = 1$, $T_2 = 17.1$, $T_3 = 292.4$, $T_4 = 5000$) for the density concentrated around a quarter circle-shaped manifold example. As we can see, the density becomes less concentrated as the temperature increases, which allows us to use RWM proposals with larger step sizes.

The untempered RWM algorithm uses Gaussian proposals with covariance matrix $\Sigma_{\text{RWM}} = \rho_1^2 I_{2\times 2}$, where $I_{2\times 2}$ is the identity matrix in $\mathbb{R}^{2\times 2}$, and $\rho_1^2 = 0.022$ is chosen in order to obtain an acceptance rate of around 0.23. For the tempered algorithms (i.e., PT, PSDPT, and both versions of GPT), we use $K = 4$ RWM kernels p_k, $k = 1, 2, 3, 4$, with proposal density $q_{\text{prop},k}(u_k^n, \cdot) = \mathcal{N}(u_k^n, \rho_k^2 I_{2\times 2})$, where ρ_k is shown in Table 4.1. This choice of ρ_k gives an acceptance rate for each chain of around 0.23 (determined empirically). Notice that ρ_1 corresponds to the "step-size" of the single-temperature RWM algorithm.

	$k=1$	$k=2$	$k=3$	$k=4$
ρ_k	0.022	0.090	0.310	0.650

Table 4.1: Step size of the RWM proposal distribution for the manifold experiment.

Experimental results for the ergodic run are shown in Table 4.2. We can see how both GPT algorithms provide a gain over RWM, PT and PSDPT algorithms, with the WGPT algorithm providing the largest gain. Scatter plots of the samples obtained with each method are presented in Figure 4.3. Here, the subplot titled "WGPT" (bottom row, middle) corresponds to weighted samples from $\boldsymbol{\mu}_{\text{W}}^y$, with weight $\widehat{w}$ as in (4.9), while the one titled "WGPT (inv)" (bottom row, right) corresponds to samples from $\boldsymbol{\mu}_{\text{W}}^y$ without any post-processing. Notice how the samples from the latter concentrates over a *thicker* manifold, which in turn makes the target density easier to explore when using state-dependent Markov transition kernels.

	Mean		MSE		$\text{MSE}_{\text{RWM}}/\text{MSE}$	
	$\widehat{u}_1$	$\widehat{u}_2$	$\widehat{u}_1$	$\widehat{u}_2$	$\widehat{u}_1$	$\widehat{u}_2$
RWM	0.50996	0.50657	0.00253	0.00261	1.00	1.00
PT	0.50978	0.51241	0.00024	0.00021	10.7	11.0
PSDPT	0.50900	0.50956	0.00027	0.00026	9.53	10.2
UGPT	0.50986	0.50987	0.00016	0.00016	16.1	16.4
WGPT	0.51062	0.50838	0.00015	0.00014	16.9	18.4

Table 4.2: Results for the density concentrated around a circle-shaped manifold experiment. As we can see, both GPT algorithms provide an improvement over PT, PSDPT and RWM. The computational cost is comparable across all algorithms.

4.5.4 Multiple source elliptic BIP

We now consider a slightly more challenging problem, for which we try to recover the probability distribution of the location of a source term in a Poisson equation (Eq. (4.20)), based on some noisy measured data. Let $(\mathsf{X}, \mathcal{B}(\mathsf{X}), \mu_{\mathsf{pr}})$ be the measure space, set $\mathsf{X} = \bar{D} := [0,1]^2$, with Lebesgue

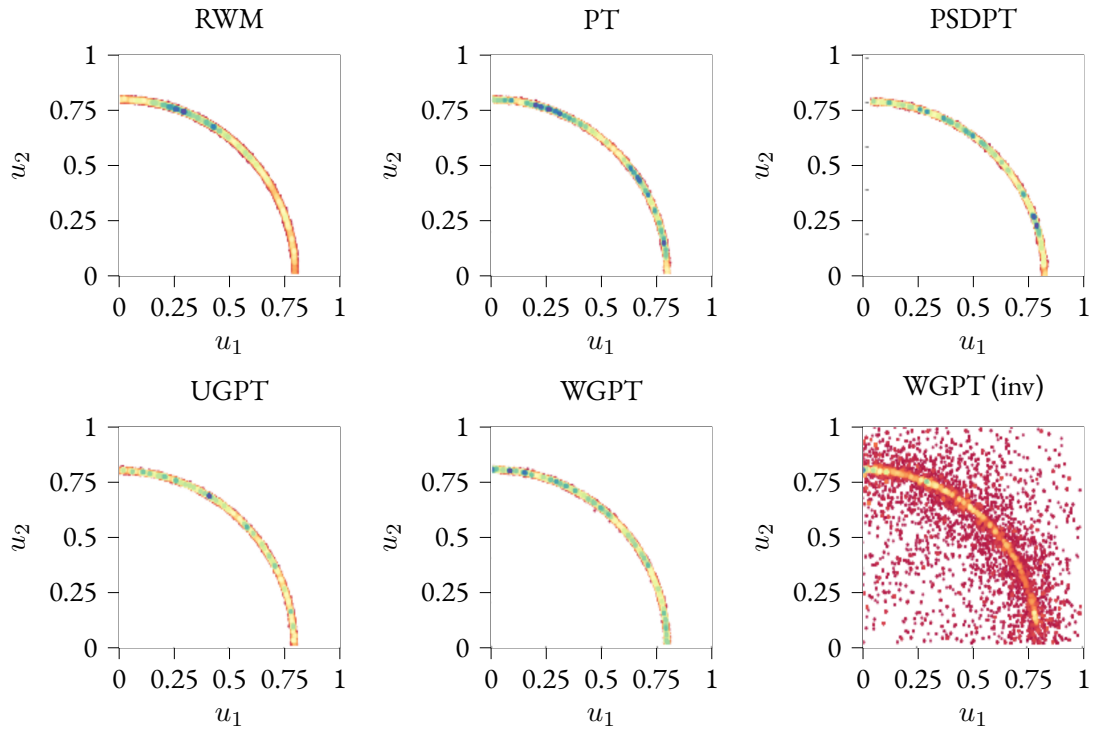

Figure 4.3: Scatter-plots of the samples from μ^y obtained with each algorithm on a single run. Top, from left to right: random walk Metropolis, PT and PSDPT. Bottom, from left to right: UGPT, WGPT (after re-weighting the samples), and WGPT, before re-weighting the samples.

(uniform) measure μ_{pr}, and consider the following Poisson's equation with homogeneous boundary conditions:

$$\begin{cases} -\Delta v(x,u) = f(x,u), & x \in D,\ u \in \mathsf{X}, \\ v(x,u) = 0, & x \in \partial D. \end{cases} \tag{4.20}$$

Such equation can model, for example, the electrostatic potential $v := v(x,u)$ generated by a charge density $f(x,u)$ depending on an *uncertain* location parameter $u \in \mathsf{X}$. Data y is recorded on an array of 64×64 equally-spaced points in D by solving (4.20) with a forcing term given by

$$f(x) = \sum_{i=1}^{4} e^{-1000[(x_1 - s_1^{(i)})^2 + (x_2 - s_2^{(i)})^2]}, \tag{4.21}$$

where the true source locations $s^{(i)}$, $i = 1,2,3,4$, are given by $s^{(1)} = (0.2, 0.2)$, $s^{(2)} = (0.2, 0.8)$, $s^{(3)} = (0.8, 0.2)$, and $s^{(4)} = (0.8, 0.8)$. Such data is assumed to be polluted by an additive Gaussian noise with distribution $\mathcal{N}(0, \eta^2 I_{64\times 64})$, with $\eta = 3.2 \times 10^{-6}$, (which

corresponds to a 1% noise) and where $I_{64\times 64}$ is the 64-dimensional identity matrix. Thus, we set $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}}) = (\mathbb{R}^{64\times 64}, \|\cdot\|)$, with $\|A\| = (64\eta)^{-2}\|A\|_F^2$, for some arbitrary matrix $A \in \mathbb{R}^{64\times 64}$, where $\|\cdot\|_F$ is the Frobenius norm. We assume a misspecified model where we only consider a single source in Eq. (4.21). That, is, we construct our forward operator $\mathcal{F} : \mathsf{X} \mapsto Y$ by solving (4.20) with a source term given by

$$f(x,u) = e^{-1000[(x_1-u_1)^2+(x_2-u_2)^2]}. \tag{4.22}$$

In this particular setting, this leads to a posterior distribution with four modes since the prior density is uniform in the domain and the likelihood has a local maximum whenever $(u_1, u_2) = (s_1^{(i)}, s_2^{(i)})$, $i = 1, 2, 3, 4$. The Bayesian inverse problem at hand can be understood as sampling from the posterior measure μ^y, which has a density with respect to the prior $\boldsymbol{\mu}_{\mathsf{pr}} = \mathcal{U}(\bar{D})$ given by

$$\pi^y(u) = \frac{1}{Z}\exp\left(-\frac{1}{2}\|y - \mathcal{F}(u)\|_\Sigma^2\right),$$

for some (intractable) normalization constant Z as in (2.7). We remark that the solution to (4.20) with a forcing term of the form of (4.22) is approximated using a second-order accurate finite difference approximation with grid-size $h = 1/64$ on each spatial component.

The difficulty in sampling from the current BIP arises from the fact that the resulting posterior μ^y is multi-modal and the number of modes is not known apriori (see Figure 4.4).

We follow a similar experimental setup to the previous example, and aim at estimating $\widehat{\mathsf{QoI}}_i = \mathbb{E}_{\mu^y}[u_i] \approx \widehat{u}_i$, for $i = 1, 2$. We use $K = 4$ temperatures and $N_{\text{runs}} = 100$. For the PT, PSDPT and GPT algorithms, four different temperatures are used, with $T_1 = 1$, $T_2 = 7.36$, $T_3 = 54.28$, and $T_4 = 400$. For each run, we obtain $N = 25,000$ samples with the PT, PSDPT, and both GPT algorithms, and $N = 100,000$ samples with RWM, discarding the first 20% of the samples in both cases (5000, 20000, respectively) as a burn-in. On each of the tempered chains, we use RWM proposals, with step-sizes shown in table 4.3. This choice of step size provides an acceptance rate of about 0.24 across all tempered chains and all tempered algorithms. For the single-temperature RWM run, we choose a larger step size ($\rho_{\text{RWM}} = 0.16$) so that the RWM algorithm is able to explore the whole distribution. Such a choice, however, provides a smaller acceptance rate of about 0.01 for the single-chain RWM.

Experimental results are shown in Table 4.4. Once again, we can see how both GPT algorithms provide a gain over RWM and both variations of the PT algorithm, with the WGPT algorithm providing a larger gain. Scatter-plots of the obtained samples are shown in Figure 4.4.

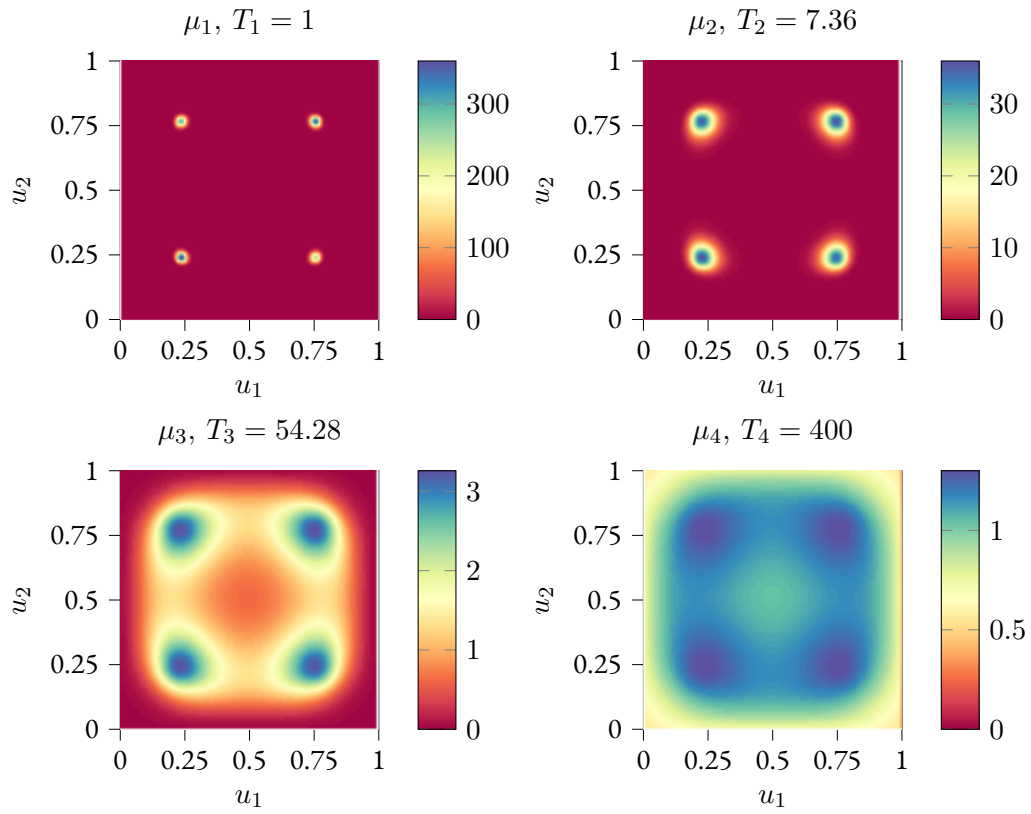

Figure 4.4: True tempered densities for the elliptic BIP example. Notice that the density is not symmetric, due to the additional random noise.

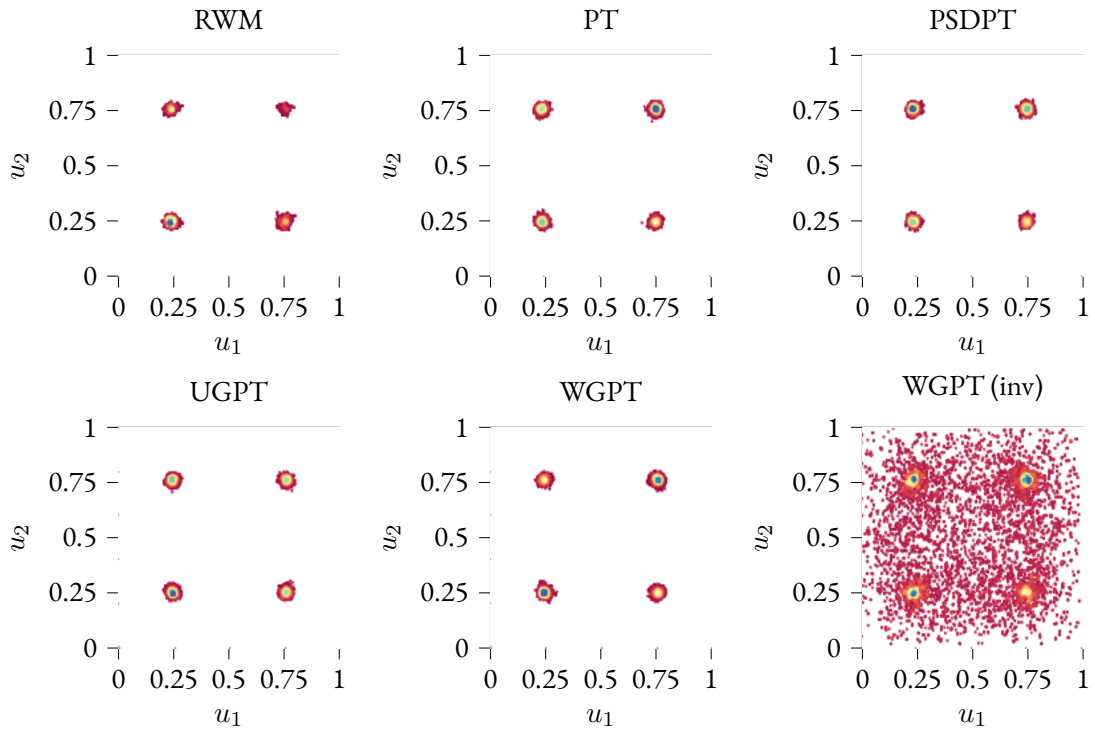

Figure 4.5: Scatterplots of the samples from μ^y obtained with different algorithms on a single run. Top, from left to right: random walk Metropolis, PT and PSDPT. Bottom, from left to right: UGPT, WGPT (after re-weighting the samples), and WGPT, before re-weighting the samples. As we can see, WGPT (before re-weighting) is able to "connect" the parameter space.

	$k=1$	$k=2$	$k=3$	$k=4$
$\rho_{k,\text{Tempered}}$	0.030	0.100	0.400	0.600
$\rho_{k,\text{RWM}}$	0.160	-	-	-

Table 4.3: Step size of the RWM proposal distribution for the elliptic BIP experiment.

	Mean		MSE		$\text{MSE}_{\text{RWM}}/\text{MSE}$	
	u_1	u_2	u_1	u_2	u_1	u_2
RWM	0.48509	0.51867	0.00986	0.01270	1.00	1.00
PT	0.48731	0.50758	0.00042	0.00036	23.0	29.2
PSDPT	0.48401	0.50542	0.00079	0.00099	12.4	10.7
UGPT	0.48624	0.50620	0.00038	0.00027	25.9	38.2
WGPT	0.48617	0.50554	0.00025	0.00023	38.6	44.9

Table 4.4: Results for the elliptic BIP problem. The computational cost is comparable across all algorithms, given that the cost of each iteration is dominated by the cost of solving the underlying PDE.

4.5.5 1D wave source inversion

We consider a small variation of example 5.1 in [115]. Let $(\mathsf{X}, \mathcal{B}(\mathsf{X}), \boldsymbol{\mu}_{\text{pr}})$ be a measure space, with $\mathsf{X} = [-5, 5]$ and uniform (Lebesgue) measure $\boldsymbol{\mu}_{\text{pr}}$, and let $I = (0, T]$ be a time interval. Consider the following Cauchy problem for the 1D wave equation:

$$\begin{cases} v_{tt}(x,t,u) - v_{xx}(x,t,u) = 0, & (x,t,u) \in \mathbb{R} \times I \times \mathsf{X}, \\ v(x,0,u) = h(x,u), & (x,t,u) \in \mathbb{R} \times \{0\} \times \mathsf{X}, \\ v_t(x,0,u) = 0, & (x,t,u) \in \mathbb{R} \times \{0\} \times \mathsf{X}. \end{cases} \tag{4.23}$$

Here, $h(x,u)$ acts as a source term generating a initial wave pulse. Notice that Equation (4.23) can be easily solved using d'Alembert's formula, namely

$$v(x,t,u) = \frac{1}{2}\left(h(x-t,u) + h(x+t,u)\right).$$

Synthetic data y is generated by solving Equation (4.23) with initial data

$$\begin{aligned} h(x,u_1,u_2) = \frac{1}{2}\Big(& e^{-100(x-u_1-0.5)^2} + e^{-100(x-u_1)^2} \\ & + e^{-100(x-u_1+0.5)^2} + e^{-100(x-u_2-0.5)^2} \\ & + e^{-100(x-u_2)^2} + e^{-100(x-u_2+0.5)^2} \Big), \end{aligned}$$

with $u_1 = -3, u_2 = 3$ and observed at $N_R = 11$ equally-spaced receiver locations between $R_1 = -5$ and $R_2 = 5$ on $N_T = 1000$ time instants between $t = 0$ and $T = 5$. The signal

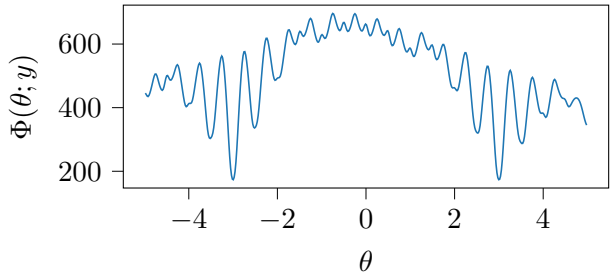

Figure 4.6: Multi-modal potential for the Cauchy problem. Notice the minima around $u = -3$ and $u = 3$.

recorded by each receiver is assumed to be polluted by additive Gaussian noise $\mathcal{N}(0, \eta^2 I_{1000\times 1000})$, with $\eta = 0.01$, which corresponds to roughly 1% noise. We set $(Y, \|\|_Y) = (\mathbb{R}^{11\times 1000}, \|\cdot\|_\Sigma)$, with

$$\|A\|_\Sigma^2 = (\sqrt{N_R}\eta)^{-2} \sum_{i=1}^{N_R} \sum_{j=1}^{N_T} A_{i,j}^2,$$

$A \in \mathbb{R}^{11\times 1000}$. Once again, we assume a misspecified model where we construct our forward operator $\mathcal{F} : \mathsf{X} \mapsto Y$ by solving (4.23) with a source term given by

$$h(x,u) = \left(e^{-100(x-u-0.5)^2} + e^{-100(x-u)^2} + e^{-100(x-u+0.5)^2}\right).$$

The Bayesian inverse problem at hand can be understood as sampling from the posterior measure μ^y, which has a density with respect to the prior $\boldsymbol{\mu}_{\mathsf{pr}} = \mathcal{U}([-5,5])$ given by

$$\begin{aligned} \pi^y(u) &= \frac{1}{Z} \exp\left(-\frac{1}{2}\|y - \mathcal{F}(u)\|_\Sigma^2\right) \\ &= \frac{1}{Z} \exp\left(-\Phi(u;y)\right), \end{aligned} \tag{4.24}$$

for some (intractable) normalization constant Z as in (2.7). The difficulty in solving this BIP comes from the high multi-modality of the potential $\Phi(u; y)$, as it can be seen in Figure 4.6. This, shape of $\Phi(u; y)$ makes the posterior difficult to explore using local proposals.
In this case, we consider $K = 5$, and set $T_1 = 1$, $T_2 = 5$, $T_3 = 25, T_4 = 125$ and $T_5 = 625$. Notice that from Figure 4.1, the computational cost per sample is dominated by the evaluation of (4.24) for values of $K \leq 6$. Once again, we obtain $N = 25,000$ samples with the PT, PSDPT, and both GPT algorithms, and $N = 125,000$ samples with RWM, discarding the first 20% of the samples in both cases (5000, 25000, respectively) as a burn-in. On each of the tempered chains, we use RWM proposals, with step-sizes shown in table 4.5. This choice of step size provides an acceptance rate of about 0.4 across all tempered chains and all tempered algorithms. The choice

of step-size for the RWM algorithm is done in such a way that it can "jump" modes, which are at distance of roughly 1/2.
We consider $\mathsf{QoI} = u$ as a quantity of interest. Experimental results are shown in Table 4.6. Once again, we can see how both GPT algorithms provide a gain over RWM and both variations of the PT algorithm, with the WGPT algorithm providing the largest gain. Notice that, given the high muti-modality of the posterior at hand, the simple RWM algorithm is not well-suited for this type of distribution, as it can be seen from its large variance; this suggests that the RWM usually gets "stuck" at one mode of the posterior. Notice that, intuitively, due to the symmetric nature of the potential, one would expect the true mean of u to be close to 0. This value was computed by means of numerical integration and is given by $\mathbb{E}^y_\mu[u] = 0.08211$.

	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$
$\rho_{k,\text{Tempered}}$	0.02	0.05	0.10	0.50	2.0
$\rho_{k,\text{RWM}}$	0.5	-	-	-	-

Table 4.5: Step size of the RWM proposal distribution for the Cauchy BIP experiment.

	Mean	MSE	$\text{MSE}_{\text{RWM}}/\text{MSE}$
RWM	-0.10120	9.36709	1.000
PT	0.05118	0.03681	254.5
PSDPT	0.15840	0.21701	43.20
UGPT	0.08976	0.03032	308.9
WGPT	0.06149	0.02518	372.0

Table 4.6: Results for the 1D Cauchy BIP problem. The computational cost is comparable across all algorithms.

4.5.6 Acoustic wave source inversion

We consider a more challenging problem, for which we try to recover the probability distribution of the spatial location of a (point-like) source term, together with the material properties of the medium, on an acoustic wave equation (see Eq. (4.25) below), based on some noisy measured data. We begin by describing the mathematical model of such wave phenomena. Let $(\mathsf{X}, \mathcal{B}(\mathsf{X}), \boldsymbol{\mu}_{\mathsf{pr}})$ be the measure space , with Lebesgue (uniform) measure $\boldsymbol{\mu}_{\mathsf{pr}}$, set $\bar{D} := [0,3] \times [0,2]$, $\partial D = \bar{\Gamma}_N \cup \bar{\Gamma}_{\text{Abs}}$, $\mathring{\Gamma}_{\text{N}} \cap \mathring{\Gamma}_{\text{Abs}} = 0$, $|\Gamma_{\text{N}}|, |\Gamma_{\text{Abs}}| > 0$, and define $\mathsf{X} = D \times \mathsf{X}_\alpha \times \mathsf{X}_\beta$, where $\mathsf{X}_\alpha = [6, 14]$, $\mathsf{X}_\beta = [4500, 5500]$. Here, we are considering a rectangular spatial domain D, with the top boundary denoted by Γ_{N} and the side and bottom boundaries denoted by Γ_{Abs}. Lastly, let

$u := (s_1, s_2, \alpha, \beta) \in \mathsf{X}$. Consider the following acoustic wave equation with absorbing boundary conditions:

$$\begin{cases} \alpha^2 v_{tt} - \nabla \cdot (\beta^2 \nabla u) = f, & \text{in } D \times (0,T) \times \mathsf{X}, \\ v = v_t = 0, & \text{in } D \times \{0\} \times \mathsf{X}, \\ \beta^2 \nabla v \cdot \widehat{n} = 0, & \text{on } \Gamma_{\text{N}} \times (0,T) \times \mathsf{X}, \\ \beta^2 \nabla v \cdot \widehat{n} = -\alpha\beta v_t, & \text{on } \Gamma_{\text{Abs}} \times (0,T) \times \mathsf{X}, \end{cases} \tag{4.25}$$

where $u = v(x,t,u)$, and $f = f(x,t,u)$. Here the boundary condition on the top boundary Γ_{N} corresponds to a Neumann boundary condition, while the boundary condition on Γ_{Abs} corresponds to the so-called absorbing boundary condition, a type of artificial boundary condition used to minimize reflection of wave hitting the boundary. Data $y \in Y$ is obtained by solving Equation (4.25) with a force term given by

$$\begin{aligned} f(x,t,u) = 10^{11} e^{-\frac{1}{2\cdot 0.1^2}\left[(x_1-s_1)^2+(x_2-s_2)^2\right]} \\ \times (1 - 2\cdot 1000\pi^2 t^2) e^{-2\cdot 1000^2 \pi^2 t^2}, \end{aligned} \tag{4.26}$$

with a true set of parameters $\mathsf{X} \ni u^* := (s_1, s_2, \alpha, \beta)$ given by $s_1 = 1.5, s_2 = 1.0, \alpha = 10, \beta = 5000$, and observed on $N_R = 3$ different receiver locations $R_1 = (1.0, 2.0), R_2 = (1.5, 2.0), R_3 = (2.0, 2.0)$ at $N_T = 117$ equally-spaced time instants between $t = 0$ and $t = 0.004$. In physical terms, the parameters s_1, s_2 represent the source location, while the parameters α, β are related to the material properties of the medium. Notice that, on a slight abuse of notation, we have used the symbol π to represent the number $3.14159\ldots$ in equation (4.26) and it should not be confused with the symbol for density. The data measured by each receiver is polluted by additive Gaussian noise $\mathcal{N}(0, \eta^2 I_{117\times 117})$, with $\eta = 0.013$, which corresponds to roughly a 2% noise. Thus, we have that $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}}) = (\mathbb{R}^{3\times 117}, \|\cdot\|_\Sigma)$, where $\|A\|_\Sigma^2 := (\sqrt{N_R}\eta)^{-2} \sum_{i=1}^{N_R} \sum_{j=0}^{N_T} A_{i,j}^2$. Thus, the forward mapping operator $\mathcal{F} : \mathsf{X} \mapsto \mathsf{Y}$ can be understood as the numerical solution of Equation (4.25) evaluated at 117 discrete time instants at each of the 3 receiver locations. Such a numerical approximation is obtained by the finite element method using piece-wise linear elements and the time stepping is done using a Forward Euler scheme with sufficiently small time-steps to respect the so-called Courant-Friedrichs-Lewy condition [134]. This numerical solution is implemented using the Python library FEniCS [101], using 40×40 triangular elements. The Bayesian inverse problem at hand can thus be understood as sampling from the posterior measure μ^y, which has a density with respect to the prior $\boldsymbol{\mu}_{\text{pr}} = \mathcal{U}(\mathsf{X})$ given by

$$\pi^y(u) = \frac{1}{Z} \exp\left(-\frac{1}{2}\|y - \mathcal{F}(u)\|_\Sigma^2\right).$$

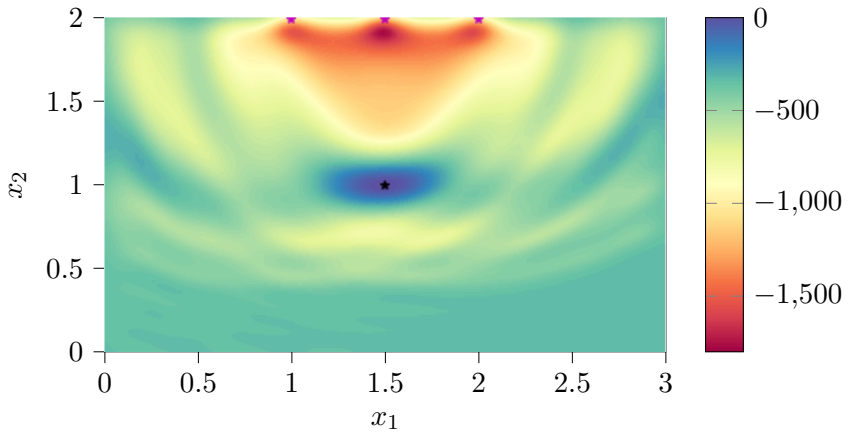

Figure 4.7: Plot of the log-likelihood for different values of s_1, s_2 and fixed values of $\alpha = 10$ and $\beta = 5000$. The magenta points represent the reciever locations R_1, R_2, R_3. The black point represents the true location of the source $(s_1, s_2) = (1.5, 1.0)$.

The previous BIP presents two difficulties; on the one hand, Equation (4.25) is, typically, expensive to solve, which in turn translates into expensive evaluations of the posterior density. On the other, the log-likelihood has an extremely complicated structure, which in turn makes its exploration difficult. This can be seen in Figure 4.7, where we plot of the log-likelihood for different source locations (s_1, s_2) and for fixed values of the material properties $\alpha = 10, \beta = 5000$. More precisely, we plot $\tilde{\Phi}((s_1, s_2); y) := -\frac{1}{2} \|y - \mathcal{F}(s_1, s_2, 10, 5000)\|_{\Sigma}^2$ on a grid of 100×100 equally spaced points (s_1, s_2) in D. It can be seen that, even though the log-likelihood has a clear peak around the true value of (s_1, s_2), there are also regions of relatively high concentration of log-probability, surrounded by regions with a significantly smaller log-probability, making it a suitable problem for our setting.

Following the same set-up of previous experiments, we aim at estimating $\widehat{\mathsf{QoI}}_i = \mathbb{E}_{\mu^y}[u_i] \approx \widehat{u}_i$, for $i = 1, 2$. Once again, we consider $K = 4$ temperatures for the tempered algorithms (PT, PSDPT, UGPT, and WGPT), and set temperatures to $T_1 = 1, T_2 = 7.36, T_3 = 54.28, T_4 = 400$. We compare the quality of our algorithms by examining the variance of the estimators $\widehat{u}_i$, $i = 1, 2$ computed over $N_{\text{runs}} = 50$ independent MCMC runs of each algorithm. For each run, we run the tempered algorithms obtaining $N = 7,000$ samples, discarding the first 20% of the samples (1400) as a burn-in. For the RWM algorithm, we run the inversion experiment for $N_{\text{RWM}} = KN = 28,000$ iterations, and discard the first 20% of the samples obtained (5600) as a burn-in.

Each individual chain is constructed using Gaussian RWM proposals $q_{\text{prop},k}(u_k^n, \cdot) = \mathcal{N}(u_k^n, \mathcal{C}_k)$, $k = 1, 2, 3, 4$, with covariance $\mathcal{C}_k$ described in Table 4.7. The covariance is tuned in such a way that the acceptance rate of each chain is around 0.2. The variance of the estimators obtained with each method is presented in Table 4.8. Once again, our GPT algorithms outperform all other tested methods for this particular setting. In particular, our methods provide huge computational

gains when compared to RWM and the PSDPT algorithm of [90], as well as some moderate computational gains when compared to the standard PT.

	$\mathcal{C}^{1/2}_{k,\text{Tempered}}$	$\mathcal{C}^{1/2}_{k,\text{RWM}}$
$k=1$	$\text{Diag}(0.01, 0.01, 0.2, 5)$	$\text{Diag}(0.02, 0.02, 0.2, 5)$
$k=2$	$\text{Diag}(0.06, 0.06, 0.4, 14)$	-
$k=3$	$\text{Diag}(0.3, 0.3, 0.6, 20)$	-
$k=4$	$\text{Diag}(1, 1, 1, 50)$	-

Table 4.7: Step size of the RWM proposal distribution for the acoustic BIP experiment. Here $\text{Diag}(d_1, d_2, \dots, d_N)$ is to be understood as the $N \times N$ diagonal matrix with entries $d_1, d_2, \dots, d_N$.

	Mean		Var		$\text{Var}_{\text{RWM}}/\text{Var}$	
	s_1	s_2	s_1	s_2	s_1	s_2
RWM	1.33801	1.54293	9.86×10^{-1}	8.21×10^{-2}	1.000000	1.000
PT	1.50121	1.00829	6.61×10^{-6}	2.77×10^{-4}	149136.1	296.2
PSDPT	1.39775	1.23119	2.48×10^{-1}	6.54×10^{-2}	3.900000	1.200
UGPT	1.50177	1.00711	2.72×10^{-6}	2.38×10^{-4}	361744.5	345.0
WGPT	1.50174	1.00601	2.08×10^{-6}	1.46×10^{-4}	472133.2	558.6

Table 4.8: Results for the acoustic BIP problem. Once again, we can see that both GPT algorithm provide an improvement over RWM, PT and PSDPT. The computational cost is comparable across all algorithms, given that the cost of each iteration is dominated by the cost of solving the underlying PDE.

4.5.7 High-dimensional acoustic wave inversion

Lastly, we present a high-dimensional example for which we try to invert for the material properties β^2 in (4.25). For simplicity, we will consider fixed values of $\alpha = 1$, $s_1 = 1.5$, and $s_2 = 1$. In this case, we set $\beta^2 = 10 + \widehat{\beta}^2(x)$, where $\widehat{\beta}(x)$ is taken to be a realization of a random field discretized on a mesh of $N_x \times N_y$ triangular elements. This modeling choice ensures that β^2 is lower bounded. In this case, we will invert for the nodal values of (the finite element discretization of) $\widehat{\beta}$, which will naturally result in a high-dimensional problem. We remark that one is usually interested in including the randomness in β^2, instead of $\widehat{\beta}$; however, we purposely choose to do so to induce an explicitly multi-modal posterior, and as such, to better illustrate the advantages of our proposed methods when sampling from these types of distributions.
We begin by formalizing the finite-element discretization of the parameter space (see e.g., [24] for a more detailed discussion).

Let $\bar{D} = [0,3] \times [0,2]$, denote the physical space of the problem and let V_h be a finite-dimensional subspace of $L_2(D)$ arising from a given finite element discretization. We write the finite element approximation $\widehat{\beta}_h \in V_h$ of $\widehat{\beta}$ as

$$\widehat{\beta}(x) \approx \widehat{\beta}_h(x) = \sum_{n=1}^{N_\text{v}} b_n \phi_n(x),$$

where $\{\phi\}_{n=1}^{N_\text{v}}$ are the Lagrange basis functions corresponding to the nodal points $\{x_n\}_{n=1}^{N_\text{v}}$, $(b_1, \ldots, b_{N_\text{v}})^T =: u \in \mathbb{R}^{N_\text{v}}$ is the set of nodal parameters and N_v corresponds to the number of vertices used in the FE discretization. Thus, the problem of inferring the distribution of β given some data y, can be understood as inferring the distribution of u given y. For our particular case, we will discretize D using 28×28 (non-overlapping) piece-wise linear finite elements, which results in $N_\text{v} = 841$ and as such $\mathsf{X} = \mathbb{R}^{841}$. We consider a Gaussian prior $\mu^y_{\text{pr},\infty} = \mathcal{N}(0, \mathcal{A}^{-2})$ (c.f Section 2.2.1), where $\mathcal{A}$ is a differential operator acting on $L_2(D)$ of the form

$$\mathcal{A} := -a\nabla \cdot (H\nabla) + dI, \quad a, d > 0,$$

together with Robin boundary conditions $\nabla(\cdot) \cdot \widehat{n} + \sqrt{ad}(\cdot) = 0$, where, following [164], H is taken of the form

$$H := \begin{pmatrix} e_1 \sin^2(\ell) + e_2 \cos^2(\ell) & (e_1 - e_2)\sin(\ell)\cos(\ell) \\ (e_1 - e_2)\sin(\ell)\cos(\ell) & e_1 \cos^2(\ell) + e_2 \sin^2(\ell) \end{pmatrix}.$$

Here H models the spatial anisotropy of a Gaussian Random field sampled from $\mu_{\text{pr},\infty}$. It is known that for a two-dimensional (spatial) space, the covariance operator $\mathcal{A}^{-2}$ is symmetric and trace-class [24], and as such, the (infinite-dimensional) prior measure is well-defined. Thus, we set

$$\beta(x) \sim \mu_{\text{pr},\infty},$$

which in turn induces the discretized prior:

$$\widehat{\beta}_h(x) \sim \boldsymbol{\mu}_\text{pr} := \mathcal{N}(0, \mathcal{A}_h^{-2}),$$

where $\mathcal{A}_h^{-2}$ is a finite-element approximation of $\mathcal{A}$ using 28×28 (non-overlapping) piece-wise linear finite elements. Samples from $\boldsymbol{\mu}_\text{pr}$ are obtained using the `FEniCS` package [101] and the `hIPPYlib` library [164].

We follow an approach similar to our previous example. We collect data $y \in Y$ by solving Equation (4.25) with a force term given by (4.26) and a true field $\widehat{\beta}^*_h \sim \boldsymbol{\mu}_\text{pr}$ with $a = 0.1$, $d = 0.5$, $\ell = \pi^y/4$, $e_1 = 2$ and $e_2 = 0.5$. Such a realization of β is shown in Figure 4.8.

Furthermore, data is observed at $N_R = 5$ different receiver locations $R_1 = (1.0, 2.0)$, $R_2 = (1.25, 2.0)$, $R_3 = (1.5, 2.0)$, $R_4 = (1.75, 2.0)$, and $R_5 = (2.0, 2.0)$ at $N_T = 600$ equally-

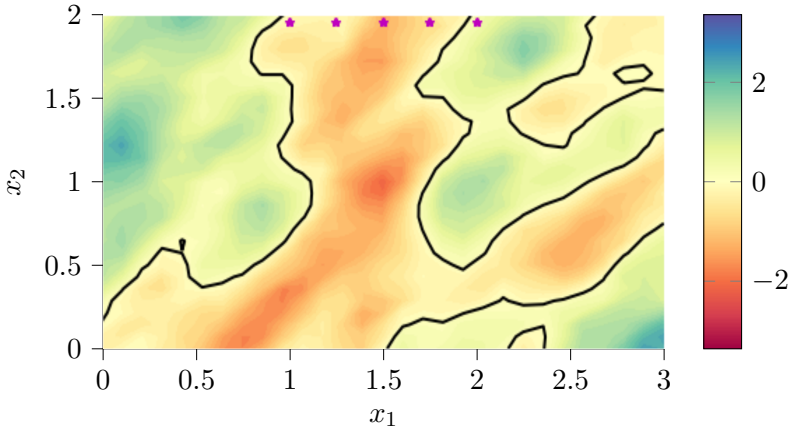

Figure 4.8: True field $\beta(x)$. Notice the anisotropy on the field. The magenta points represent the receiver locations. The black line represents the zero-level set of the field.

spaced time instants between $t = 0$ and $t = 0.6$. The data measured by each receiver is polluted by an (independent) additive Gaussian noise $\eta \sim \mathcal{N}(0, \sigma^2_{\text{noise}} I_{600\times 600})$, with $\sigma = 0.021$, which corresponds to roughly a 0.5% noise. Thus, we have that $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}}) = (\mathbb{R}^{5\times 600}, \|\cdot\|_{\Sigma})$. Similarly as in Section 4.5.6, the forward mapping operator $\mathcal{F} : \mathsf{X} \mapsto \mathsf{Y}$ can be understood as the numerical solution of Equation (4.25) evaluated at 600 discrete time instants at each of the 5 receiver locations. Numerical implementation follows a similar set-up as in Section 4.5.6, however, for simplicity, we use 28×28 triangular elements to approximate the forward operator $\mathcal{F}$. The Bayesian inverse problem at hand can thus be understood as sampling from the posterior measure μ^y, which has a Radon-Nikodym derivative with respect to the prior μ_{pr} given by

$$\pi^y(u) = \frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u) = \frac{1}{Z} \exp\left(-\frac{1}{2}\|y - \mathcal{F}(u)\|^2_{\Sigma}\right). \tag{4.27}$$

The previous BIP has several difficulties; clearly, it is a high-dimensional posterior. Furthermore, just as in the previous example, the underlying mathematical model for the forward operator is a costly time-dependent PDE. Lastly, by choosing to invert for $\widehat{\beta} \sim \mu_{\mathsf{pr}}$ (instead of β^2), and since μ_{pr} is centered at zero, we induce a multi-modal posterior, indeed, if the posterior concentrates around $\widehat{\beta}^*_h$ it will also have peaks at any other $\widehat{\beta}^j$ obtained by flipping the sign of $\widehat{\beta}^*_h$ in a concentrated region separated by the zero level set of $\widehat{\beta}^*_h$ (we identify 7 regions in Figure 4.8). This can be seen in Figure 4.9, where we plot 4 samples from μ. Notice the change in sign between some regions. Lastly, as a quantities of interest, we will consider $\mathsf{QoI}_1 = \int_D \exp(\widehat{\beta}(x))\mathrm{d}x$ and $\mathsf{QoI}_2 = \exp(\widehat{\beta}(1.5, 1))$. We remark that, although these quantities of interest do not have any meaningful physical interpretation, they are, however, affected by the multi-modality of the posterior, and as such, well suited to exemplify the capabilities of our method.

Given the high-dimensionality of the posterior, we present a slightly different experimental setup in order to estimate $\mathbb{E}_{\mu}[\mathsf{QoI}_i] \approx \widehat{\mathsf{QoI}}_i$, $i = 1, 2$. In particular, we will use the preconditioned

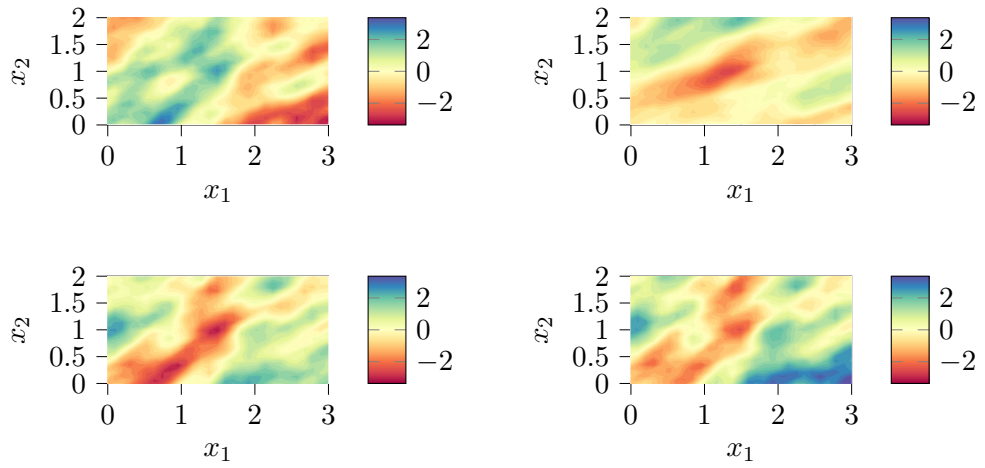

Figure 4.9: Posterior samples obtained with the UW GPT algorithm. Notice the resemblance to Figure 4.8.

Crank-Nicolson (pCN) as a base method, instead of RWM, for the transition kernel p. We compare the quality of our algorithms by examining the variance of the estimators $\widehat{\mathsf{QoI}}_i$ computed over $N_{\text{runs}} = 50$ independent MCMC runs of each algorithm, with $K = 4$ temperatures for the tempered algorithms given by $T_1 = 1, T_2 = 4.57, T_3 = 20.89, T_4 = 100$. For the tempered algorithms, each estimator is obtained by running the inversion experiment for $N = 4,800$ samples, discarding the first 20% of the samples (800) as a burn-in. For the untempered pCN algorithm, we run the inversion experiment for $N_{\text{pCN}} = KN = 19,200$ iterations, and discard the first 20% of the samples obtained (3840) as a burn-in.

Each individual chain is constructed using pCN proposals $q_{\text{prop},k}(u_k^n, \cdot) = \mathcal{N}(\sqrt{1-\rho_k^2}u_k^n, \rho_k^2 \mathcal{A}_h^{-2})$, $k = 1, 2, 3, 4$, with ρ_k described in Table 4.9. The simple, un-tempered pCN algorithm is run with a step size given by $\rho = \rho_1$. The values of ρ_k are tuned in such a way that the acceptance rate of each chain is around 0.3 and are reported in Table 4.9. The variance of the estimators obtained with each method is presented in Table 4.10. Once again, even for this high-dimensional, highly multi-modal case, our proposed methods perform considerably better than the other algorithms.

	$k=1$	$k=2$	$k=3$	$k=4$
ρ_k	0.1	0.2	0.4	0.8

Table 4.9: Values of ρ_k for the pCN kernel for the high-dimensional wave inversion problem.

4.5.8 Application to a (semi-)realistic seismic source inversion problem: Tanzania case study

Lastly, we conclude this chapter by applying our WGPT algorithm to the solution of a BIP arising in seismic source inversion. Given some noise-polluted data recorded at three different locations, we are interested in obtaining the probability distribution for the source location of an earthquake

	Mean		Var		$\text{Var}_{\text{pCN}}/\text{Var}$	
	$\widehat{\text{QoI}}_1$	$\widehat{\text{QoI}}_2$	$\widehat{\text{QoI}}_1$	$\widehat{\text{QoI}}_2$	$\widehat{\text{QoI}}_1$	$\widehat{\text{QoI}}_2$
pCN	8.8665	1.5255	5.7362	0.6029	1.00	1.00
PT +pCN	8.7710	1.5311	1.3308	0.1380	4.31	4.36
PSDPT +pCN	8.5546	1.4453	2.1289	0.2666	2.69	2.26
UGPT +pCN	8.7983	1.4614	1.0543	0.1051	5.49	5.73
WGPT +pCN	8.6464	1.4643	1.0126	0.1016	5.74	5.93

Table 4.10: Results for the high-dimensional acoustic BIP problem. As for the previous examples, The computational cost is comparable across all algorithms.

given that the material properties of the medium are also unknown. Such an experiment is a computational model of an earthquake that took place on the Tanzania basin on the 12th of October 2016 at 1:31:53. Given the source-receiver configuration, this seismic source inversion problem can be well-approximated by a two-dimensional model (c.f. Figure 4.10). We consider a rectangular domain $\bar{D} = [0, 145000] \times [0, 87000]$ m^2, together with a time interval $I = [0, T]$ $T = 17s$. We will model the seismic event as an elastic wave equation (c.f. Equation (1.4)), that we restate here for convenience. Given some Banach space X (that we will define shortly), the forward model of the wave phenomena reads as *find a displacement field* $w : I \times D \times \mathsf{X} \to \mathbb{R}^2$ *such that:*

$$\begin{cases} \rho(x,u)w_{tt}(t,x,u) - \nabla \cdot \sigma(x,u,w) = -M \cdot \nabla\delta(x-u_s)S(t), & \text{for } (t,x,u) \in I \times D \times \mathsf{X} \\ w(0,x,u) = 0,\ w_t(0,x,u) = 0, & \text{for } \{t=0\},\ (x,u) \in \times D \times \mathsf{X}, \end{cases}$$

where

$$\begin{aligned} \sigma(x,u,w) &= \lambda(x,u)\nabla \cdot wI + m(x,u)(\nabla w + (\nabla w)^T), \\ S(t) &= \frac{3f_0}{\sqrt{2\pi}} \exp\left(-\frac{2f_0^2(t+t_0)^2}{2}\right), \quad t_0 = -0.6s,\ f_0 = 2\text{Hz}, \\ M &= \begin{pmatrix} 5.5895 \times 10^{13} & 7.9762 \times 10^{13} \\ 7.9762 \times 10^{13} & -2.5698 \times 10^{14} \end{pmatrix}, \end{aligned}$$

together with Neumann Boundary conditions at the surface of the domain, and with perfectly matched layers (a kind of absorbing boundary conditions) on the side and lower boundary, to simulate the propagation of the wave through a "infinite", layered medium on those directions. Here, $\rho(x,u)$, represents the density of the material, $\lambda(x,u), m(x,u)$ represent the Lamé parameters, and the forcing term models an explosion centered at $u_s \in \mathsf{X}$. We write the material properties

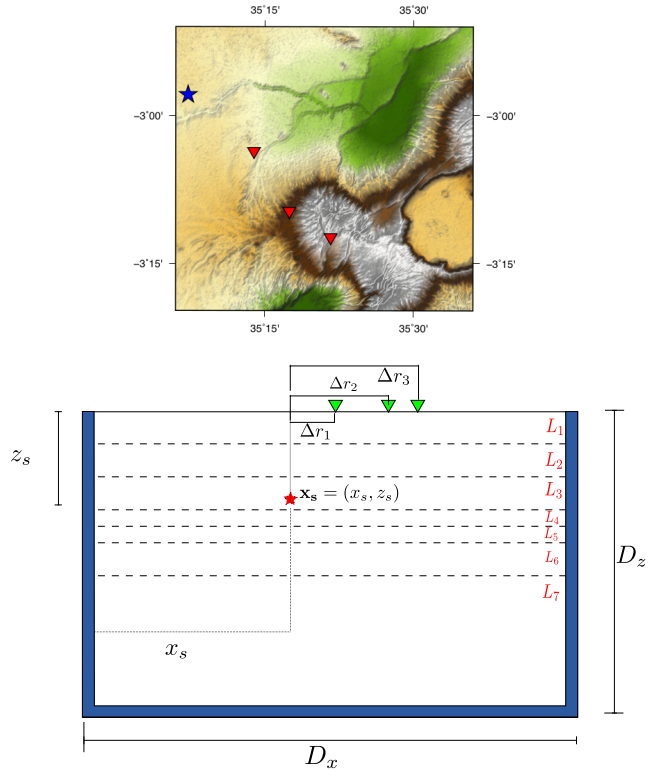

Figure 4.10: (Top). Aerial view of the source-receiver geometry. Receivers are denoted in red and source location is in blue. Figure reproduced from [4], with permission from the publisher (Springer Nature). (Bottom). Depiction of the computational domain of the Tanzania test-case. Blue represents the PML.

(ρ, λ, μ) of the earth in terms of its density, compressional V_p and shear wave V_s velocities, given by

$$V_p(x,u) = \sqrt{\frac{\lambda(x,u) + 2m(x,u)}{\rho(x,u)}}, \quad V_s(x,u) = \sqrt{\frac{m(x,u)}{\rho(x,u)}}.$$

We make the following simplifying assumptions:

1. The number of layers (7) and their depth are known beforehand.
2. The material properties (ρ, V_p, V_s) are constant on each layer.
3. The Moment tensor M is known.

These assumptions can be justified by known models for the structure of the Earth (see, e.g., [51]), and were discussed in collaboration with the Computational Earthquake Seismology group from the King Abdullah University of Science and Technology (KAUST), lead by Prof. Martin Mai.

Such assumptions drastically reduce the number of unknown parameters in the inversion; indeed we would have two parameters for the source location + 3×7 parameters for the material properties. Synthetic data is generated using the values shown in Table 4.11 and recorded by three receivers located at the top boundary of the domain. 1700 data-points per seismograph are obtained. Synthetic data is polluted by Gaussian additive noise representing 1% of the maximum amplitude of the recorded signal. We assume there is no correlation on the noise between time instances or receivers.

Source Location			
$x_s = 54000$		$z_s = 59500$	
Layer \ Property	ϱ	V_p	V_s
Layer 1	2571	6128	3459
Layer 2	2426	6355	3799
Layer 3	2520	6799	3823
Layer 4	2599	6854	3985
Layer 5	2972	7906	4673
Layer 6	3076	8424	4928
Layer 7	3060	8434	4999

Table 4.11: Set of true parameters, by which the data are synthetically generated, approximated to the closest unit.

Computationally, The domain is discretized using the spectral element method, using 116×68 elements, with 5 Gauss-Legendre-Lobatto (GLL) nodes per element. We use a leap-frog scheme for the evolution of the forward model, up to the final time of $T = 17s$, and with a time discretization of $\Delta t = 5 \times 10^{-3}$ s. This is implemented using the software SPECFEM2D [88].
We record (noise-polluted) data and aim to recover the probability distribution of the source location, as well as the uncertain material properties. Thus, we have $M = 23$ total unknown parameters (2 spatial components + 3×7 material properties), namely

$$u = (\underbrace{x_0, z_0}_{:=u_s}, \varrho_1, V_{p_1}, V_{s_1}, \dots, \varrho_7, V_{p_7}, V_{s_7}),$$

where ϱ_i (resp. V_{p_i}, V_{s_i}) represents the density (resp. bulk modulus and compressional velocity) at the i^{th} layer. For the prior distribution μ_{pr}, we set

$$\mu_{\text{pr}} = \bigotimes_{i=1}^{M} \mu_{\text{pr}_i},$$

where μ_{pr_i} is the prior of the i^{th} parameter. In particular, we consider uniform priors for all components of u, thus making $\pi_i^0 = \mathcal{U}(a, b)$, where a and b are the minimum and maximum admissible values for u_i. For the source location, we set $(x_0, z_0) = u_s \sim \mathcal{U}(D)$ where x_0, z_0 represent the horizontal and vertical component of the source location, respectively. As for the material properties, following [4], we use the following priors.

$$\begin{aligned} \varrho_i &\sim \mathcal{U}(0.9\varrho_i^{\text{true}}, 1.1\varrho^{\text{true}}), \\ V_{s,i} &\sim \mathcal{U}(0.95V_{s,i}^{\text{true}}, 1.05V_{s,i}^{\text{true}}), \\ V_{p,i} &\sim \mathcal{U}(1.558V_{s,i}^{\text{true}}, 1.869V_{s,i}^{\text{true}}). \end{aligned}$$

Notice that the priors on $V_{p,i}$ are expressed in terms of $V_{s,i}$. This is due to the high correlation between these parameters in an attenuating medium.

We implement our WGPT algorithm with $K = 4$ temperatures, with $T_1 = 1, T_2 = 5, T_3 = 25$ and $T_4 = 125$, obtaining $N = 5000$ samples, after a burn-in period of 1000 samples. Each kernel p_i is a RWM algorithm with covariance

$$\Sigma_i = \begin{pmatrix} \Sigma_{\text{source},i} & 0 \\ 0 & \Sigma_{\text{mat}}^2 \end{pmatrix}$$

where $\Sigma_{\text{mat}}^2 = (25)^2 I_{21\times 21}$, and

$$\begin{aligned} \Sigma_{\text{source},1}^{1/2} &= \begin{pmatrix} 20 & 0 \\ 0 & 50 \end{pmatrix}, \ \Sigma_{\text{source},2}^{1/2} = \begin{pmatrix} 100 & 0 \\ 0 & 300 \end{pmatrix}, \\ \Sigma_{\text{source},3}^{1/2} &= \begin{pmatrix} 500 & 0 \\ 0 & 2000 \end{pmatrix}, \ \Sigma_{\text{source},4}^{1/2} = \begin{pmatrix} 5000 & 0 \\ 0 & 5000 \end{pmatrix} \end{aligned}$$

We plot the density of the source in Figure 4.11. There we can see, denoted by the "+" symbols some of the samples obtained from the WGPT algorithm (after re-weighting the samples), and the blue regions represent the density of the concentration of the points. Notice that, as expected, there's a strong concentration of points around the true location of the source ($x_s = 54000$, $z_s = 59500$). Although in practical applications 5000 is certainly a small number of samples for a BIP, we can still see that the resulting density is both mutli-modal and heavily concentrated around an extremely small region of the computational domain; indeed notice that the plot concentrates on an area that is less than 1% of the total computational domain. On the contrary, given this multi-modality and concentration of the source location in the domain, running a RWM algorithm for this problem (with an equivalent number of samples and using Σ_1 as a covariance) results in samples that are not able to identify the region of source location (not shown). Once again, we have seen how these hierarchical methods are well-suited for these types of problems.

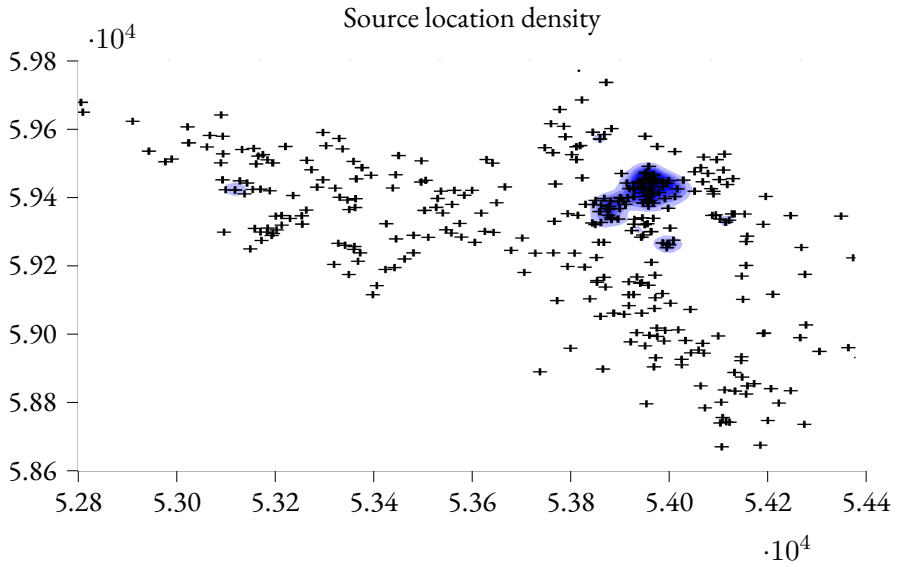

Figure 4.11: Density of source location

5 A class of Multi-Level MCMC algorithms based on Independent Metropolis-Hastings

This chapter is mostly the same as the pre-print J.P. Madrigal-Cianci, F. Nobile, and R. Tempone. *Analysis of a class of Multi-Level Markov Chain Monte Carlo algorithms based on Independent Metropolis-Hastings.* arXiv:2105.02035 (2021) [108]. Some modifications have been made with respect to such a pre-print; some material was removed, as this has already been presented in Chapters 2 and 3 of this thesis. Furthermore, the theoretical analysis has been greatly simplified (following the suggestion of anonymous referees). Furthermore, a challenging, high-dimensional example has been added in Section 5.6.4.

In this work we present, analyze, and implement a class of Multi-Level Markov Chain Monte Carlo (ML-MCMC) algorithms based on independent Metropolis-Hastings proposals for Bayesian inverse problems. In this context, the evaluation of the likelihood function involves solving a complex differential model, which is approximated using a sequence of increasingly accurate discretizations. The key point of this algorithm is to construct highly coupled Markov chains together with the standard multi-level Monte Carlo argument to obtain a better cost-tolerance complexity than a single level MCMC algorithm. Our method extends the ideas of [45] to a wider range of proposal distributions. We present a thorough convergence analysis of the proposed ML-MCMC method and demonstrate that (i) under some mild conditions on the (independent) proposals and family of posteriors, a unique invariant probability measure exists for the coupled chains generated by the proposed method, and (ii) that such coupled chains are uniformly ergodic. We also generalize the cost-tolerance theorem of Dodwell et al., to our wider class of ML-MCMC algorithms. Finally, we propose a self-tuning continuation-type ML-MCMC algorithm (C-ML-MCMC). The presented method is tested on an array of academic examples, where some of our theoretical results are numerically verified. These numerical experiments reveal how the extended ML-MCMC method is robust when targeting some *pathological* posteriors, for which some of the previously proposed ML-MCMC algorithms fail.

5.1 Introduction

Multi-Level Monte Carlo (MLMC) methods are well-known computational techniques [59] used to compute expectations that arise in stochastic simulations in cases in which the stochastic model cannot be simulated exactly, but can be approximated at different levels of accuracy and different computational costs. Despite their wide-spread applicability, extending these MLMC ideas to Multi-Level Markov Chain Monte Carlo (ML-MCMC) methods to compute expectations with respect to a complex target distribution from which independent (whether exact or approximate) sampling is not accessible, has only recently been attempted, with only a handful of works dedicated to this task. This situation arises, for instance, in Bayesian inverse problems (BIPs) where the aim is to compute the expectation $\mathbb{E}_{\mu^y}[\mathsf{Qoi}]$ of some output quantity of interest Qoi with respect to the posterior measure μ^y of some parameters $u \in \mathsf{X}$ given some indirect noise measurements $y = \mathcal{F}(u) + \eta$, where η is the additive noise and $\mathcal{F}$ is the forward operator, which may involve the solution of a differential equation (see Chapter 2 for more details). At their core, ML-MCMC methods for BIPs introduce a hierarchy of discretization levels $\ell = 0, 1, \ldots, \mathsf{L}$ of the underlying forward operator, which induces a family of posterior probability measures μ^y_ℓ, approximating μ^y with increasing levels of accuracy as $\ell \to \infty$. Given some μ^y-integrable quantity of interest Qoi, we can approximate the expectation of Qoi over μ^y by the usual telescoping sum argument of MLMC,

$$\begin{aligned}\mathbb{E}_{\mu^y}[\mathsf{Qoi}] \simeq \mathbb{E}_{\mu^y_{\mathsf{L}}}[\mathsf{Qoi}_{\mathsf{L}}] = \mathbb{E}_{\mu^y_0}[\mathsf{Qoi}_0] + \sum_{\ell=1}^{\mathsf{L}} \left(\mathbb{E}_{\mu^y_\ell}[\mathsf{Qoi}_\ell] - \mathbb{E}_{\mu^y_{\ell-1}}[\mathsf{Qoi}_{\ell-1}]\right) \\ = \sum_{\ell=0}^{\mathsf{L}} \Delta\mathsf{E}_\ell, \end{aligned} \tag{5.1}$$

with $\Delta\mathsf{E}_\ell := \mathbb{E}_{\mu^y_\ell}[\mathsf{Qoi}_\ell] - \mathbb{E}_{\mu^y_{\ell-1}}[\mathsf{Qoi}_{\ell-1}]$, $\Delta\mathsf{E}_0 = \mathbb{E}_{\mu^y_0}[\mathsf{Qoi}_0]$ and where, for $\ell = 0, 1, \ldots, \mathsf{L}$, Qoi_ℓ is a μ^y_ℓ-integrable, level ℓ approximation of the quantity of interest Qoi. This telescoping sum presents the basis for various types of multi-level techniques for BIPs. The work [71], for example, approximates the expectation (5.1) by splitting each $\Delta\mathsf{E}_\ell$ into three different terms, which are then computed using a mixture of importance-sampling and MCMC techniques. A multi-index generalization of such method is presented in [78]. In addition, similar multi-level ideas have also been attempted in the context of Multi-Level Sequential Monte Carlo (MLSMC) in the works [13, 79, 96].

In this work, we follow the approach proposed in [45], which is probably the first proposition of multi-level ideas for BIPs and consists of approximating $\mathbb{E}_{\mu^y_{\mathsf{L}}}[\mathsf{Qoi}_{\mathsf{L}}]$ using the following ergodic estimator:

$$\mathbb{E}_{\mu^y_{\mathsf{L}}}[\mathsf{Qoi}_{\mathsf{L}}] \approx \frac{1}{N_0}\sum_{n=1}^{N_\ell} \mathsf{Qoi}_0(u^{(n)}_{0,0}) + \sum_{\ell=1}^{\mathsf{L}} \frac{1}{N_\ell}\sum_{n=1}^{N_\ell} \underbrace{\mathsf{Qoi}_\ell(u^n_{\ell,\ell}) - \mathsf{Qoi}_{\ell-1}(u^n_{\ell,\ell-1})}_{:= Y^n_\ell},$$

where $\{u^n_{\cdot,\ell}\}_{n=0}^{N_\ell}$ is an ergodic Markov chain with invariant distribution μ^y_ℓ. The key idea is to couple the chains $\{u^n_{\ell,\ell-1}, u^n_{\ell,\ell}\}_{n=0}^{N_\ell}$ so that they are highly correlated and the variance of the ergodic estimator $\mathbb{V}[N_\ell^{-1}\sum_n Y^n_\ell]$ becomes increasingly smaller as ℓ increases. By carefully choosing N_ℓ, this method can achieve a much better sampling complexity (in terms of cost versus tolerance) than its single-level counterparts (see [45]).

Most of the existing literature on ML-MCMC has focused on constructing these types of couplings [35, 45]. In [45], the authors use (an approximation of) the posterior distribution at the previous discretization level $\ell - 1$ as a proposal for level ℓ. This is practically implemented by sub-sampling from the chain $\{u^n_{\ell-1,\ell-1}\}_{n=0}^{N_{\ell-1}}$.

Such an idea has been recently expanded in [35], where the subsampling idea is combined with the so-called Dimension Independent Likelihood Informed (DILI) MCMC method of [36] to generate proposed samples at level 0 in their ML-MCMC algorithm. Some further work combining multi-level Monte Carlo ideas with Bayesian inference has been presented in [80], where the authors use rejection-free Markov transitions kernels, such as the Gibbs sampler, in order to couple the multi-level MCMC chains at two consecutive levels.

However, investigating more theoretical aspects of ML-MCMC algorithms, such as the existence of an invariant measure for the coupled chains and the type of convergence to such a measure (if it exists), has been widely overlooked, and one of the aims of this chapter is to fill this gap.

This work presents several novel contributions. First, we present an ML-MCMC algorithm where chains are coupled using Independent Metropolis Hastings (IMH)-type proposals as in [45], however, allowing for a wider class of admissible proposals. In particular, we show that the sub-sampling approach in [45] can be replaced by a properly chosen IMH proposal (that is, a proposal for which the proposed state is independent of the current state of the chain), which proposes the same state to the two chains $\{u^n_{\ell,\ell-1}, u^n_{\ell,\ell}\}_{n=0}^{N_\ell}$ targeting $\mu^y_{\ell-1}, \mu^y_\ell$ respectively, which is then accepted by the usual Metropolis-Hastings (MH) criterion. This ensures the coupling of the chains. Such a proposal can be, for example, the prior, a Laplace approximation, or even a kernel density approximation of the posterior at the previous level. Obviously, the choice of proposal has a direct influence on the joint invariant distribution ν_ℓ of the coupled chain $\{u^n_{\ell,\ell-1}, u^n_{\ell,\ell}\}_{n=0}^{N_\ell}$ (if it exists), and thus, on the variance of the ergodic estimator $N_\ell^{-1}\sum_n Y^n_\ell$.

The main contribution of this work is an in-depth convergence analysis of the extended ML-MCMC method. More precisely, we provide sufficient conditions on the (marginal) level ℓ posterior and proposal probability measure Q_ℓ so that a unique joint invariant probability measure exists for the coupled chain. Such a contribution is presented in Theorem 5.3.1, where it is shown that, under some mild conditions on $Q_\ell, \mu^y_\ell, \mu^y_{\ell-1}$, the presented ML-MCMC algorithm (i) has a unique, invariant probability measure for the joint chain at level ℓ and (ii) is uniformly ergodic. Following the convergence results presented in Chapter 3, we provide computable, quantitative, non asymptotic error estimators for the ergodic estimator (5.1). These estimators allow us to generalize the cost-tolerance result of [45] to our extended MLMCMC method and propose

an adaptive ML-MCMC algorithm in which the number of levels L and chain lengths N_ℓ are determined on the fly, in the spirit of the continuation MLMC method presented in [31].
The rest of this chapter is organized as follows. In Section 5.2 we present our ML-MCMC method, and then proceed to analyze its convergence in Section 5.3. Section 5.4 is dedicated to the generalization to our case of the cost-tolerance analysis result in [45]. In Section 5.5 we discuss the continuation-type algorithm and implementation details. Lastly, we illustrate our method in several numerical experiments in Section 5.6.

5.2 Multi-level Markov Chain Monte Carlo

Let $(\mathsf{X}, \|\cdot\|_{\mathsf{X}})$ and $(\mathsf{Y}, \|\cdot\|_{\mathsf{Y}})$ be separable Banach spaces with associated Borel σ-algebras $\mathcal{B}(\mathsf{X})$, $\mathcal{B}(\mathsf{Y})$. As in Chapter 2, we consider the BIP of finding the posterior distribution μ^y of some state $u \in \mathsf{X}$ given noisy observations $y \in \mathsf{Y}$ where

$$y = \mathcal{F}(u) + \eta,$$

with $\mathcal{F} : \mathsf{X} \to \mathsf{Y}$ the forward operator and $\eta \sim \mu_{\text{noise}}$ some polluting noise with known distribution μ_{noise} on $(\mathsf{Y}, \mathcal{B}(\mathsf{Y}))$. Furthermore, recall that assuming that u follows a prior probability measure μ_{pr} on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$ before any data has been observed, it can be shown under some technical assumptions (c.f. Chapter 2) that $\mu^y \ll \mu_{\mathsf{pr}}$ with $\mu^y(\mathrm{d}u) = Z^{-1} \exp(-\Phi(u; y))\mu_{\mathsf{pr}}(\mathrm{d}u)$, $Z = \int_{\mathsf{X}} e^{-\Phi(u;y)}\mu_{\mathsf{pr}}(\mathrm{d}u)$, and $\Phi(u; y)$ defined as in (2.6). It is often the case that the forward mapping $u \mapsto \mathcal{F}(u)$ involves the numerical approximation of the underlying mathematical model driving the BIP, and as such, $\mathcal{F}$ needs to be approximated at an accuracy level L, i.e., $\mathcal{F}_{\mathsf{L}} \approx \mathcal{F}$, with $\mathcal{F}_{\mathsf{L}} \to \mathcal{F}$ as $\mathsf{L} \to \infty$. This induces the discretized posterior μ^y_{L}, given in terms of its Radon-Nikodym derivative with respect to the prior by:

$$\pi^y_{\mathsf{L}}(u) := \frac{\mathrm{d}\mu^y_{\mathsf{L}}}{\mathrm{d}\mu_{\mathsf{pr}}}(u) = \frac{1}{Z_{\mathsf{L}}} e^{-\Phi_{\mathsf{L}}(u;y)}, \quad Z_{\mathsf{L}} = \int_{\mathsf{X}} e^{-\Phi_{\mathsf{L}}(u;y)}\mu_{\mathsf{pr}}(\mathrm{d}u),$$

with $\mu^y_{\mathsf{L}} \to \mu^y$ as $\mathsf{L} \to \infty$ in some sense. Throughout this chapter we will assume that $\Phi(u; y), \Phi_{\mathsf{L}}(u; y) \geq 0 \, \forall u \in \mathsf{X}$ and $y \in \mathsf{Y}$. The sampling from the posterior μ^y_{L} will in turn be done using the Metropolis-Hastings algorithm (c.f. 3.4).
In general, such an algorithm requires running the Markov chain for a long time to obtain a good approximation of the posterior, and it is not uncommon for N to be of the order of tens of thousands. Furthermore, such a method requires the evaluation of the posterior density $\pi^y_{\mathsf{L}}(z)$ at each newly proposed state z every time the acceptance rate $\alpha_{\mathsf{L}}(u^n, z)$ in the MH algorithm is evaluated. In PDE-driven BIP, where evaluating $\pi^y_{\mathsf{L}}(z)$ implies solving a possibly non-linear and time-dependent PDE on a sufficiently fine mesh (i.e., with high accuracy), the cost associated with the MH algorithm can rapidly become prohibitive. One technique to alleviate this issue is to introduce multi-level techniques. Thus, we let $\{M_\ell\}_{\ell=0}^{\mathsf{L}}$ be a hierarchy of discretization parameters of the underlying mathematical model $\mathcal{F}(\cdot)$, which could represent, for example, the number of

degrees of freedom used in the discretization of the underlying PDE. We consider only geometric sequences for $\{M_\ell\}_{\ell=0}^{\mathsf{L}}$ with $M_\ell = sM_{\ell-1}$ for some $M_0 > 0$ and $s > 1$. We denote the corresponding discretized forward models by $\mathcal{F}_\ell(\cdot)$ and the corresponding approximate quantity of interest by QoI_ℓ. We assume that the accuracy of the discretization and the cost of evaluating the discretized model, increase as ℓ (and hence M_ℓ) increases. This hierarchy of discretizations induces a hierarchy of posterior probability measures $\{\mu_\ell^y\}_{\ell=0}^{\mathsf{L}}$ approximating μ^y with increasing accuracy and cost. We can write the posterior expectation $\mathbb{E}_{\mu^y}[\mathsf{QoI}]$, approximated on the finest available discretization level L, in terms of the following telescoping sum:

$$\mathbb{E}_{\mu^y}[\mathsf{QoI}] \approx \mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{QoI}_{\mathsf{L}}] = \mathbb{E}_{\mu_0^y}[\mathsf{QoI}_0] + \sum_{\ell=1}^{\mathsf{L}} \left(\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell] - \mathbb{E}_{\mu_{\ell-1}^y}[\mathsf{QoI}_{\ell-1}] \right).$$

This result motivates introducing the following MLMCMC ergodic estimator:

$$\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}} := \frac{1}{N_0} \sum_{n=1}^{N_0} [\mathsf{QoI}_0(u_{0,0}^n)] + \sum_{\ell=1}^{\mathsf{L}} \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} \underbrace{\left(\mathsf{QoI}_\ell(u_{\ell,\ell}^n) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1}^n)\right)}_{:= Y_\ell^n}. \quad (5.2)$$

where we have introduced the notation $u_{\ell,\ell} \sim \mu_\ell^y$ and $u_{\ell,\ell-1} \sim \mu_{\ell-1}^y$, and $u_{\ell,j} = u_{\ell,\ell-1}$ if $j = \ell - 1$ and $u_{\ell,j} = u_{\ell,\ell}$ if $j = \ell$. The terms Y_ℓ^n are generally small if $(u_{\ell,\ell-1}, u_{\ell,\ell})$ are *close*. The key to the method is to design a coupled Markov chain $\{(u_{\ell,\ell-1}^n, u_{\ell,\ell}^n)_{n\geq 0}\}$ for which $u_{\ell,\ell-1}^n$, and $u_{\ell,\ell}^n$ stay highly correlated and close to each other with high probability for every n, while keeping the right marginal invariant distributions $\mu_{\ell-1}^y$, and μ_ℓ^y, respectively. This is necessary for the terms in (5.2) to telescope in the mean. Constructing a coupled Markov chain (with marginal target measures $\mu_{\ell-1}^y, \mu_\ell^y$) for which $\left\| u_{\ell,\ell-1}^n - u_{\ell,\ell}^n \right\|_{\mathsf{X}} \to 0$ in a suitable sense, as $\ell \to \infty$, results in $\mathbb{V}_{\nu_\ell}[Y_\ell] \to 0$ as $\ell \to \infty$, where $\nu_\ell \in \mathcal{M}(\mathsf{X}^2)$ is the invariant measure of the coupled Markov chain (if it exists). Hence, by using an adequate proposal distribution and properly choosing L and $\{N_\ell\}_{\ell=0}^{\mathsf{L}}$ one can obtain a significantly better complexity than that of a single-level MCMC estimator (see [45] for a general complexity result of the ML-MCMC approach). To achieve this, following [45], we will use what we call an *Independent Metropolis-Hastings coupling* (IMH-coupling) of $u_{\ell,\ell-1}, u_{\ell,\ell}$. The main idea of such a coupling is to create two simultaneous Markov Chains $\{u_{\ell,\ell-1}^n, u_{\ell,\ell}^n\}_{n\in\mathbb{N}}$ at two adjacent discretization levels, using as a proposal a probability measure $\tilde{Q}_\ell$ (with $\mu_j^y \ll \tilde{Q}_\ell$ $j = \ell - 1, \ell$), having a (strictly positive) μ_{pr}-density Q_ℓ, in such a way that (i) $\tilde{Q}_\ell$ generates proposed states $z \in \mathsf{X}$ independently of the current state of either chain, and (ii) at every iteration, the same candidate state z is proposed as the new state of both chains, which then accept or reject it using the standard MH accept-reject step with the same uniform random variable $u \sim \mathcal{U}(0,1)$. This will in turn guarantee that, marginally $u_{\ell,j} \sim \mu_j^y$, asymptotically for both $j = \ell - 1$ and $j = \ell$ (i.e., the marginal chains follow the right distribution), and that the pair $(u_{\ell,\ell-1}^n, u_{\ell,\ell}^n)$ is highly correlated for any $n \in \mathbb{N}$, provided the acceptance rate is sufficiently high. A depiction of one step of such a coupling procedure is presented in Algorithm

7. We emphasize that such an algorithm also couples the Metropolisation step by comparing the acceptance probabilities $\alpha_j, j = \ell - 1, \ell$, with respect to the same uniform random number U. The full ML-MCMC procedure is presented in Algorithm 8. At each level $\ell = 1, 2, \ldots, \mathsf{L}$, the coupled chains $\{u^n_{\ell,\ell-1}, u^n_{\ell,\ell}\}_{n=0}^{N_\ell}$ in Algorithm 8 start from the same state $u^0_{\ell,\ell-1} = u^0_{\ell,\ell}$ (the *diagonal* of the set X^2).

Algorithm 7 One-step IMH coupling

procedure IMH_Coupling($\{\pi^y_{\ell-1}, \pi^y_\ell\}, \{u^n_{\ell,\ell-1}, u^n_{\ell,\ell}\}, Q_\ell$)
  Sample $z \sim Q_\ell$.
  Sample $U \sim \mathcal{U}(0,1)$.
  for $j = \ell - 1, \ell$ **do**
    Set $u^{n+1}_{\ell,j} = z$ if $U < \alpha_j(u^n_{\ell,j}, z)$, where

$$\alpha_j(u^n_{\ell,j}, z) := \min\left[1, \frac{\pi^y_j(z)Q_\ell(u^n_{\ell,j})}{\pi^y_j(u^n_{\ell,j})Q_\ell(z)}\right].$$

    Set $u^{n+1}_{\ell,j} = u^n_{\ell,j}$ otherwise.
  end for
  Output $\{u^{n+1}_{\ell,\ell-1}, u^{n+1}_{\ell,\ell}\}$.
end procedure

Algorithm 7 is, effectively a type of independent sampler Metropolis [3] on the marginal chains. As such, the sampling efficiency of such an algorithm critically depends on how well the proposal $\tilde{Q}_\ell$ *approximates* μ^y_ℓ and $\mu^y_{\ell-1}$. Choosing a proposal $\tilde{Q}_\ell$ that closely resembles μ^y_ℓ or $\mu^y_{\ell-1}$ reduces the number of rejection steps, enhancing the mixing of the chains (see [3, 21] for a more in-depth discussion). In principle, $\tilde{Q}_\ell$ can be chosen to be, e.g., the prior, or, an empirical version of the posterior based on the samples $\{u^n_{\ell-1}\}_{n=0}^{N_\ell}$ collected at the previous level, as originally proposed in [45]. It can also be any reasonable approximation of μ^y_ℓ, $\mu^y_{\ell-1}$ such as, e.g., a Laplace approximation or a kernel density estimator (KDE), again based on the sample $\{u^n_{\ell-1}\}_{n=0}^{N_\ell}$ collected at the previous level.

Each step of Algorithm 7 produces 1 out of 4 possible configurations S_1, S_2, S_3, S_4 :

$S_1 : (u^{n+1}_{\ell,\ell-1}, u^{n+1}_{\ell,\ell}) = (z, z)$ (both chains accept the proposed state),

$S_2 : (u^{n+1}_{\ell,\ell-1}, u^{n+1}_{\ell,\ell}) = (z, u^n_{\ell,\ell})$ (chain at level $\ell - 1$ accepts and chain at level ℓ rejects),

$S_3 : (u^{n+1}_{\ell,\ell-1}, u^{n+1}_{\ell,\ell}) = (u^n_{\ell,\ell-1}, z)$ (chain at level $\ell - 1$ rejects and chain at level ℓ accepts),

$S_4 : (u^{n+1}_{\ell,\ell-1}, u^{n+1}_{\ell,\ell}) = (u^n_{\ell,\ell-1}, u^n_{\ell,\ell})$ (both chains reject the proposed state).

Algorithm 8 Multi-level Markov chain Monte Carlo

procedure ML-MCMC($\{\pi_\ell^y\}_{\ell=0}^{\mathsf{L}}, Q, \{N_\ell\}_{\ell=0}^{\mathsf{L}}, \lambda^0$)
  if $\ell = 0$ **then**
    $\{u_{0,0}^n\}_{n=0}^{N_0}$ =`Metropolis-Hastings`$(\pi_0^y, Q, N_0, \lambda^0)$
    Set $\chi_{0,0} = \{u_{0,0}\}_{n=0}^{N_0}$.
  end if
  for $\ell = 1, \ldots, \mathsf{L}$ **do**
    "Construct" Q_ℓ (e.g., from $\chi_{\ell-1,\ell-1}$).
    Sample $u_{\ell,\ell-1}^0 \sim \lambda^0$, and set $u_{\ell,\ell}^0 = u_{\ell,\ell-1}^0$
    for $n = 0, \ldots, N_\ell - 1$ **do**
      # Create a coupled chain using IMH coupling
      $\{u_{\ell,\ell-1}^{n+1}, u_{\ell,\ell}^{n+1}\}$ = `IMH_Coupling`$(\{\pi_{\ell-1}^y, \pi_\ell^y\}, \{u_{\ell,\ell-1}^n, u_{\ell,\ell}^n\}, Q_\ell)$
    end for
    Set $\chi_{\ell,j} = \{u_{\ell,j}^n\}_{n=0}^{N_\ell}, j = \ell - 1, \ell$.
  end for
  Output $\chi_{0,0} \cup \{\chi_{\ell,\ell-1}, \chi_{\ell,\ell}\}_{\ell=1}^{\mathsf{L}}$ and $\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}$.
end procedure

These configurations are illustrated in Figure 5.1. More formally, we set $\mathsf{X}^2 \ni \boldsymbol{u}_\ell^n := (u_{\ell,\ell-1}^n, u_{\ell,\ell}^n)$. Then, for any $A \in \mathcal{B}(\mathsf{X}^2)$, Algorithm 7 induces the *multilevel Markov transition kernel* $\boldsymbol{p}_\ell : \mathsf{X}^2 \times \mathcal{B}(\mathsf{X}^2) \mapsto [0,1]$ given by the following:

$$
\begin{aligned}
\boldsymbol{p}_\ell(\boldsymbol{u}_\ell^n, A) &:= \int_{\mathsf{X}} \min\{\alpha_{\ell-1}(u_{\ell,\ell-1}^n, z), \alpha_\ell(u_{\ell,\ell}^n, z)\} Q_\ell(z) \mathbb{1}_{\{(z,z)\in A\}} \mu_{\mathsf{pr}}(\mathrm{d}z) \\
&+ \int_{\mathsf{X}} (\alpha_{\ell-1}(u_{\ell,\ell-1}^n, z) - \alpha_\ell(u_{\ell,\ell}^n, z))^+ Q_\ell(z) \mathbb{1}_{\left\{(z,u_{\ell,\ell}^n)\in A\right\}} \mu_{\mathsf{pr}}(\mathrm{d}z) \\
&+ \int_{\mathsf{X}} (\alpha_\ell(u_{\ell,\ell}^n, z) - \alpha_{\ell-1}(u_{\ell,\ell-1}^n, z))^+ Q_\ell(z) \mathbb{1}_{\left\{(u_{\ell,\ell-1}^n,z)\in A\right\}} \mu_{\mathsf{pr}}(\mathrm{d}z) \\
&+ \mathbb{1}_{\left\{(u_{\ell,\ell-1}^n,u_{\ell,\ell}^n)\in A\right\}} \left(1 - \int_{\mathsf{X}} \max\{\alpha_{\ell-1}(u_{\ell,\ell-1}^n, z), \alpha_\ell(u_{\ell,\ell}^n, z)\} Q_\ell(z) \mu_{\mathsf{pr}}(\mathrm{d}z)\right),
\end{aligned}
\tag{5.3}
$$

where $(x)^+ := \frac{x+|x|}{2}$, $x \in \mathbb{R}$. Each line on the right-hand side of (5.3) corresponds to the transition kernel proposing to move from the state $\boldsymbol{u}_\ell^n$ to one of the four possible configurations S_i, $i = 1, 2, 3, 4$. Although $\boldsymbol{p}_\ell$ targets the right marginals, the properties related to the convergence of the chain generated by $\boldsymbol{p}_\ell$, such as irreducibility, the existence of an invariant (joint) measure ν_ℓ, or geometric ergodicity, are not obvious. We investigate these convergence properties in the following section.

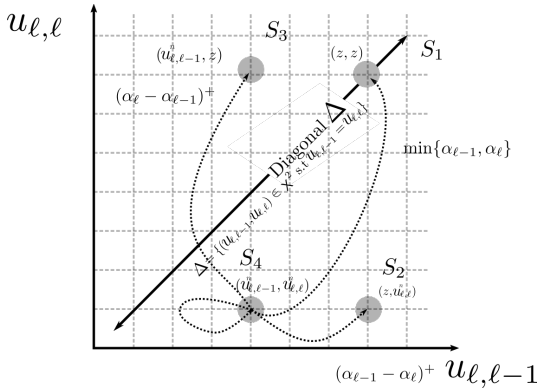

Figure 5.1: Schematic of the possible configurations S_1, S_2, S_3, S_4. The sampler moves to the diagonal $\Delta = \{(u_{\ell,\ell-1}, u_{\ell,\ell}) \in \mathsf{X}^2 \text{ s.t } u_{\ell,\ell-1} = u_{\ell,\ell}\}$ whenever both chains accept (regardless of their current state) or when both chains reject, assuming that they were at the diagonal.

5.3 Convergence analysis of the ML-MCMC algorithm

We now proceed to analyze the convergence of the level-wise coupled chains generated by Algorithm 7. The main result in this section is stated in Theorem 5.3.1. Loosely speaking, this theorem (i) provides conditions for the existence and uniqueness of a joint invariant measure of the multi-level Markov transition kernel (5.3), and (ii) indicates that such a kernel generates a uniformly ergodic chain under certain conditions (i.e., a chain that converges exponentially fast to its invariant distribution with a constant that does not depend on the initial state of the chain).
At each level ℓ, Algorithm 7 creates two coupled chains using the same proposal Q_ℓ, inducing two Markov transition kernels, each generating a marginal chain. We formalize this in the following definition.

Definition 5.3.1 (Marginal kernel): *For a given level ℓ, $\ell = 1, 2 \ldots, \mathsf{L}$ and proposal Q_ℓ, we define the μ_j^y-invariant* marginal Markov transition kernel $p_{\ell,j} : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \to [0,1]$, *with $j = \ell - 1, \ell$, as*

$$p_{\ell,j}(u_{\ell,j}, A) := \int_A \alpha_j(u_{\ell,j}, z) Q_\ell(z) \mu_{\mathsf{pr}}(\mathrm{d}z) + \mathbb{1}_{\{u_{\ell,j} \in A\}} \int_{\mathsf{X}} (1 - \alpha_j(u_{\ell,j}, z))\, Q_\ell(z) \mu_{\mathsf{pr}}(\mathrm{d}z), \tag{5.4}$$

for any $u_{\ell,j} \in \mathsf{X}$, and $A \in \mathcal{B}(\mathsf{X})$. Similarly, we denote its corresponding marginal Markov transition operator *by $P_{\ell,j}$.*

The marginal chains $\{u^n_{\ell,\ell}\}_{n=0}^{N_\ell}$, $\{u^n_{\ell,\ell-1}\}_{n=0}^{N_\ell}$ generated by (5.4) are indeed Markov chains. Furthermore, by construction, $P_{\ell,j}$ is μ_j^y-invariant, (i.e., $\mu_j^y P_{\ell,j} = \mu_j^y$).
We make the following assumptions on the proposal and the (marginal) posterior densities.

Assumption 5.3.1 (Assumptions on proposal and posterior densities): *The following conditions hold for all* $\ell = 1, \dots, \mathsf{L}$*:*

5.3.1.1. *There exists a positive constant* $c \in (0,1)$*, independent of* ℓ*, such that*

$$\operatorname*{ess\,inf}_{z \in \mathsf{X}} \left\{ Q_\ell(z)/\pi_j^y(z) \right\} \geq c > 0, \quad j = \ell - 1, \ell.$$

5.3.1.2. *For any fixed* $y \in \mathsf{Y}$*, the potential function* $\Phi_\ell(\cdot; y) : \mathsf{X} \to \mathbb{R}_+$ *is strictly positive.*

5.3.1.3. *There exist positive constants* $r > 1$*, and* C_r*, independent of* ℓ*, such that* $\int_{\mathsf{X}} Q_\ell^r(u)\mu_{\mathsf{pr}}(\mathrm{d}u) \leq C_r$*, for any* ℓ*.*

Assumption 5.3.1.1 implies that the tails of the proposal Q_ℓ must decay more slowly than those of $\mu_\ell^y, \mu_{\ell-1}^y$ at infinity, (i.e., Q_ℓ has heavier tails than μ_j^y, $j = \ell - 1, \ell$). In practice, this is a moderately restrictive assumption however, it is crucial for the convergence of both the marginal IMH and the multi-level algorithm. Assumption 5.3.1.2 requires the potential to be strictly positive in X (for some fixed $y \in \mathsf{Y}$). This assumption is relatively mild, and will be used in the next Section (c.f. Lemmata 5.4.2 and 5.4.5). Lastly, Assumption 5.3.1.3 is an integrability condition on Q_ℓ with respect to the prior. Just as Assumption 5.3.1.1, this assumption is quite mild and will also become useful in the next section (c.f. Lemma 5.4.6).

5.3.1 Convergence of the level-wise coupled chain

In most MCMC methods, one typically designs a Markov chain with a given invariant probability measure, which automatically ensures the existence of (at least) one invariant probability measure. However, this is not the case for Multi-level MCMC algorithms (including the one presented here), and as such, we now proceed to demonstrate that such an invariant measure uniquely exists. The main result of this subsection is given below.

Theorem 5.3.1: (Uniform ergodicity of the coupled chain) *Suppose that Assumption 5.3.1 holds. Then, for any level* $\ell = 0, 1, 2 \dots, \mathsf{L}$*, there exists a unique invariant probability measure* ν_ℓ *on* $(\mathsf{X}^2, \mathcal{B}(\mathsf{X}^2))$ *for the Markov transition operator* $\boldsymbol{P}_\ell$*. Furthermore, the Markov chain induced by such an operator is uniformly ergodic, i.e.,*

$$\sup_{\|f\|_{L_\infty(\mathsf{X}^2, \mu_{\mathsf{pr}} \times \mu_{\mathsf{pr}})} \leq 1} \left| \int_{\mathsf{X}^2} f(\boldsymbol{u}_\ell') \boldsymbol{p}_\ell{}^n(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell') - \int_{\mathsf{X}^2} f(\boldsymbol{u}_\ell) \nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \right| \leq 2(1 - \rho_\ell)^n, \quad \forall \boldsymbol{u}_\ell \in \mathsf{X}^2, n \in \mathbb{N},$$

with $\rho_\ell := c \int_{\mathsf{X}} \min\left\{ \pi_\ell^y(z), \pi_{\ell-1}^y(z) \right\} \mu_{\mathsf{pr}}(\mathrm{d}z)$*, and* $c \in (0,1)$ *as in Assumption 5.3.1.*

Proof. We begin by showing that the whole space X^2 is a small set. Indeed, notice that for any $(u_{\ell,\ell-1}, u_{\ell,\ell}) = \boldsymbol{u}_\ell \in \mathsf{X}^2$ and $A \in \mathcal{B}(\mathsf{X}^2)$, it follows from Equation (5.3) that

$$\begin{aligned}
\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, A) &\geq \int_{\mathsf{X}} \min\{\alpha_{\ell-1}(u_{\ell,\ell-1}, z), \alpha_\ell(u_{\ell,\ell}, z)\} Q_\ell(z) \mathbb{1}_{\{(z,z)\in A\}} \mu_{\mathsf{pr}}(\mathrm{d}z) \\
&= \int_{\mathsf{X}} \min\left\{1, \frac{\pi^y_{\ell-1}(z)}{Q_\ell(z)} \frac{Q_\ell(u_{\ell,\ell-1})}{\pi^y_{\ell-1}(u_{\ell,\ell-1})}, \frac{\pi^y_\ell(z)}{Q_\ell(z)} \frac{Q_\ell(u_{\ell,\ell})}{\pi^y_\ell(u_{\ell,\ell})}\right\} Q_\ell(z) \mathbb{1}_{\{(z,z)\in A\}} \mu_{\mathsf{pr}}(\mathrm{d}z) \\
&\geq \int_{\mathsf{X}} \min\left\{1, \frac{\pi^y_{\ell-1}(z)}{Q_\ell(z)} c, \frac{\pi^y_\ell(z)}{Q_\ell(z)} c\right\} Q_\ell(z) \mathbb{1}_{\{(z,z)\in A\}} \mu_{\mathsf{pr}}(\mathrm{d}z) \quad \text{(By Assumption 5.3.1)} \\
&\geq c \int_{\mathsf{X}} \min\left\{\frac{\pi^y_{\ell-1}(z)}{Q_\ell(z)}, \frac{\pi^y_\ell(z)}{Q_\ell(z)}\right\} Q_\ell(z) \mathbb{1}_{\{(z,z)\in A\}} \mu_{\mathsf{pr}}(\mathrm{d}z) \\
&= c \int_{\mathsf{X}} \min\left\{\pi^y_\ell(z), \pi^y_{\ell-1}(z)\right\} \mathbb{1}_{\{(z,z)\in A\}} \mu_{\mathsf{pr}}(\mathrm{d}z) =: c\tilde{\nu}_\ell(A),
\end{aligned}$$

where we have set

$$\tilde{\nu}_\ell(A) := \int_{\mathsf{X}} \min\left\{\pi^y_\ell(z), \pi^y_{\ell-1}(z)\right\} \mathbb{1}_{\{(z,z)\in A\}} \mu_{\mathsf{pr}}(\mathrm{d}z).$$

Notice that $\tilde{\nu}_\ell$ defines then a measure on X^2. Thus, since such a minorization condition holds for the whole space, X^2 is a small set and the chain is $\tilde{\nu}_\ell$-irreducible and strongly aperiodic. Setting $\rho_\ell := c\tilde{\nu}_\ell(\mathsf{X}^2)$, it then follows from Theorem 3.2.2 that the Markov chain generated by $\boldsymbol{P}_\ell$ is Harris recurrent, and as such, it admits a unique invariant probability measure ν_ℓ. Lastly, it follows from Theorem 3.2.3 (c.f. also [113, Theorem 16.2.4]) that the chain is uniformly ergodic and

$$\sup_{\|f\|_{L_\infty(\mathsf{X}^2, \mu_{\mathsf{pr}}\times\mu_{\mathsf{pr}})}\leq 1} \left| \int_{\mathsf{X}^2} f(\boldsymbol{u}_\ell{}') \boldsymbol{p}_\ell{}^n(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell{}') - \int_{\mathsf{X}^2} f(\boldsymbol{u}_\ell) \nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \right| \leq 2(1-\rho_\ell)^n, \quad \forall \boldsymbol{u}_\ell \in \mathsf{X}^2, n \in \mathbb{N},$$

with $\rho_\ell \to c$ as $\ell \to \infty$. □

We have demonstrated that the joint chain generated by the multi-level algorithm with independent proposals (i) has an invariant measure and (ii) is uniformly ergodic.
Notice that the previous theorem is closely related to the following standard result in the theory of Markov chains (see, e.g., [111]), and which we recall here for convenience.

Theorem 5.3.2 (Uniform ergodicity of IMH): *For any $\ell = 1, 2, \ldots, \mathsf{L}$ and $j = \ell - 1, \ell$, let $p_{\ell,j} : \mathsf{X} \times \mathcal{B}(\mathsf{X}) \to [0,1]$ denote the μ^y_j-reversible Markov transition kernel associated with an IMH algorithm with proposal Q_ℓ. If Q_ℓ and μ^y_j are such that* $\operatorname{ess\,inf}_{z\in\mathsf{X}} \left\{Q_\ell(z)/\pi^y_j(z)\right\} > 0$, *then $p_{\ell,j}$ is uniformly ergodic. Conversely, if* $\operatorname{ess\,inf}_{z\in\mathsf{X}} \left\{Q_\ell(z)/\pi^y_j(z)\right\} = 0$, *then, $p_{\ell,j}$ fails to be ergodic in the sense of* (3.15).

Proof. See [111, Theorem 2.1]. □

Thus, from Theorem 5.3.2, Assumption 5.3.1.1 also implies uniform ergodicity of the marginal chains of the ML-MCMC algorithm. We remark however, that such a result cannot directly be used instead of our Theorem 5.3.1, since it presumes the existence of an invariant probability measure for the chain.
The choice of Q_ℓ is delicate for the ML-MCMC algorithm to work. For instance, consider the case $\mathsf{L} = 1$, $\mu_0^y = \mathcal{N}(1,1)$ and $\mu_1^y = \mathcal{N}(\frac{1}{2},1)$. What might initially appear to be a good proposal for the coupled chain at level $(\ell-1,\ell) = (0,1)$ is to take $Q_1 = \mu_0^y$, i.e., the (exact) posterior at the previous level. However, this proposal choice (which is unfeasible in practice, as direct sampling from $\mu_{\ell-1}^y$ is inaccessible) does not lead to a geometrically ergodic chain given Theorem 5.3.1, because $Q_1(z)/\pi_1^y(z)$ has essential infimum 0. The idea of proposing from the previous level is somehow what is advocated in [45], which could work only if $\exists c_1, c_2 \in \mathbb{R}_+$ such that $c_1 \leq \pi_{\ell-1}^y(z)/\pi_\ell^y(z) \leq c_2, \forall z \in \mathsf{X}$ and $\forall \ell$.
Lastly, notice that, by construction, the ML-MCMC algorithm 8 starts from a measure $\widehat{\lambda}^0(A) := \lambda^0(A_\Delta)$, $\lambda^0 \ll \mu_{\mathsf{pr}}$, where, for any set $A \in \mathcal{B}(\mathsf{X}^2)$, we define $A_\Delta := \{z \in \mathsf{X} : (z,z) \in A\}$. We now show that, for any level $\ell = 1,2,\ldots,\mathsf{L}$, $\widehat{\lambda}^0 \ll \nu_\ell$.

Theorem 5.3.3 (Absolute continuity of initial measure): *Under the same assumptions as in Theorem 5.3.1, for any level $\ell = 1,2,\ldots,\mathsf{L}$, it holds that $\widehat{\lambda}^0 \ll \nu_\ell$.*

Proof. Let $A \in \mathcal{B}(\mathsf{X}^2)$ be a compact set such that $\nu_\ell(A) = 0$ (the case for the non-compact set is shown later). Furthermore, from the tightness of ν_ℓ, we have that, given some $\epsilon > 0$, there exists a compact $K_\epsilon \in \mathcal{B}(\mathsf{X}^2)$ such that $\nu_\ell(K_\epsilon) \geq 1-\epsilon$. Thus

$$
\begin{aligned}
0 = \nu_\ell(A) &= \int_{\mathsf{X}^2} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, A)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&\geq \int_{\mathsf{X}^2}\int_{A_\Delta} \min_j \left\{ \min\left\{ Q_\ell(z), \frac{\pi_j^y(z)Q_\ell(u_{\ell,j})}{\pi_j^y(u_{\ell,j})} \right\} \right\} \mu_{\mathsf{pr}}(\mathrm{d}z)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&\geq \int_{K_\epsilon}\int_{A_\Delta} \min_j \left\{ \min\left\{ Q_\ell(z), \frac{\pi_j^y(z)Q_\ell(u_{\ell,j})}{\pi_j^y(u_{\ell,j})} \right\} \right\} \mu_{\mathsf{pr}}(\mathrm{d}z)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell). \qquad (5.5)
\end{aligned}
$$

By Assumption 5.3.1 and the compactness of K_ϵ and A, we have that there exists a $c' > 0$ such that $c' \leq \min_j \left\{ \min\left\{ Q_\ell(z), \frac{\pi_j^y(z)Q_\ell(u_{\ell,j})}{\pi_j^y(u_{\ell,j})} \right\} \right\}, \forall \boldsymbol{u}_\ell \in K_\epsilon, \forall z \in A_\Delta$. Then, we obtain the following:

$$
(5.5) \geq c'(1-\epsilon)\mu_{\mathsf{pr}}(A_\Delta),
$$

which implies that $\mu_{\mathsf{pr}}(A_\Delta) = 0$. Moreover, because $\lambda^0 \ll \mu_{\mathsf{pr}}$, we have $\widehat{\lambda}^0(A) = \lambda^0(A_\Delta) = 0$; therefore $\widehat{\lambda}^0 \ll \nu_\ell$. Suppose A is not compact. As $\hat{\lambda}^0$ is a tight probability measure it follows that (see, e.g., [17]),

$$\hat{\lambda}^0(A) = \sup_{\substack{K \subset A \\ K \text{ compact}}} \hat{\lambda}^0(K) = 0,$$

and we can conclude the proof as in the previous case . □

5.3.2 Non-asymptotic bounds on the level-wise ergodic estimator

Recall that given some $q \in [1, \infty]$, the $L_q(\mathsf{X}, \mu)$-*spectral gap* of $P : L_q(\mathsf{X}, \mu) \to L_q(\mathsf{X}, \mu)$ is given by:

$$\gamma_q[P] := 1 - \|P\|_{L_q^0 \to L_q^0}.$$

Whenever $\gamma_q[P] > 0$, $\nu^0 P^n$ converges to μ for any $\nu^0 \in \mathcal{M}(\mathsf{X})$ in some appropriate distance for probability measures (see, e.g., [95, 143]). Recall, furthermore, that the *pseudo-spectral gap* of a given Markov operator $\boldsymbol{P}_\ell : L_2(\mathsf{X}^2, \nu_\ell) \to L_2(\mathsf{X}^2, \nu_\ell)$ is given by:

$$\gamma_{\text{ps}}[\boldsymbol{P}_\ell] := \max_{k \geq 1} \left\{ \gamma_2[(\boldsymbol{P}_\ell^*)^k \boldsymbol{P}_\ell^k]/k \right\}, \; k \in \mathbb{N}, \tag{5.6}$$

where $\boldsymbol{P}_\ell^* : L_{q'}(\mathsf{X}^2 \nu_\ell) \to L_{q'}(\mathsf{X}^2, \nu_\ell)$ is the adjoint operator of $\boldsymbol{P}_\ell$. It is shown in [127, Proposition 3.4] that for a uniformly ergodic chain with Markov kernel, $\boldsymbol{P}_\ell$, it holds that $\gamma_{\text{ps}}[\boldsymbol{P}_\ell] > 0$.

For all $\ell = 1, 2, \ldots, \mathsf{L}$, and for a μ_j^y-integrable quantity of interest QoI_j, $j = \ell - 1, \ell$, we write the following:

$$Y_\ell(\boldsymbol{u}_\ell) := \mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1}), \; \boldsymbol{u}_\ell = (u_{\ell,\ell-1}, u_{\ell,\ell}) \in \mathsf{X}^2,$$

$$\hat{Y}_{\ell,N_\ell} := \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} Y_\ell(\boldsymbol{u}_\ell{}^{n+n_{b,\ell}}), \quad \boldsymbol{u}_\ell^n \sim \boldsymbol{p}_\ell(\boldsymbol{u}_\ell{}^{n-1}, \cdot), \; n_{b,\ell} \in \mathbb{N}.$$

Next, we analyze the level-wise contribution to the ML-MCMC ergodic estimator (5.2), which we write hereafter in more general terms, including a burn-in phase.

For $\ell = 1, 2, \ldots, \mathsf{L}$, let $\mathsf{QoI}_\ell - \mathsf{QoI}_{\ell-1} =: Y_\ell : \mathsf{X}^2 \to \mathbb{R}$ be a ν_ℓ square-integrable function, and ν^0 be a probability measure on $(\mathsf{X}^2, \mathcal{B}(\mathsf{X}^2))$ such that $\nu^0 \ll \nu_\ell$. In addition, denote by $\mathsf{E}_{\nu^0, \boldsymbol{P}_\ell}[Y_\ell]$ (resp $\mathsf{V}_{\nu^0, \boldsymbol{P}_\ell}[Y_\ell]$) the expectation (resp. variance) of Y_ℓ over the Markov chain generated by $\boldsymbol{P}_\ell$, starting from an initial probability measure ν^0, and consider the following ergodic estimator:

$$\hat{Y}_{\ell,N_\ell,n_{b,\ell}} := \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} Y_\ell(\boldsymbol{u}_\ell{}^{n+n_{b,\ell}}), \quad \boldsymbol{u}_\ell^n \sim \boldsymbol{p}_\ell(\boldsymbol{u}_\ell{}^{n-1}, \cdot), \tag{5.7}$$

where $n_{b,\ell} \in \mathbb{N}$ is the usual *burn-in period*. This section provides error bounds on the *non-asymptotic statistical Mean Square Error* (MSE) of (5.7)

$$\begin{aligned}\mathrm{MSE}(\hat{Y}_{\ell,N_\ell,n_{b,\ell}};\nu^0) &:= \mathsf{E}_{\nu^0,\boldsymbol{P}_\ell}\left[\left(\hat{Y}_{\ell,N_\ell,n_{b,\ell}} - \nu_\ell(Y_\ell)\right)^2\right] \\ &= \mathsf{V}_{\nu^0,\boldsymbol{P}_\ell}\left[\hat{Y}_{\ell,N_\ell,n_{b,\ell}}\right] + \left[\mathsf{E}_{\nu^0,\boldsymbol{P}_\ell}[\hat{Y}_\ell] - \nu_\ell(Y_\ell)\right]^2 \end{aligned} \tag{5.8}$$

In particular, we aim to obtain a bound of the following form

$$\mathrm{MSE}(\hat{Y}_{\ell,N_\ell,n_{b,\ell}};\nu^0) \leq C_{\mathrm{mse},\ell}\frac{\mathbb{V}_{\nu_\ell}[Y_\ell]}{N_\ell}, \tag{5.9}$$

for some level-dependent, positive constant $C_{\mathrm{mse},\ell}$. Such a bound is presented in Theorem 5.3.4, the main result of this subsection. A bound of the form (5.9) is required for the cost analysis in Section 5.4. As discussed in Chapter 3, bounds such as (5.9) exist for geometrically ergodic and reversible Markov transition kernels [143]. However, the chain generated by $\boldsymbol{P}_\ell$ is not ν_ℓ-reversible. Consequently, we can not directly apply the nonasymptotic bounds presented in [143]. Instead, inspired by the error analysis of [143] and the *pseudo-spectral* approach of [127], we construct a bound of the form (5.9) for general (i.e., not necessarily multi-level) nonreversible, discrete-time Markov chains. To the best of the authors' knowledge, this result is new.

Theorem 5.3.4 (Nonasymptotic bound on the mean square error): *Suppose Assumption 5.3.1 holds. Furthermore, for any $\ell = 1, 2, \ldots, \mathsf{L}$, let $Y_\ell \in L_2(\mathsf{X}^2, \nu_\ell)$, and write $g_\ell(\boldsymbol{u}_\ell) = Y_\ell(\boldsymbol{u}_\ell) - \int_{\mathsf{X}^2} Y_\ell(\boldsymbol{u}_\ell)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell)$, and assume the Markov chain generated by $\boldsymbol{P}_\ell$ starts from a measure ν^0 with $\nu^0 \ll \nu_\ell$, and $\frac{\mathrm{d}\nu^0}{\mathrm{d}\nu_\ell} \in L_\infty(\mathsf{X}^2, \nu_\ell)$. Then,*

$$\mathrm{MSE}(\hat{Y}_{\ell,N_\ell,n_{b,\ell}};\nu^0) := \mathsf{E}_{\nu^0,\boldsymbol{P}_\ell}\left|\frac{1}{N_\ell}\sum_{n=1}^{N_\ell} g_\ell(\boldsymbol{u}_\ell{}^{n+n_{b,\ell}})\right|^2 \leq C_{\mathrm{mse},\ell}\frac{\mathbb{V}_{\nu_\ell}[Y_\ell]}{N_\ell}, \tag{5.10}$$

where $C_{\mathrm{mse},\ell} = C_{\mathrm{inv},\ell} + C_{\mathrm{ns},\ell}$, with

$$C_{\mathrm{inv},\ell} = \left(1 + \frac{4}{\gamma_{\mathrm{ps}}[\boldsymbol{P}_\ell]}\right), \quad C_{\mathrm{ns},\ell} = \left(2\left\|\frac{\mathrm{d}\nu^0}{\mathrm{d}\nu_\ell} - 1\right\|_{L_\infty}\left(1 + \frac{4}{\gamma_{\mathrm{ps}}[\boldsymbol{P}_\ell]}\right)\right),$$

where $\gamma_{\mathrm{ps}}[\boldsymbol{P}_\ell]$ is the pseudo-spectral gap of $\boldsymbol{P}_\ell$, defined in (5.6).

Proof. This is an application of Theorem 3.3.2. □

Remark 5.3.1: *Notice that Assumption $\nu^0 \ll \nu_\ell$ holds in our setting by Theorem 5.3.3 for $\nu^0(A) = \lambda^0(A_\Delta)$.*

Moreover, although constants $C_{\mathrm{inv},\ell}$ and $C_{\mathrm{ns},\ell}$ depend on the level ℓ, we do not expect them to degenerate as $\ell \to \infty$. In particular, the dependency on the level is given by two terms: $\gamma_{\mathrm{ps}}[\boldsymbol{P}_\ell]$

and $\left\| \frac{\mathrm{d}\nu^0}{\mathrm{d}\nu_\ell} - 1 \right\|_{L_\infty}$. For the first term, we expect $\gamma_{\text{ps}}[\boldsymbol{P}_\ell]$ to become smaller and smaller as $\ell \to \infty$ and for it to converge to a limit value $\gamma_{\text{ps}}[\boldsymbol{P}_\infty] > 0$ (see also the discussion of synchronization of the coupled chains in Section 5.4). For the second term $\left\| \frac{\mathrm{d}\nu^0}{\mathrm{d}\nu_\ell} - 1 \right\|_{L_\infty}$, notice that ν_ℓ converges to a measure that has all of its mass in the diagonal set of X^2. Because ν^0 is a finite measure on such a diagonal, we also expect that this term remains bounded as $\ell \to \infty$. However, we are not able to prove these claims at the moment, thus we formulate the following assumption.

Assumption 5.3.2: *There exist a level independent constant C_{mse} such that, for any $\ell = 0, 1, \ldots,$ it holds that $C_{\text{mse},\ell} < C_{\text{mse}}$.*

The fact that $C_{\text{mse},\ell}$ does not blow-up as $\ell \to \infty$ is an important requirement on the asymptotic analysis of ML-(MC)MC methods.
The bound (5.10) should be compared to the bound presented in [143, Theorem 3.34]. In particular, that work presents a sharper bound than (5.10), however, such a bound necessitates more restrictive assumptions which we list in the next theorem for completeness, whose proof is an easy adaptation of [143, Theorem 3.34] to our setting and is omitted.

Theorem 5.3.5: *Suppose that the Assumptions of Theorem 5.3.4 hold. In addition, assume that for any $\ell = 1, 2, \ldots, \mathsf{L}$:*

R1. *(L_2-spectral gap) there exists $b_\ell \in (0, 1)$ such that*

$$\|\boldsymbol{P}_\ell\|_{L_2^0(\mathsf{X}^2,\nu_\ell)\to L_2^0(\mathsf{X}^2,\nu_\ell)} < b_\ell,$$

R2. *(L_1-exponential convergence) there exists $\tilde{c}_\ell \in \mathbb{R}_+$, $a_\ell \in (0, 1)$ such that*

$$\left\|\nu^0 \boldsymbol{P}_\ell^n - \nu_\ell\right\|_{L_1(\mathsf{X}^2,\nu_\ell)} := \left\| \frac{\mathrm{d}(\nu^0 \boldsymbol{P}_\ell^n)}{\mathrm{d}\nu_\ell} - 1 \right\|_{L_1(\mathsf{X}^2,\nu_\ell)} \leq \tilde{c}_\ell a_\ell^n,$$

Then, the non-asymptotic MSE is given by

$$\mathsf{E}_{\nu^0,\boldsymbol{P}_\ell}\left| \frac{1}{N_\ell} \sum_{n=1}^{N_\ell} g_\ell(\boldsymbol{u}_\ell{}^{n+n_{b,\ell}}) \right|^2 \leq \frac{\mathbb{V}_{\nu_\ell}[Y_\ell]}{N_\ell} \left(\frac{2}{(1-b_\ell)} + \frac{2\tilde{c}_\ell \left\| \frac{\mathrm{d}\nu^0}{\mathrm{d}\nu_\ell} - 1 \right\|_{L_\infty} a_\ell^{n_{b,\ell}}}{N_\ell (1-a_\ell)^2} \right), \quad (5.11)$$

where the first term in the parenthesis is associated with the variance contribution to the MSE, whereas the second term corresponds to the statistical squared bias and is of higher order in N_ℓ.

In general, the stronger Assumptions R1 and R2 are known to hold for Markov chains which are both reversible and geometrically ergodic. However, due to its construction, the Markov transition kernel $\boldsymbol{P}_\ell$ of the ML-MCMC algorithm is not reversible. Nevertheless, we believe that the presented algorithm satisfies Assumptions R1 and R2 and as such, a bound on the MSE of the

form (5.11), should hold. However, we are currently unable to verify this claim either, and we will restrict to the bound of Theorem 5.3.4 and the less restrictive Assumption 5.3.2.

5.4 Cost analysis of the ML-MCMC algorithm

For $\ell = 0, 1, \dots, \mathsf{L}$, let $\mathsf{Qol}_\ell : \mathsf{X} \mapsto \mathbb{R}$ be a μ_ℓ^y-integrable quantity of interest, denote by $\mathbb{E}_{\mathrm{ML}}$ (resp. $\mathbb{V}_{\mathrm{ML}}$) the sample mean (resp. variance) of the multi-level ergodic estimator (5.2), and denote by $\mathsf{E} = \mathsf{E}_{\nu^0, \boldsymbol{P}_\ell}$ (resp. $\mathsf{V} = \mathsf{V}_{\nu^0, \boldsymbol{P}_\ell}$) the sample mean (resp. variance) with respect to Markov chain generated by a ν_ℓ-invariant Markov kernel $\boldsymbol{P}_\ell$, started form an initial measure ν^0. The *Total Mean Square Error* of the multi-level estimator (5.2) is given by the following:

$$\hat{\mathrm{e}}_{\mathrm{ML}}(\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}) := \mathbb{E}_{\mathrm{ML}}\left[\left(\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}} - \mathbb{E}_{\mu^y}[\mathsf{Qol}]\right)^2\right].$$

Notice that the estimator $\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}$ also depends on $\{\boldsymbol{P}_\ell\}_{\ell=1}^{\mathsf{L}}$, the burn-in, and initial measure for each level; however, for the sake of readability, we opted not to write these dependencies explicitly throughout this section. The previous term can be upper bounded by

$$\begin{aligned}\hat{\mathrm{e}}_{\mathrm{ML}}(\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}) &= \mathbb{V}_{\mathrm{ML}}[\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}] + \left[\mathbb{E}_{\mathrm{ML}}\left[\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}\right] - \mathbb{E}_{\mu^y}[\mathsf{Qol}]\right]^2 \\ &\leq \underbrace{\mathbb{V}_{\mathrm{ML}}[\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}]}_{\text{Variance contr.}} + 2\underbrace{\left[\mathbb{E}_{\mathrm{ML}}\left[\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}\right] - \mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{Qol}_{\mathsf{L}}]\right]^2}_{\text{MCMC bias contr.}} + 2\underbrace{\left[\mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{Qol}_{\mathsf{L}}] - \mathbb{E}_{\mu^y}[\mathsf{Qol}]\right]^2}_{\text{Discretization contr.}}.\end{aligned}$$

Notice that

$$\mathbb{V}_{\mathrm{ML}}[\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}] = \sum_{\ell=0}^{\mathsf{L}} \mathsf{V}[\hat{Y}_\ell] + 2\sum_{0\leq\ell\leq\ell'\leq\mathsf{L}} \mathrm{Cov}(\hat{Y}_\ell, \hat{Y}_{\ell'}) \leq 2(\mathsf{L}+1)\sum_{\ell=0}^{\mathsf{L}} \mathsf{V}[\hat{Y}_\ell].$$

Furthermore, we have that

$$2\left[\mathbb{E}_{\mathrm{ML}}\left(\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}\right) - \mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{Qol}]\right]^2 \leq 2(\mathsf{L}+1)\sum_{\ell=0}^{\mathsf{L}}\left(\mathsf{E}[\hat{Y}_\ell] - \mathbb{E}_{\nu_\ell}[Y_\ell]\right)^2.$$

Thus, recognizing from Equation (5.8) the level-wise (statistical) MSE of $\hat{Y}_\ell$ as

$$\mathrm{MSE}(\hat{Y}_\ell) = \mathsf{V}[\hat{Y}_\ell] + \left(\mathsf{E}[\hat{Y}_\ell] - \mathbb{E}_{\nu_\ell}[Y_\ell]\right)^2,$$

we then have that

$$\hat{\mathrm{e}}_{\mathrm{ML}}(\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}) \leq$$
$$\underbrace{2(\mathsf{L}+1)\sum_{\ell=0}^{\mathsf{L}} \mathrm{MSE}(\hat{Y}_\ell)}_{\text{Total statistical error}} + \underbrace{2\left[\mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{QoI}_{\mathsf{L}}] - \mathbb{E}_{\mu^y}[\mathsf{QoI}]\right]^2}_{\text{Discretization error}} =: \mathrm{e}_{\mathrm{ML}}(\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}). \tag{5.12}$$

For some tolerance $\mathsf{tol} > 0$, we denote the minimal computational cost required to obtain $\mathrm{e}_{\mathrm{ML}}(\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}) \leq \mathsf{tol}^2$ by $\mathcal{C}\left(\mathrm{e}_{\mathrm{ML}}\left(\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}\right), \mathsf{tol}^2\right)$. The focus of this section is to provide upper bounds on this computational cost, while quantifying the computational advantage of the ML-MCMC method over its single-level counter part (at level L). In particular, our result can be thought of as an extension of [45, Theorem 3.4]. The main result of this section is presented in Theorem 5.4.1. To establish a cost-tolerance relation, we must first make assumptions on the decay of the discretization error and the corresponding increase in computational cost for the evaluation of $\mathcal{F}_\ell$ as a function of the discretization parameter $M_\ell = s^\ell M_0$.

Assumption 5.4.1: *For any $\ell \geq 0$, the following hold:*

5.4.1.1. *There exist positive functions $C_{\mathcal{F}}, C_\Phi : \mathsf{X} \to \mathbb{R}_+$ independent of ℓ, and positive constants C_{e}, α independent of u and ℓ such that*

a) $\|\mathcal{F}_\ell(u) - \mathcal{F}(u)\|_{\mathsf{Y}} \leq C_{\mathcal{F}}(u) s^{-\alpha\ell}, \ \forall u \in \mathsf{X}.$

b) $|\Phi_\ell(u;y) - \Phi(u;y)| \leq C_\Phi(u) \|\mathcal{F}_\ell(u) - \mathcal{F}(u)\|_{\mathsf{Y}}, \ \forall u \in \mathsf{X},$

c) $\int_{\mathsf{X}} \exp(C_{\mathcal{F}}(u) C_\Phi(u)) \mu_{\mathsf{pr}}(\mathrm{d}u) \leq C_{\mathrm{e}} < \infty.$

5.4.1.2. *Given a μ_ℓ^y-integrable quantity of interest QoI_ℓ, there exits a function $C_q : \mathsf{X} \to \mathbb{R}_+$ independent of ℓ and positive constants $\tilde{C}_q, \alpha_q, C_{\mathrm{m}}$, and $m > 2$, independent of u and ℓ such that*

a) $|\mathsf{QoI}_\ell(u) - \mathsf{QoI}(u)| \leq C_q(u) s^{-\alpha_q \ell}, \ \forall u \in \mathsf{X}.$

b) $\int_{\mathsf{X}} C_q^2(u) \mu_{\mathsf{pr}}(\mathrm{d}u) \leq \tilde{C}_q^2 < \infty.$

c) $\left(\int_{\mathsf{X}} |\mathsf{QoI}_\ell(u)|^m \mu_{\mathsf{pr}}(\mathrm{d}u)\right)^{1/m} \leq C_{\mathrm{m}} < \infty.$

5.4.1.3. *There exist positive constants γ and C_γ, such that, for each discretization level ℓ, the computational cost of obtaining one sample from a μ_ℓ^y-integrable quantity of interest $\mathsf{QoI}_\ell(u_{\ell,\ell})$, $u_{\ell,\ell} \sim \mu_\ell^y$, with $u_{\ell,\ell}$ generated by Algorithm 7, denoted by $\mathcal{C}_\ell(\mathsf{QoI}_\ell)$, scales as*

$$\mathcal{C}_\ell(\mathsf{QoI}_\ell) \leq C_\gamma s^{\gamma\ell}.$$

Remark 5.4.1: *With a slight abuse of notation, we have used the symbol α to denote the (strong) rate in 5.4.1.1, and $\alpha_\ell(\cdot,\cdot)$ to denote acceptance probability at level ℓ. We hope this does not create any confusion.*

We state the main result of this section.

Theorem 5.4.1 (Decay of errors): *For any $\ell = 0, 1, \ldots, \mathsf{L}$, let QoI_ℓ be an $L_1(\mathsf{X}, \mu_\ell^y)$-integrable quantity of interest and suppose Assumptions 5.3.1, 5.3.2, and 5.4.1 hold. Then, there exist positive constants C_w, C_v, C_{mse}, independent of ℓ such that:*

T1. (Weak convergence) $\left|\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell] - \mathbb{E}_{\mu^y}[\mathsf{QoI}]\right| \leq C_w s^{-\alpha_w \ell}$,

T2. (Strong convergence) $\mathbb{V}_{\nu_\ell}[Y_\ell] \leq C_v s^{-\beta \ell}$.

T3. (MSE bound) $\text{MSE}(\hat{Y}_{\ell,N_\ell}) \leq N_\ell^{-1} C_{\text{mse}} \mathbb{V}_{\nu_\ell}[Y_\ell]$.

Here, $\alpha_w = \min\{\alpha_q, \alpha\}$ and $\beta = \min\{2a_q, \alpha(1-2/m)\}$, with α, α_q, and m as in Assumption 5.4.1.

The proof of Theorem 5.4.1 is presented in Section 5.4.1. In [45, Theorem 3.4], it has been shown that, if an ML-MCMC algorithm satisfies conditions T1-T3, then it has a complexity (cost-tolerance relation) analogous to a standard MLMC algorithm to compute expectations (when independent sampling from the underlying probability measure is possible) up to logarithmic terms. This result is stated in Theorem 5.4.2 below. The purpose of Theorem 5.4.1 is to demonstrate that our class of ML-MCMC algorithms does actually fulfill conditions T1-T3.

Remark 5.4.2: *Throughout this work, we have the tacit assumption that the chain at level 0 (i.e., the one that does not require an IMH sampler, is geometrically ergodic with respect to μ_0^y).*

Theorem 5.4.2: ([45, Theorem 3.4]) *Under the same assumptions as in Theorem 5.4.1, with $\alpha_w \geq \frac{1}{2}\min\{\gamma, \beta\}$, for any $\mathsf{tol} > 0$ there exist a number of levels $\mathsf{L} = \mathsf{L}(\mathsf{tol})$, a decreasing sequence of integers $\{N_\ell(\mathsf{tol})\}_{\ell=0}^{\mathsf{L}}$, and a positive constant C_{ML} independent of tol, such that the* MSE *bound of the multilevel estimator, $\mathrm{e}_{\text{ML}}(\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}})$, satisfies*

$$\mathrm{e}_{\text{ML}}\left(\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}\right) \leq \mathsf{tol}^2,$$

whereas, the corresponding total ML-MCMC cost is bounded by

$$\mathcal{C}\left(\mathrm{e}_{\text{ML}}\left(\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}\right), \mathsf{tol}^2\right) \leq C_{\text{ML}} \begin{cases} \mathsf{tol}^{-2}|\log \mathsf{tol}| & if\ \beta > \gamma, \\ \mathsf{tol}^{-2}|\log \mathsf{tol}|^3, & if\ \beta = \gamma, \\ \mathsf{tol}^{-2-(\gamma-\beta)/\alpha_w}|\log \mathsf{tol}|, & if\ \beta < \gamma. \end{cases}$$

Proof. See [45]. □

The rates in Theorem 5.4.2 are independent of the dimension of X. In [45] it is shown that the cost of obtaining an equivalent single-level (at level L) mean square error of an estimator $\widehat{\mathsf{QoI}}_N$

based on a single-level MCMC algorithm (e.g., standard MH) (denoted by e_{SL}), generated by a reversible and geometrically ergodic Markov kernel is given by

$$\mathcal{C}\left(\mathrm{e}_{\mathrm{SL}}\left(\widehat{\mathsf{QoI}}_N\right), \mathsf{tol}^2\right) \leq C_{\mathrm{SL}}\mathsf{tol}^{-2-\gamma/\alpha_w}, \quad C_{\mathrm{SL}} \in \mathbb{R}_+,$$

where α_w and γ are the same constants as in Theorem 5.4.1, and C_{SL} is a positive constant independent of the tolerance tol.

5.4.1 Proof of Theorem 5.4.1

We decompose the proof of Theorem 5.4.1 in a series of auxiliary results. Further, T3 is obtained from Theorem 5.3.4 with a level dependent constant and we postulated in Assumption 5.3.2 that this constant can be bounded by a finite, level-independent constant C_{mse}, and as such, we can use it in T3. Thus, we just need to prove that T1 and T2 hold, which is done in Lemmata 5.4.3 and 5.4.7. We first prove some auxiliary results needed to prove implication T1.

Lemma 5.4.1: *Suppose* Assumption 5.4.1 *holds. Then, for* $\ell = 1, 2, \ldots, \mathsf{L}$ *it holds*

$$c_I \leq Z_\ell \leq C_e,$$

where $c_I = \int_{\mathsf{X}} \exp(-\Phi(u;y) - C_{\mathcal{F}}(u)C_\Phi(u))\mu_{\mathsf{pr}}(\mathrm{d}u)$ *and* C_e *as in* Assumption 5.4.1.

Proof. From Assumption 5.4.1.1, for all $\ell \geq 0$, and $u \in \mathsf{X}$,

$$\Phi(u;y) - C_\Phi(u)C_{\mathcal{F}}(u) \leq \Phi_\ell(u;y) \leq \Phi(u;y) + C_\Phi(u)C_{\mathcal{F}}(u).$$

Hence,

$$\begin{aligned} Z_\ell &= \int_{\mathsf{X}} \exp(-\Phi_\ell(u;y))\mu_{\mathsf{pr}}(\mathrm{d}u) \leq \int_{\mathsf{X}} \exp\left(-(\Phi(u;y) - C_\Phi(u)C_{\mathcal{F}}(u))\right)\mu_{\mathsf{pr}}(\mathrm{d}u) \\ &\leq \int_{\mathsf{X}} \exp(C_\Phi(u)C_{\mathcal{F}}(u))\mu_{\mathsf{pr}}(\mathrm{d}u) = C_e, \end{aligned}$$

where the last step follows from the assumption of nonnegativity of $\Phi(\theta;y)$. Similarly, $Z_\ell \geq \int_{\mathsf{X}} \exp(-\Phi(u;y) - C_\Phi(u)C_{\mathcal{F}}(u))\mu_{\mathsf{pr}}(\mathrm{d}u) = c_I$, independently of ℓ. □

Lemma 5.4.2: *Suppose* Assumption 5.4.1 *holds. Then, for any* $\ell \geq 1$, *there exist positive functions* $C_{\pi,\ell}(u) : \mathsf{X} \to \mathbb{R}_+$, $\tilde{C}_{\pi,\ell}(u) : \mathsf{X} \to \mathbb{R}_+$, *such that*

$$|\pi_\ell^y(u) - \pi_{\ell-1}^y(u)| \leq C_{\pi,\ell}(u)s^{-\alpha\ell}, \quad \forall u \in \mathsf{X}, \tag{5.13}$$

$$|\pi_\ell^y(u) - \pi^y(u)| \leq \tilde{C}_{\pi,\ell}(u)s^{-\alpha\ell}, \quad \forall u \in \mathsf{X}. \tag{5.14}$$

Moreover, $C_{\pi,\ell}(u) = (\pi^y_\ell(u) + \pi^y_{\ell-1}(u))K_{\pi,\ell}(u)$, $\tilde{C}_{\pi,\ell}(u) = (\pi^y_\ell(u) + \pi^y(u))\tilde{K}_{\pi,\ell}(u)$, *with*

$$\begin{aligned}\tilde{K}_{\pi,\ell}(u) &= C_\Phi(u)C_{\mathcal{F}}(u) + c_I^{-1}C_{\mathrm{e}},\\ K_{\pi,\ell}(u) &= (1+s^\alpha)\tilde{K}_{\pi,\ell}(u).\end{aligned}$$

Furthermore, for any $p \in [1, +\infty)$,

$$\begin{aligned}K_p &:= \left(\int_{\mathsf{X}} |K_{\pi,\ell}(u)|^p \mu_{\mathsf{pr}}(\mathrm{d}u)\right)^{1/p} < +\infty,\\ \tilde{K}_p &:= \left(\int_{\mathsf{X}} |\tilde{K}_{\pi,\ell}(u)|^p \mu_{\mathsf{pr}}(\mathrm{d}u)\right)^{1/p} < +\infty.\end{aligned}$$

Proof. We begin with the proof of (5.13). We consider first the case $\Phi_\ell(u;y) \leq \Phi_{\ell-1}(u;y)$.

$$\begin{aligned}|\pi^y_\ell(u) - \pi^y_{\ell-1}(u)| &= \left|\frac{e^{-\Phi_\ell(u;y)}}{Z_\ell} - \frac{e^{-\Phi_{\ell-1}(u;y)}}{Z_{\ell-1}}\right|\\ &\leq \underbrace{\left|\frac{e^{-\Phi_\ell(u;y)}}{Z_\ell} - \frac{e^{-\Phi_{\ell-1}(u;y)}}{Z_\ell}\right|}_{I} + \underbrace{\left|\frac{e^{-\Phi_{\ell-1}(u;y)}}{Z_\ell} - \frac{e^{-\Phi_{\ell-1}(u;y)}}{Z_{\ell-1}}\right|}_{II}.\end{aligned}$$

We first focus on I. A straightforward application of the mean value theorem (c.f. Assumption 5.3.1.2) results in the following:

$$\left|e^{-\Phi_\ell(u;y)} - e^{-\Phi_{\ell-1}(u;y)}\right| \leq e^{-\Phi_\ell(u;y)}|\Phi_\ell(u;y) - \Phi_{\ell-1}(u;y)|. \tag{5.15}$$

Thus, from (5.15), together with Assumptions 5.4.1.1 we have the following:

$$\begin{aligned}I = Z_\ell{}^{-1}\left|e^{-\Phi_\ell(u;y)} - e^{-\Phi_{\ell-1}(u;y)}\right| &\leq \pi^y_\ell(u)|\Phi_\ell(u;y) - \Phi_{\ell-1}(u;y)|\\ &\leq \pi^y_\ell(u)\left(|\Phi_\ell(u;y) - \Phi(u;y)| + |\Phi_{\ell-1}(u;y) - \Phi(u;y)|\right)\\ &\leq \pi^y_\ell(u)C_\Phi(u)\left(\|\mathcal{F}_\ell(u) - \mathcal{F}(u)\|_{\mathsf{Y}} + \|\mathcal{F}_{\ell-1}(u) - \mathcal{F}(u)\|_{\mathsf{Y}}\right)\\ &\leq \pi^y_\ell(u)C_\Phi(u)C_{\mathcal{F}}(u)\left(1+s^\alpha\right)s^{-\alpha\ell}.\end{aligned} \tag{5.16}$$

We shift our attention to II. Following a similar procedure as for I, we have the following:

$$\begin{aligned} II &\leq \frac{\pi^y_{\ell-1}(u)}{Z_\ell} \int_{\mathsf{X}} \left| e^{-\Phi_\ell(z;y)} - e^{-\Phi_{\ell-1}(z;y)} \right| \mu_{\mathsf{pr}}(\mathrm{d}z) \\ &\leq \frac{\pi^y_{\ell-1}(u)}{Z_\ell} \int_{\mathsf{X}} e^{-\min\{\Phi_\ell(z;y),\Phi_{\ell-1}(z;y)\}} \left|\Phi_\ell(z;y) - \Phi_{\ell-1}(z;y)\right| \mu_{\mathsf{pr}}(\mathrm{d}z) \\ &\leq \pi^y_{\ell-1}(u)\,(1+s^\alpha)\, s^{-\alpha\ell} c_I^{-1} \int_{\mathsf{X}} C_\Phi(z) C_{\mathcal{F}}(z) e^{-\min\{\Phi_\ell(z;y),\Phi_{\ell-1}(z;y)\}} \mu_{\mathsf{pr}}(\mathrm{d}z) \\ &\leq \pi^y_{\ell-1}(u)\,(1+s^\alpha)\, s^{-\alpha\ell} c_I^{-1} \int_{\mathsf{X}} C_\Phi(z) C_{\mathcal{F}}(z) \mu_{\mathsf{pr}}(\mathrm{d}z) \\ &\leq \pi^y_{\ell-1}(u)\,(1+s^\alpha)\, s^{-\alpha\ell} c_I^{-1} C_e, \end{aligned} \tag{5.17}$$

where in the last step we used the fact that

$$\int_{\mathsf{X}} C_\Phi(u) C_{\mathcal{F}}(u) \mu_{\mathsf{pr}}(\mathrm{d}u) \leq \int_{\mathsf{X}} \exp(C_\Phi(u) C_{\mathcal{F}}(u)) \mu_{\mathsf{pr}}(\mathrm{d}u) \leq C_{\mathrm{e}}.$$

Adding (5.16) and (5.17) provides the desired result with

$$C'_{\pi,\ell}(u) = \left(\pi^y_\ell(u) C_\Phi(u) C_{\mathcal{F}}(u) + \pi^y_{\ell-1}(u) c_I^{-1} C_e\right)(1+s^\alpha).$$

The case $\Phi_\ell(u;y) > \Phi_{\ell-1}(u;y)$ can be treated analogously by considering the alternative splitting

$$|\pi^y_\ell(u) - \pi^y_{\ell-1}(u)| \leq \left(\left| \frac{e^{-\Phi_\ell(u;y)}}{Z_\ell} - \frac{e^{-\Phi_\ell(u;y)}}{Z_{\ell-1}} \right| + \left| \frac{e^{-\Phi_\ell(u;y)}}{Z_{\ell-1}} - \frac{e^{-\Phi_{\ell-1}(u;y)}}{Z_{\ell-1}} \right| \right),$$

which yields the constant $C''_{\pi,\ell}(u) = \left(\pi^y_{\ell-1}(u) C_\Phi(u) C_{\mathcal{F}}(u) + \pi^y_\ell(u) c_I^{-1} C_e\right)(1+s^\alpha)$. Thus, one can obtain the desired bound $|\pi^y_\ell(u) - \pi^y_{\ell-1}(u)| \leq C_{\pi,\ell}(u) s^{-\alpha\ell}$ with

$$\begin{aligned} C_{\pi,\ell}(u) &= (\pi^y_{\ell-1}(u) + \pi^y_\ell(u)) K_{\pi,\ell}(u), \\ K_{\pi,\ell}(u) &= (C_\Phi(u) C_{\mathcal{F}}(u) + c_I^{-1} C_{\mathrm{e}})(1+s^\alpha). \end{aligned}$$

A similar procedure reveals that the bound (5.14) holds with

$$\begin{aligned} \tilde{C}_{\pi,\ell}(u) &= (\pi^y(u) + \pi^y_\ell(u)) \tilde{K}_{\pi,\ell}(u), \\ \tilde{K}_{\pi,\ell}(u) &= C_\Phi(u) C_{\mathcal{F}}(u) + c_I^{-1} C_{\mathrm{e}}. \end{aligned}$$

Finally,

$$K_p := \left(\int_{\mathsf{X}} |K_{\pi,\ell}(u)|^p \mu_{\mathsf{pr}}(\mathrm{d}u)\right)^{1/p} = (1+s^\alpha)\left(\int_{\mathsf{X}} (C_\Phi(u)C_{\mathcal{F}}(u) + c_I^{-1}C_e)^p \mu_{\mathsf{pr}}(\mathrm{d}u)\right)^{1/p}$$
$$\leq (1+s^\alpha)\left(\frac{p}{e}\right)\left(\int_{\mathsf{X}} \exp\left\{C_\Phi(u)C_{\mathcal{F}}(u) + c_I^{-1}C_e\right\}\mu_{\mathsf{pr}}(\mathrm{d}u)\right)^{1/p} \quad (\text{using } x^p \leq \left(\frac{p}{e}\right)^p e^x)$$
$$\leq (1+s^\alpha)\left(\frac{p}{e}\right)\left(C_e \exp\left\{c_I^{-1}C_e\right\}\right)^{1/p} < +\infty.$$

A similar calculation for $\tilde{K}_{\pi,\ell}$ leads to the following:

$$\tilde{K}_p = \left(\int_{\mathsf{X}} |\tilde{K}_{\pi,\ell}(u)|^p \mu_{\mathsf{pr}}(\mathrm{d}u)\right)^{1/p} \leq (p/e)\left(C_e \exp\left\{c_I^{-1}C_e\right\}\right)^{1/p} < +\infty.$$

□

Thus, we can show implication T1.

Lemma 5.4.3: *Suppose Assumption 5.4.1 holds. Then, for any $\ell = 0, 1, \ldots \mathsf{L}$, there exists a positive constant $C_w \in \mathbb{R}_+$, independent of ℓ, such that:*

$$|\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell(u)] - \mathbb{E}_{\mu^y}[\mathsf{QoI}(u)]| \leq C_w s^{-\alpha_w \ell},$$

with $\alpha_w = \min\{\alpha_q, \alpha\}$ and α_q, α as in Assumption 5.4.1.

Proof. We follow an approach similar to that of [45].

$$\left|\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell(u)] - \mathbb{E}_{\mu^y}[\mathsf{QoI}(u)]\right| \leq \left|\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell(u)] - \mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}(u)]\right| + \left|\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}(u)] - \mathbb{E}_{\mu^y}[\mathsf{QoI}(u)]\right|.$$

For the first term:

$$\left|\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell(u)] - \mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}(u)]\right| \leq \mathbb{E}_{\mu_\ell^y}[|\mathsf{QoI}_\ell(u) - \mathsf{QoI}(u)|]$$
$$\leq \left(\int_{\mathsf{X}} C_q(u)\mu_\ell^y(\mathrm{d}u)\right) s^{-\alpha_q \ell} \leq \frac{s^{-\alpha_q\ell}}{Z_\ell}\int_{\mathsf{X}} C_q(u)\mu_{\mathsf{pr}}(\mathrm{d}u) \leq c_I^{-1}\tilde{C}_q s^{-\alpha_q\ell}. \tag{5.18}$$

For the second term:

$$\left|\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}(u)] - \mathbb{E}_{\mu^y}[\mathsf{QoI}(u)]\right| = \left|\int_{\mathsf{X}} \mathsf{QoI}(u)[\pi_\ell^y(u) - \pi^y(u)]\mu_{\mathsf{pr}}(\mathrm{d}u)\right|$$
$$\leq \int_{\mathsf{X}} |\mathsf{QoI}(u)|\,(\pi_\ell^y(u) + \pi^y(u))\tilde{K}_{\pi,\ell}(u)\mu_{\mathsf{pr}}(\mathrm{d}u)s^{-\alpha\ell}. \tag{5.19}$$

Working on the first term of the previous integral, we obtain the following from Hölder's inequality:

$$\begin{aligned}
&\left| \int_{\mathsf{X}} \mathsf{QoI}(u) \pi_\ell^y(u) \tilde{K}_{\pi,\ell}(u) \mu_{\mathsf{pr}}(\mathrm{d}u) \right| \\
&\leq \left(\int_{\mathsf{X}} |\mathsf{QoI}(u)|^m \mu_{\mathsf{pr}}(\mathrm{d}u) \right)^{1/m} \left(\int_{\mathsf{X}} (\pi_\ell^y(u))^m |\tilde{K}_{\pi,\ell}(u)|^{m'} \mu_{\mathsf{pr}}(\mathrm{d}u) \right)^{1/m'} \\
&\leq C_{\mathrm{m}} c_I^{-1} \tilde{K}_{m'},
\end{aligned}$$

where we have taken m as in Assumption 5.4.1, $m' = 1 - 1/m$ and $\tilde{K}_{m'}$ as in Lemma 5.4.2. A similar bound holds for the second term in (5.19), thus leading to

$$\left| \mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}(u)] - \mathbb{E}_{\mu^y}[\mathsf{QoI}(u)] \right| \leq 2 c_I^{-1} C_{\mathrm{m}} \tilde{K}_{m'} s^{-\alpha \ell}. \tag{5.20}$$

The desired result follows from (5.18) and (5.20), with $\alpha_w = \min\{\alpha_q, \alpha\}$, and a level independent constant $C_w = c_I^{-1}(2 C_{\mathrm{m}} \tilde{K}_{m'} + \tilde{C}_q)$. □

We now turn our attention to implication T2. We first prove several auxiliary results.
For any given level $\ell = 0, 1, \ldots, \mathsf{L}$, we say that the joint chains created by Algorithm 7 are *synchronized* at step n if $u_{\ell,\ell}^n = u_{\ell,\ell-1}^n$. Conversely, we say they are *unsynchronized* at step n if $u_{\ell,\ell}^n \neq u_{\ell,\ell-1}^n$. Notice that if the chains are synchronized at a state $u_{\ell,\ell}^n = u_{\ell,\ell-1}^n = u$, and the new proposed state at the $(n+1)^{\text{th}}$ iteration of the algorithm is $z \in \mathsf{X}$, they de-synchronize at the next step with probability $|\alpha_\ell(u,z) - \alpha_{\ell-1}(u,z)|$ (c.f. Figure 5.1). Intuitively, one would expect that such a probability approaches 0 as $\ell \to \infty$. We formalize this intuition below.

Lemma 5.4.4: *Suppose Assumptions 5.4.1.1 hold. Then, the following bound holds*

$$|\alpha_\ell(u,z) - \alpha_{\ell-1}(u,z)| \leq h_\ell(u,z) s^{-\alpha \ell}, \quad u, z \in \mathsf{X},$$

with

$$h_\ell(u,z) := \frac{Q_\ell(u)}{Q_\ell(z)} \frac{1}{\pi_\ell^y(u) \pi_{\ell-1}^y(u)} \left| \pi_\ell^y(z) C_{\pi,\ell}(u) + \pi_\ell^y(u) C_{\pi,\ell}(z) \right|$$

and $C_{\pi,\ell}(\cdot)$ as in Lemma 5.4.2.

Proof. From the definition of α_ℓ, and the fact that $\psi(x) := \min\{1, x\}$ is Lipschitz continuous with a constant of 1, it can be seen that

$$|\alpha_\ell(u,z) - \alpha_{\ell-1}(u,z)| \leq \left| \frac{Q_\ell(u)}{Q_\ell(z)} \frac{\pi^y_\ell(z)}{\pi^y_\ell(u)} - \frac{Q_\ell(u)}{Q_\ell(z)} \frac{\pi^y_{\ell-1}(z)}{\pi^y_{\ell-1}(u)} \right| = \frac{Q_\ell(u)}{Q_\ell(z)} \left| \frac{\pi^y_\ell(z)}{\pi^y_\ell(u)} - \frac{\pi^y_{\ell-1}(z)}{\pi^y_{\ell-1}(u)} \right|$$
$$= \frac{Q_\ell(u)}{Q_\ell(z)} \frac{1}{\pi^y_\ell(u)\pi^y_{\ell-1}(u)} \left| \pi^y_\ell(z)(-\pi^y_\ell(u) + \pi^y_{\ell-1}(u)) + \pi^y_\ell(u)(\pi^y_\ell(z) - \pi^y_{\ell-1}(z)) \right|$$
$$\leq \frac{Q_\ell(u)}{Q_\ell(z)} \frac{1}{\pi^y_\ell(u)\pi^y_{\ell-1}(u)} \left(\pi^y_\ell(z) C_{\pi,\ell}(u) + \pi^y_\ell(u) C_{\pi,\ell}(z) \right) s^{-\alpha\ell}.$$

□

Lemma 5.4.5: *Suppose Assumptions 5.3.1 and 5.4.1 hold. Furthermore, denote the diagonal set of* X^2 *as* $\Delta := \{(u,z) \in \mathsf{X}^2 \text{ s.t. } u = z\}$. *The transition probability to* Δ^c *for the coupled chain of Algorithm 7 is such that*

$$\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c) \leq R_\ell(u) s^{-\alpha\ell}, \quad \forall \boldsymbol{u}_\ell = (u,u) \in \Delta,$$

with

$$R_\ell(u) = \frac{Q_\ell(u)}{\pi^y_\ell(u)\pi^y_{\ell-1}(u)} \left(C_{\pi,\ell}(u) + \pi^y_\ell(u) K_1 \right),$$

and $C_{\pi,\ell}(\cdot)$ *and* K_1 *as in Lemma 5.4.2. Moreover, whenever* $\boldsymbol{u}_\ell \in \Delta^c$,

$$\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c) \leq 1 - c \int_{\mathsf{X}} \min\{\pi^y_\ell(u), \pi^y_{\ell-1}(u)\} \mu_{\mathsf{pr}}(\mathrm{d}u),$$

where c *is the same constant as in Assumption 5.3.1.1. Furthermore,* $\exists \delta > 0$ *independent of* ℓ *such that*

$$\inf_{\ell \in \mathbb{N}} \int_{\mathsf{X}} \min\{\pi^y_\ell(u), \pi^y_{\ell-1}(u)\} \mu_{\mathsf{pr}}(\mathrm{d}u) > \delta > 0. \tag{5.21}$$

Proof. We begin with the first inequality. For $\boldsymbol{u}_\ell \in \Delta$ and from the definition of $\boldsymbol{p}_\ell$ we obtain the following

$$\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c) = \int_{\mathsf{X}} (\alpha_{\ell-1}(u_{\ell,\ell-1}, z) - \alpha_\ell(u_{\ell,\ell}, z))^+ Q_\ell(z) \mathbb{1}_{\{(z,u_{\ell,\ell}) \in \Delta^c\}} \mu_{\mathsf{pr}}(\mathrm{d}z)$$
$$+ \int_{\mathsf{X}} (\alpha_\ell(u_{\ell,\ell}, z) - \alpha_{\ell-1}(u_{\ell,\ell-1}, z))^+ Q_\ell(z) \mathbb{1}_{\{(u_{\ell,\ell-1},z) \in \Delta^c\}} \mu_{\mathsf{pr}}(\mathrm{d}z),$$

where the first and last term in (5.3) are both zero. Writing $u_{\ell,\ell} = u_{\ell,\ell-1} = u$, it then follows from Lemma 5.4.4 that:

$$\begin{aligned}
\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c) &\leq \int_{\mathsf{X}} |\alpha_{\ell-1}(u,z) - \alpha_\ell(u,z)| Q_\ell(z) \mu_{\mathsf{pr}}(\mathrm{d}z) \\
&\leq \frac{Q_\ell(u) s^{-\alpha\ell}}{\pi^y_\ell(u)\pi^y_{\ell-1}(u)} \int_{\mathsf{X}} \left| \pi^y_\ell(z) C_{\pi,\ell}(u) + \pi^y_\ell(u) C_{\pi,\ell}(z) \right| \mu_{\mathsf{pr}}(\mathrm{d}z) \\
&\leq \frac{Q_\ell(u) s^{-\alpha\ell}}{\pi^y_\ell(u)\pi^y_{\ell-1}(u)} \left(C_{\pi,\ell}(u) + \pi^y_\ell(u) \int_{\mathsf{X}} C_{\pi,\ell}(z) \mu_{\mathsf{pr}}(\mathrm{d}z) \right) \\
&\leq \frac{Q_\ell(u) s^{-\alpha\ell}}{\pi^y_\ell(u)\pi^y_{\ell-1}(u)} \left(C_{\pi,\ell}(u) + 2c_I^{-1} K_1 \pi^y_\ell(u) \right)
\end{aligned}$$

Thus, $\forall \boldsymbol{u}_\ell \in \Delta$,

$$\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c) \leq R_\ell(u) s^{-\alpha\ell},$$

with

$$R_\ell(u) = \frac{Q_\ell(u)}{\pi^y_\ell(u)\pi^y_{\ell-1}(u)} \left(C_{\pi,\ell}(u) + 2\pi^y_\ell(u) c_I^{-1} K_1 \right).$$

Next, we focus on the second inequality which holds for $\boldsymbol{u}_\ell \in \Delta^c$. Thus, from the fact that $\max\{a,b\} - |a-b| = \min\{a,b\}\ \forall a, b \in \mathbb{R}$, using Assumption 5.3.1.1, we obtain

$$\begin{aligned}
\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c) &\leq \int_{\mathsf{X}} \left(1 - \min_{j=\ell-1,\ell} \{\alpha_j(u_{\ell,j}, u)\} \right) Q_\ell(u) \mu_{\mathsf{pr}}(\mathrm{d}u) \\
&\leq 1 - \int_{\mathsf{X}} \min_{j=\ell-1,\ell} \left[\min \left\{ 1, c \frac{\pi^y_j(u)}{Q_\ell(u)} \right\} \right] Q_\ell(u) \mu_{\mathsf{pr}}(\mathrm{d}u) \\
&= 1 - \int_{\mathsf{X}} \min_{j=\ell-1,\ell} \left[\min \left\{ Q_\ell(u), c\pi^y_j(u) \right\} \right] \mu_{\mathsf{pr}}(\mathrm{d}u) \\
&= 1 - \int_{\mathsf{X}} \min_{j=\ell-1,\ell} \left[\min \left\{ \frac{Q_\ell(u)}{\pi^y_j(u)}, c \right\} \pi^y_j(u) \right] \mu_{\mathsf{pr}}(\mathrm{d}u) \\
&\leq 1 - c \int_{\mathsf{X}} \min_{j=\ell-1,\ell} \left\{ \pi^y_j(u) \right\} \mu_{\mathsf{pr}}(\mathrm{d}u).
\end{aligned}$$

where c is the same constant as in Assumption 5.3.1.1 (notice that $c < 1$).

Finally, we demonstrate that the integral term in the previous equation is lower bounded by a strictly positive constant independent of the ℓ. First notice that

$$\lim_{\ell\to\infty}\int_{\mathsf{X}}\min_{j=\ell-1,\ell}\left\{\pi_j^y(u)\right\}\mu_{\mathsf{pr}}(\mathrm{d}u) = 1 - \lim_{\ell\to\infty}\frac{1}{2}\int_{\mathsf{X}}|\pi_\ell^y(u)-\pi_{\ell-1}^y(u)|\mu_{\mathsf{pr}}(\mathrm{d}u)$$
$$\geq \lim_{\ell\to\infty}(1-K_1 s^{-\alpha\ell}) = 1,$$

and, by definition,

$$\int_{\mathsf{X}}\min_{j=\ell-1,\ell}\left\{\pi_j^y(u)\right\}\mu_{\mathsf{pr}}(\mathrm{d}u) \leq 1, \quad \forall \ell\in\mathbb{N}$$

Thus, the sequence $\{\int_{\mathsf{X}}\min_{j=\ell-1,\ell}\left\{\pi_j^y(u)\right\}\mu_{\mathsf{pr}}(\mathrm{d}u)\}_{\ell\in\mathbb{N}}$ has 1 as an accumulation point, as $\ell\to\infty$, and, fixed any $\delta\in(0,1)$, there exists $\ell'\geq 0$ such that, for any $\ell\geq\ell'$, $\int_{\mathsf{X}}\min_{j=\ell-1,\ell}\left\{\pi_j^y(u)\right\}\mu_{\mathsf{pr}}(\mathrm{d}u)\} \geq \delta'$. Lastly, recall that by Assumption 5.3.1.2 π_ℓ^y and $\pi_{\ell-1}^y$ are continuous and strictly positive. Thus, for any compact set $A\subset\mathsf{X}$ with $\mu_{\mathsf{pr}}(A)>0$, and for any $\ell=\{0,1,\ldots,\ell'\}$, we have

$$\int_{\mathsf{X}}\min_{j=\ell-1,\ell}\left\{\pi_j^y(u)\right\}\mu_{\mathsf{pr}}(\mathrm{d}u) \geq \int_{A}\min_{j=\ell-1,\ell}\left\{\pi_j^y(u)\right\}\mu_{\mathsf{pr}}(\mathrm{d}u) =: \delta_\ell > 0.$$

Thus setting $\hat{\delta}=\min_{0\leq\ell\leq\ell'}\{\delta_\ell\}$, and $\delta=\min\{\hat{\delta},\delta'\}$ we obtain that, for any $\ell\geq 0$

$$\int_{\mathsf{X}}\min_{j=\ell-1,\ell}\left\{\pi_j^y(u)\right\}\mu_{\mathsf{pr}}(\mathrm{d}u) \geq \delta > 0.$$

□

Remark 5.4.3 (On the dependence of the TV distance between posteriors): *Notice that the term* $\int_{\mathsf{X}}\min\{\pi_\ell^y(u),\pi_{\ell-1}^y(u)\}\mu_{\mathsf{pr}}(\mathrm{d}u)$ *can be written as*

$$\int_{\mathsf{X}}\min\{\pi_\ell^y(u),\pi_{\ell-1}^y(u)\}\mu_{\mathsf{pr}}(\mathrm{d}u) = 1-\frac{1}{2}\int_{\mathsf{X}}\left|\pi_\ell^y(u)-\pi_{\ell-1}^y(u)\right|\mu_{\mathsf{pr}}(\mathrm{d}u) = 1-d_{TV}(\mu_\ell^y,\mu_{\ell-1}^y).$$

Furthermore, it is a consequence of Lemma 5.4.2 that $d_{TV}(\mu_\ell^y,\mu_{\ell-1}^y)\to 0$ *as* $\ell\to\infty$. *Thus, a bound on* δ *depends on the largest TV distance between two consecutive posteriors, which, intuitively, one would expect to occur at the coarser discretization levels.*

Lemma 5.4.6: *Suppose Assumptions 5.3.1 and 5.4.1 hold. Then, for all* $\ell=1,2,\ldots,\mathsf{L}$, *there exist a positive constant* $C_{r,\ell}$ *with* $C_{r,\ell}\to C_r^*>0$ *as* $\ell\to\infty$, *such that*

$$\mathbb{P}_{\nu_\ell}(u_{\ell,\ell}^n\neq u_{\ell,\ell-1}^n)\leq C_{r,\ell}s^{-\alpha\ell}, \quad \forall n\in\mathbb{N},$$

with c *as in Assumption 5.3.1.1 and* r *as in Assumption 5.3.1.3.*

Proof. For notational simplicity, for the remainder of this proof we will write $\mathsf{P}_n := \mathbb{P}_{\nu_\ell}(u^n_{\ell,\ell-1} \neq u^n_{\ell,\ell}), u^n_{\ell,\ell-1}, u^n_{\ell,\ell} \in \mathsf{X},\ n \in \mathbb{N}$. Let $Z_{\Delta,\ell} := \int_\Delta \nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) = (1 - \mathsf{P}_n)$. From Lemma 5.4.5 we obtain, for any $n \in \mathbb{N}$

$$\begin{aligned}
&\mathbb{P}_{\nu_\ell}(u^{n+1}_{\ell,\ell} \neq u^{n+1}_{\ell,\ell-1} | \boldsymbol{u}^n_\ell \in \Delta) = Z^{-1}_{\Delta,\ell} \int_\Delta \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&\leq \frac{s^{-\alpha\ell}}{Z_{\Delta,\ell}} \int_\Delta R_\ell(u)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \quad \text{(with } \boldsymbol{u}_\ell = (u,u) \text{ on } \Delta\text{)} \\
&\leq \frac{s^{-\alpha\ell}}{Z_{\Delta,\ell}} \underbrace{\int_\Delta \frac{Q_\ell(u)}{\pi^y_{\ell-1}(u)}(K_{\pi,\ell}(u) + 2c_I^{-1}K_1)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell)}_{\mathrm{I}} + \frac{s^{-\alpha\ell}}{Z_{\Delta,\ell}} \underbrace{\int_\Delta \frac{Q_\ell(u)}{\pi^y_\ell(u)} K_{\pi,\ell}(u)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell)}_{\mathrm{II}}
\end{aligned}$$

We begin with integral I:

$$\begin{aligned}
\mathrm{I} &= \int_{\mathsf{X}^2} \frac{Q_\ell(u_{\ell,\ell-1})}{\pi^y_{\ell-1}(u_{\ell,\ell-1})}(K_{\pi,\ell}(u_{\ell,\ell-1}) + 2c_I^{-1}K_1)\mathbb{1}_{\{(u_{\ell,\ell-1},u_{\ell,\ell})\in\Delta\}}\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&\leq \int_{\mathsf{X}} \frac{Q_\ell(u_{\ell,\ell-1})}{\pi^y_{\ell-1}(u_{\ell,\ell-1})}(K_{\pi,\ell}(u_{\ell,\ell-1}) + 2c_I^{-1}K_1) \int_{\mathsf{X}} \nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&= \int_{\mathsf{X}} Q_\ell(u_{\ell,\ell-1})(K_{\pi,\ell}(u_{\ell,\ell-1}) + 2c_I^{-1}K_1)\mu_{\mathsf{pr}}(\mathrm{d}u_{\ell,\ell-1}) \\
&\leq \left(\int_{\mathsf{X}} |Q_\ell(u_{\ell,\ell-1})|^r \mu_{\mathsf{pr}}(\mathrm{d}u_{\ell,\ell-1})\right)^{1/r} \left(2c_I^{-1}K_1 + \left(\int_{\mathsf{X}} |K_{\pi,\ell}(u_{\ell,\ell-1})|^{r'}\mu_{\mathsf{pr}}(\mathrm{d}u_{\ell,\ell-1})\right)^{1/r'}\right) \\
&= C_r(2c_I^{-1} + 1)K_{r'}.
\end{aligned}$$

Similarly, for II, we get:

$$\begin{aligned}
\mathrm{II} &= \int_{\mathsf{X}^2} \frac{Q_\ell(u_{\ell,\ell})}{\pi^y_\ell(u_{\ell,\ell})} K_{\pi,\ell}(u_{\ell,\ell})\mathbb{1}_{\{(u_{\ell,\ell-1},u_{\ell,\ell})\in\Delta\}}\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&\leq \int_{\mathsf{X}} Q_\ell(u_{\ell,\ell})K_{\pi,\ell}(u_{\ell,\ell})\mu_{\mathsf{pr}}(\mathrm{d}u_{\ell,\ell}) \leq C_r K_{r'}.
\end{aligned}$$

Setting $\hat{C} = 2C_rK_{r'}(c_I^{-1} + 1)$, one then has

$$\mathbb{P}_{\nu_\ell}(u^{n+1}_{\ell,\ell} \neq u^{n+1}_{\ell,\ell-1} | u^n_{\ell,\ell} = u^n_{\ell,\ell-1}) \leq \hat{C}Z^{-1}_{\Delta,\ell}s^{-\alpha\ell} := Z^{-1}_{\Delta,\ell}s_\ell,$$

where we have set $s_\ell = \hat{C}s^{-\alpha\ell}$. Similarly, letting $Z_{\Delta^c,\ell} := \int_{\Delta^c} \nu_\ell(\mathrm{d}\boldsymbol{u}_\ell)$, one has

$$\begin{aligned}
&\mathbb{P}_{\nu_\ell}(u^{n+1}_{\ell,\ell} \neq u^{n+1}_{\ell,\ell-1} | \boldsymbol{u}^n_\ell \in \Delta^c) = Z^{-1}_{\Delta^c,\ell} \int_{\Delta^c} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&\leq 1 - c\int_{\mathsf{X}} \min\{\pi^y_\ell(u), \pi^y_{\ell-1}(u)\}\mu_{\mathsf{pr}}(\mathrm{d}u) =: \tilde{c}_\ell \quad \text{(from Lemma 5.4.5).}
\end{aligned}$$

We write the de-synchronization probability at the $(n+1)^{\text{th}}$ step as follows:

$$\begin{aligned}\mathsf{P}_{n+1} &= \mathbb{P}_{\nu_\ell}(u_{\ell,\ell}^{n+1} \neq u_{\ell,\ell-1}^{n+1}) = \mathbb{P}_{\nu_\ell}(u_{\ell,\ell}^{n+1} \neq u_{\ell,\ell-1}^{n+1} | u_{\ell,\ell}^{n} = u_{\ell,\ell-1}^{n}) \mathbb{P}_{\nu_\ell}(u_{\ell,\ell}^{n} = u_{\ell,\ell-1}^{n}) \\ &\quad + \mathbb{P}_{\nu_\ell}(u_{\ell,\ell}^{n+1} \neq u_{\ell,\ell-1}^{n+1} | u_{\ell,\ell}^{n} \neq u_{\ell,\ell-1}^{n}) \mathbb{P}_{\nu_\ell}(u_{\ell,\ell}^{n} \neq u_{\ell,\ell-1}^{n}) \\ &\leq Z_{\Delta,\ell}^{-1} s_\ell \underbrace{(1-\mathsf{P}_n)}_{=Z_{\Delta,\ell}} + \tilde{c}_\ell \mathsf{P}_n \\ &\leq s_\ell + \tilde{c}_\ell \mathsf{P}_n. \end{aligned} \tag{5.22}$$

However, by stationarity, $\mathsf{P}_{n+1} = \mathsf{P}_n =: \mathsf{P}$. Thus, from Equation (5.22),

$$\mathbb{P}_{\nu_\ell}(u_{\ell,\ell-1}^{n} \neq u_{\ell,\ell}^{n}) = \mathsf{P} \leq \frac{\hat{C} s^{-\alpha\ell}}{1-\tilde{c}_\ell} = \frac{\hat{C}}{c \int_{\mathsf{X}} \min\{\pi_\ell^y(u), \pi_{\ell-1}^y(u)\} \mu_{\mathsf{pr}}(\mathrm{d}u)} s^{-\alpha\ell}.$$

By (5.21) the integral term in the denominator is lower bounded by a constant δ independent of the level. Furthermore, this integral converges to 1 as $\ell \to \infty$. □

We are now ready to prove implication T2.

Lemma 5.4.7: *Suppose Assumptions 5.4.1 and 5.3.1.2 hold. Then, for any $\ell \geq 1$, there exists a positive constant C_v such that*

$$\mathbb{V}_{\nu_\ell}[Y_\ell] \leq C_v s^{-\beta\ell},$$

where $\beta = \min\{2\alpha_q, \alpha(1-2/m)\}$, and α, α_q, m as in Assumption 5.4.1.

Proof. We follow an argument similar to that of [45, Lemma 4.8]. From Young's inequality we have

$$\begin{aligned}&\mathbb{V}_{\nu_\ell}[Y_\ell] \leq \mathbb{E}_{\nu_\ell}\left[(\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1}))^2\right] \\ &\leq 2\mathbb{E}_{\nu_\ell}\left[(\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_\ell(u_{\ell,\ell-1}))^2\right] + 2\mathbb{E}_{\nu_\ell}\left[(\mathsf{QoI}_\ell(u_{\ell,\ell-1}) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1}))^2\right].\end{aligned}$$

In the case in which $\mathsf{QoI}_\ell(\cdot)$ and $\mathsf{QoI}_{\ell-1}(\cdot)$ are the same (which could happen when the quantity of interest, seen as a functional, is mesh-independent), the second term vanishes. Otherwise, we have, using Assumption 5.4.1.2, that

$$\mathbb{E}_{\nu_\ell}\left[(\mathsf{QoI}_\ell(u_{\ell,\ell-1}) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1}))^2\right] \leq 2\tilde{C}_q^2(1+s^{2\alpha_q})s^{-2\alpha_q\ell}.$$

The first term is only nonzero when $u_{\ell,\ell} \neq u_{\ell,\ell-1}$. Thus, it can be rewritten as

$$2\mathbb{E}_{\nu_\ell}\left[(\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_\ell(u_{\ell,\ell-1}))^2\right] = 2\mathbb{E}_{\nu_\ell}\left[(\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_\ell(u_{\ell,\ell-1}))^2 \mathbb{1}_{\{u_{\ell,\ell} \neq u_{\ell,\ell-1}\}}\right].$$

By applying Hölder's inequality, we can write the above expression as follows:

$$
\begin{aligned}
&2\mathbb{E}_{\nu_\ell}\left[(\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_\ell(u_{\ell,\ell-1}))^2 \mathbb{1}_{\{u_{\ell,\ell}\neq u_{\ell,\ell-1}\}}\right] \\
&\leq 2\mathbb{E}_{\nu_\ell}\left[|\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_\ell(u_{\ell,\ell-1})|^{2m/2}\right]^{2/m} \mathbb{E}_{\nu_\ell}[\mathbb{1}^{m'}_{\{u_{\ell,\ell}\neq u_{\ell,\ell-1}\}}]^{1/m'} \text{ (with } m' = m/(m-2)) \\
&= 2\mathbb{E}_{\nu_\ell}\left[|\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_\ell(u_{\ell,\ell-1})|^{m}\right]^{2/m} \mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1})^{1/m'}. \qquad (5.23)
\end{aligned}
$$

From Assumption 5.4.1.2c, it follows that we can bound the first term in Equation (5.23) by

$$
\begin{aligned}
&\mathbb{E}_{\nu_\ell}\left[|\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_\ell(u_{\ell,\ell-1})|^{m}\right]^{2/m} \\
&\leq \left(\mathbb{E}_{\mu^y_\ell}[\mathsf{QoI}_\ell(u_{\ell,\ell})^m]^{\frac{1}{m}} + \mathbb{E}_{\mu^y_{\ell-1}}[\mathsf{QoI}_\ell(u_{\ell,\ell-1})^m]^{\frac{1}{m}}\right)^2 \\
&\leq 4c_I^{-2/m} C_{\mathrm{m}}^2.
\end{aligned}
$$

Moreover, from Lemma 5.4.6, we have that $\mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1}) \leq C_{r,\ell} s^{-\alpha\ell}$. Thus,

$$
\mathbb{V}_{\nu_\ell}[Y_\ell] \leq C_v s^{-\beta\ell},
$$

where $C_v = 8c_I^{-2/m} C_{\mathrm{m}}^2 \max_{\ell\in\mathbb{N}}\{C_{r,\ell}\} + 4\tilde{C}_q^2(1 + s^{2\alpha_q})$. □

5.5 Implementation

We discuss how to choose the optimal number of samples N_ℓ. For $\ell = 0, \ldots, \mathsf{L}$, we denote the total cost of producing one coupled sample of $(\mathsf{QoI}_{\ell-1}, \mathsf{QoI}_\ell)$ at level ℓ using Algorithm 8 by $\mathcal{C}_\ell$. The total cost of the multi-level MCMC estimator is calculated as follows:

$$
\mathcal{C}\left(\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}}\right) = \sum_{\ell=0}^{\mathsf{L}} \mathcal{C}_\ell N_\ell. \qquad (5.24)
$$

To bound the statistical contribution of the total error bound, from (5.10) and (5.12),we have the following constraint:

$$
2(\mathsf{L}+1)\sum_{\ell=0}^{\mathsf{L}} C_{\mathrm{mse}} \frac{\mathbb{V}_{\nu_\ell}[Y_\ell]}{N_\ell} \leq \frac{\mathsf{tol}^2}{2},
$$

where tol is a user-prescribed tolerance. However, it is generally not a simple task to compute or estimate the constant C_{mse}. We ignore it hereafter, and aim at bounding the following quantity:

$$
2(\mathsf{L}+1)\sum_{\ell=0}^{\mathsf{L}} \frac{\mathbb{V}_{\nu_\ell}[\hat{Y}_\ell]}{N_\ell} \leq \frac{\mathsf{tol}^2}{2}. \qquad (5.25)
$$

To that end we will use the so-called *batched means estimator* of $\mathbb{V}_{\nu_\ell}[\hat{Y}_\ell]$ denoted by $\hat{\sigma}^2_\ell$ (see [54] for further details). In this case, treating N_ℓ as a real number and minimizing (5.24) subject to (5.25), gives the optimal samples sizes

$$N_\ell = \left\lceil 2\mathsf{tol}^{-2}\sqrt{\hat{\sigma}^2_\ell/\mathcal{C}_\ell}\left(\sum_{j=0}^{\mathsf{L}}\sqrt{\hat{\sigma}^2_j\mathcal{C}_j}\right)\right\rceil, \tag{5.26}$$

where $\lceil\cdot\rceil$ is the ceiling function. Lastly, we must also ensure that the second contribution to the total error (i.e., the discretization bias at level L), is such that

$$\left|\mathbb{E}_{\mu^y_\mathsf{L}}[\mathsf{QoI}_\mathsf{L}] - \mathbb{E}_{\mu^y}[\mathsf{QoI}]\right| \leq \frac{\mathsf{tol}}{\sqrt{2}}.$$

From T1 it follows

$$\begin{aligned}\left|\mathbb{E}_{\mu^y_\mathsf{L}}[\mathsf{QoI}_\mathsf{L}] - \mathbb{E}_{\mu^y}[\mathsf{QoI}]\right| &= \left|\sum_{j=\mathsf{L}+1}^{\infty}\mathbb{E}_{\mu^y_j}[\mathsf{QoI}_j] - \mathbb{E}_{\mu^y_{j-1}}[\mathsf{QoI}_{j-1}]\right| \\ &\approx \frac{\left|[\widehat{\mathsf{QoI}}_\mathsf{L} - \widehat{\mathsf{QoI}}_{\mathsf{L}-1}]\right|}{1-s^{-\alpha_w}}.\end{aligned}$$

Thus, to achieve a total (estimated) MSE of the ML-MCMC estimator less than tol^2, we must check that

$$2(\mathsf{L}+1)\left(\sum_{\ell=0}^{\mathsf{L}}\frac{\hat{\sigma}^2_\ell}{N_\ell}\right) + 2\left(\frac{\left|[\widehat{\mathsf{QoI}}_\mathsf{L} - \widehat{\mathsf{QoI}}_{\mathsf{L}-1}]\right|}{1-s^{-\alpha_w}}\right)^2 \leq \mathsf{tol}^2.$$

In practice, the set of parameters $\mathcal{P} := \{C_w, \alpha_w, \{\hat{\sigma}^2_\ell\}_{\ell=0}^{\mathsf{L}}, C_\sigma, \beta, C_\gamma, \gamma\}$ must be estimated with a preliminary run over L_0 levels, using $\tilde{N}_\ell$, $\ell = 0, 1, \ldots, \mathsf{L}_0$ samples per level. However, the main disadvantage of this procedure is that this screening phase can be quite inefficient for computationally expensive problems. In particular, if L_0 is chosen too large, then the screening phase might be more expensive than the overall ML-MCMC simulation on the optimal hierarchy $\{0, 1, \ldots, \mathsf{L}\}$. On the other hand, if L_0 (or $\tilde{N}_\ell$) is chosen too small, the extrapolation (or estimation) of the values of $\mathcal{P}$ might be quite unreliable, particularly at higher levels. In the MLMC literature, one way of overcoming these issues is with the so-called continuation Multi-level Monte Carlo method [31]. We will present a continuation-type ML-MCMC (C-ML-MCMC) algorithm in the following subsection, based on [31, 132].

5.5.1 A continuation-type ML-MCMC

The key idea behind this method is to iteratively implement an ML-MCMC algorithm with a sequence of decreasing tolerances while, at the same time, progressively improving the estimation of the problem-dependent parameters $\mathcal{P}$. As presented, these parameters directly control the number of levels and sample sizes. Following [31], we introduce the family of tolerances $\mathsf{tol}_i,\ i = 0, 1, \ldots,$ given by

$$\mathsf{tol}_i = \begin{cases} r_1^{i_E - i} r_2^{-1} \mathsf{tol} & i < i_E, \\ r_2^{i_E - i} r_2^{-1} \mathsf{tol} & i \geq i_E, \end{cases}$$

where $r_1 \geq r_2 > 1$, so that $\mathsf{tol}_{i_E - 1} \geq \mathsf{tol} > \mathsf{tol}_{i_E}$, with

$$i_E := \left\lfloor \frac{-\log(\mathsf{tol}) + \log(r_2) + \log(\mathsf{tol}_0)}{\log(r_1)} \right\rfloor. \tag{5.27}$$

The idea is to iteratively run the ML-MCMC algorithm for each of the tolerances $\mathsf{tol}_i, i = 0, 1, \ldots$ until the algorithm achieves convergence, based on the criteria defined in the previous subsection. Iterations for which $i < i_E$, are used to obtain increasingly more accurate estimates of $\mathcal{P}$. Notice that when $i = i_E$, the problem is solved with a slightly smaller tolerance $r_2^{-1}\mathsf{tol}$ for some carefully chosen r_2. Solving at this slightly smaller tolerance is performed to prevent any extra unnecessary iterations due to the statistical nature of the estimated quantities. Furthermore, if the algorithm has not converged at the i_E^{th} iteration, it continues running for even smaller tolerances $\mathsf{tol}_i, i > i_E$, to account for cases where the estimates of $\mathcal{P}$ are unstable. Thus, at the i^{th} iteration of the C-ML-MCMC algorithm, we run Algorithm 8 with an iteration-dependent number of levels L_i, where L_i is obtained by solving the following discrete constrained optimization problem:

$$\begin{cases} \underset{\mathsf{L}_{i-1} \leq \mathsf{L} \leq \mathsf{L}_{\text{max}}}{\arg\min} & \left\{ 2\mathsf{tol}_i^{-2} 2(\mathsf{L}+1) \left(\sum_{j=0}^{\mathsf{L}} \sqrt{C_\beta s^{-\beta_j} \mathcal{C}_j} \right)^2 \right\}, \\ \text{s.t.} & C_w s^{-\alpha_w \mathsf{L}} \leq \frac{\mathsf{tol}_i}{\sqrt{2}}, \end{cases} \tag{5.28}$$

where, $\mathsf{L}_{-1} = \mathsf{L}_0$ is the given minimum number of levels, L_{max}, is chosen as the maximum number of levels given a computational budget (which could be dictated, for example, by the minimum mesh size imposed by memory or computational restrictions). Notice that (5.28) is easily solved by exhaustive search.

We now have everything needed to implement the C-ML-MCMC algorithm, which we present in the listing 9.

Algorithm 9 Continuation ML-MCMC

procedure C-ML-MCMC($\{\pi_\ell^y\}_{\ell=0}^{\mathsf{L}}, Q, \tilde{N}, \mathsf{L}_0, \mathsf{L}_{\text{max}}, \{\nu_\ell^0\}_{\ell=0}^{\mathsf{L}}, \mathsf{tol}_0, \mathsf{tol}, r_1, r_2$)
  # Preliminary run
  Compute i_E according to Equation (5.27). Set $N_\ell = \tilde{N}, \ell = 0, 1, \ldots, \mathsf{L}_0$,
  $\{\{u_{\ell,\ell}^n\}_{n=0}^{\tilde{N}}, \{u_{\ell,\ell-1}^n\}_{n=0}^{\tilde{N}_\ell}\}_{\ell=0}^{\mathsf{L}_0}$=ML-MCMC($\{\pi_\ell^y\}_{\ell=0}^{\mathsf{L}_0}, Q, \{N_\ell\}_{\ell=0}^{\mathsf{L}_0}, \{\nu_\ell^0\}_{\ell=0}^{\mathsf{L}_0}$).
  Compute estimates for the parameters $\mathcal{P}$ using least squares fit
  set $i = 1$ and $\mathsf{te} = \infty$.
  # Starts continuation algorithm
  while $i < i_E$ **or** $\mathsf{te} > \mathsf{tol}$ **do**
    Update tolerance $\mathsf{tol}_i = \mathsf{tol}_{i-1}/r_k$, where $k = 1$ if $i < i_E$ and $k = 2$ otherwise.
    Compute $\mathsf{L}_i = \mathsf{L}_i(\mathsf{L}_{i-1}, \mathsf{L}_{\text{max}}, \mathsf{tol}_i, \mathcal{P})$ using (5.28)
    Compute $N_\ell = N_\ell(\mathsf{L}_i, \mathsf{tol}_i, \mathcal{P})$ for $\ell = 0, 1, \ldots, \mathsf{L}_i$, using (5.26), for unexplored levels, extrapolate $\mathbb{V}_{\nu_\ell}[Y_\ell]$ from previous points
    # Q can be constructed using samples from previous iterations
    $\{\{u_{\ell,\ell}^n\}_{n=0}^{N_\ell}, \{u_{\ell,\ell-1}^n\}_{n=0}^{N_\ell}\}_{\ell=0}^{\mathsf{L}_i}$=ML-MCMC($\{\pi_\ell^y\}_{\ell=0}^{\mathsf{L}_i}, Q, \{N_\ell\}_{\ell=0}^{\mathsf{L}_i}, \{\nu_\ell^0\}_{\ell=0}^{\mathsf{L}_i}$))
    Update estimates for $\mathcal{P}$ using least squares fit
    Update total error $\mathsf{te} = 2(\mathsf{L}+1)\left(\sum_{\ell=0}^{\mathsf{L}_i} \hat{\sigma}_\ell^2/N_\ell\right) + 2\left(C_w s^{-\alpha_w \mathsf{L}_i}\right)^2$
    $i = i + 1$
  end while
  Return $\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}$ computed with (5.2).
end procedure

5.6 Numerical experiments

We first present two "sanity check" experiments to verify the theory presented in previous sections numerically.
In the following two experiments we compare our proposed ML-MCMC algorithm to that in [45], which, by construction, does not satisfy our Assumption 5.3.1.1. The aim of these experiments is to verify the theoretical results of the previous sections, as well as to provide a setting for which our methods might be better suited than the sub-sampling approach of [45]. For ease of exposition, we consider as a quantity of interest $\mathsf{QoI}(u) = u, \quad u \sim \mu^y$, and we assume that the cost of evaluating the posterior density at each level grows as $2^{\gamma\ell}$, with $\gamma = 1$. For both experiments, we implement the sub-sampling ML-MCMC algorithm of [45] with a level-dependent sub-sampling rate $t_\ell := \min\left\{1 + 2\sum_{k=0}^{N_\ell} \hat{\varrho}_k, 5\right\}$, where $\hat{\varrho}_k$ is the so-called *lag-k auto-correlation time* and $1 + 2\sum_{k=0}^{N_\ell} \hat{\varrho}_k$ is the so-called *integrated auto-correlation time* [21].

5.6.1 Nested Gaussians

We begin with a scenario for which both ML-MCMC methods can be applied. In this case we aim at sampling from the family of posteriors $\mu_\ell^y = \mathcal{N}\left(1, 1 + 2^{-\ell}\right), \ell = 0, 1, 2, \ldots$, which approximate $\mu^y = \mathcal{N}(1, 1)$ as $\ell \to \infty$. For the ML-MCMC method proposed in the current work, we will use a fixed proposal across all levels given by $Q_\ell = Q = \mathcal{N}(1, 3)$. Such proposal is chosen to guarantee that Assumption 5.3.1.1 is fulfilled. The family of posteriors and the proposal Q used in our ML-MCMC algorithm are depicted in Figure 5.2. For both algorithms, the proposal distribution at level $\ell = 0$ is a random walk Metropolis proposal $Q_0(u_0^n, \cdot) = \mathcal{N}(u_0^n, 1)$. This proposal is chosen to guarantee an acceptance rate of about 40%, the value deemed close to optimum for MCMC in one dimension [21].

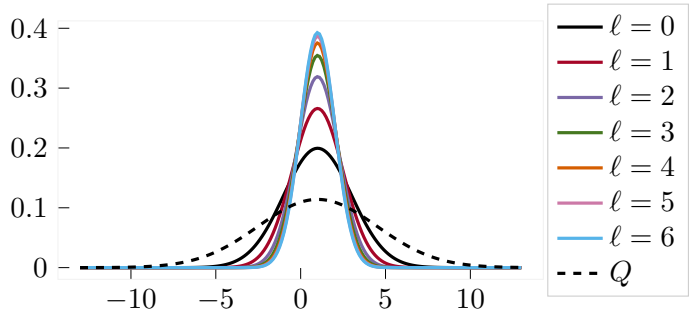

Figure 5.2: Family of posteriors μ_ℓ^y and fixed proposal distribution Q for the nested Gaussians example.

As a *sanity check*, we begin by verifying that both algorithms target the right marginal distribution at different levels. This can be seen in Figure 5.3, where the histograms of samples obtained with a simple ML-MCMC algorithm with proposal Q and prescribed number of levels $\mathsf{L} = 7$ and number of samples $N_\ell = 50000$ for $\ell = 0, 1, \ldots, \mathsf{L}$ (top row) and the algorithm of [45] (bottom

row) are shown for levels $\ell = 0, 3, 6$. The true posterior at level ℓ is shown in red. As it can be seen, both methods are able to sample from the right marginal distribution for the family of posteriors considered here-in.

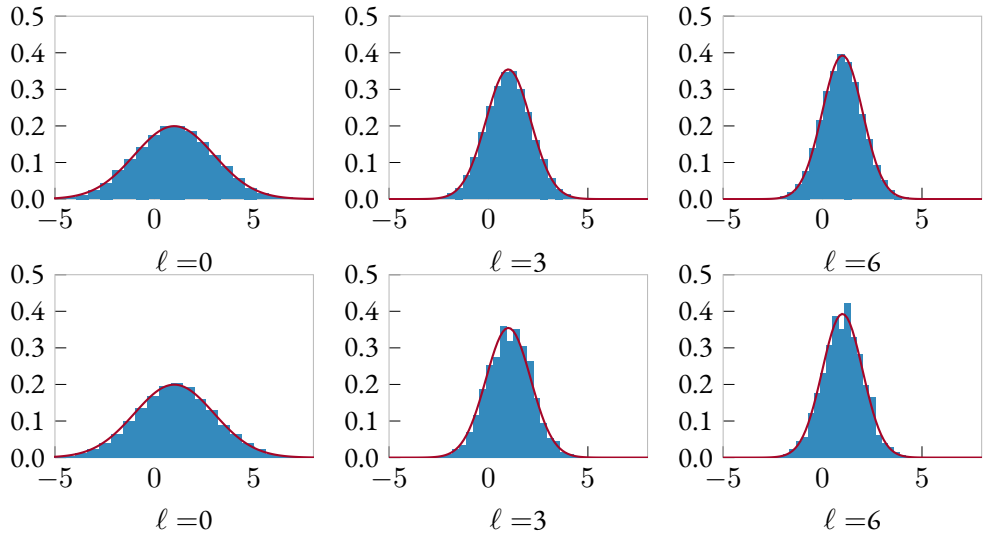

Figure 5.3: True posterior μ_ℓ^y for different levels $\ell = 0, 3, 6$ and histogram of the samples of $u_\ell \sim \mu_\ell^y$ obtained with the ML-MCMC algorithm described herein with $Q_\ell = \mathcal{N}(1, 3)$ (Top row)and the sub-sampling ML-MCMC algorithm (Bottom row). Both methods are able to obtain samples from the right posterior distribution.

We now aim at verifying the rates presented in Theorem 5.4.1. To that end, we run the ML-MCMC algorithm 100 independent times. For each independent run, we obtained 50,000 samples on each level and investigate the behavior of $|\mathbb{E}_{\nu_\ell}[Y_\ell]|$ (Figure 5.4 (left)) and $\mathbb{V}_{\nu_\ell}[Y_\ell]$ (Figure 5.4 (right)) with respect to the level ℓ. As it can be seen from Figure 5.4, both $|\mathbb{E}_{\nu_\ell}[Y_\ell]|$ and $\mathbb{V}_{\nu_\ell}[Y_\ell]$ decay with respect to ℓ with nearly the same estimated rate ≈ -1.34 for the ML-MCMC algorithm discussed in this current work, close to the predicted one in Theorem 5.4.1. It can be seen, however, from Figure 5.4 (right), that the variance decay of the sub-sampling algorithm is slightly better than the one obtained by the method presented herein. This, in turn, results in a smaller overall sample size at each level for a given particular error tolerance, as it can be seen in Figure 5.5 (left). We believe that this difference in rate is due (i) to the slightly higher synchronization rate of the sub-sampling ML-MCMC algorithm (Figure 5.5 (right)) and (ii) to the fact that the convergence rate of the marginal chain in the sub-sampling algorithm also increases with level, which is not necessarily the case for our method. These results suggest that, for this particular case, it is more cost-efficient to use the sub-sampling ML-MCMC algorithm.

We plot sample size vs level (Figure 5.5 (left)) and synchronization rate vs. level (Figure 5.5 (right)). Both figures were obtained from 100 independent runs: solid lines indicate the average value and dashed lines indicate 95% confidence intervals. The computation of N_ℓ for each level $\ell = 0, 1, \ldots, \mathsf{L}$ was done by estimating $\hat{\sigma}_\ell$ with 50,000 samples per level and a tolerance $\mathsf{tol} = 0.07$.

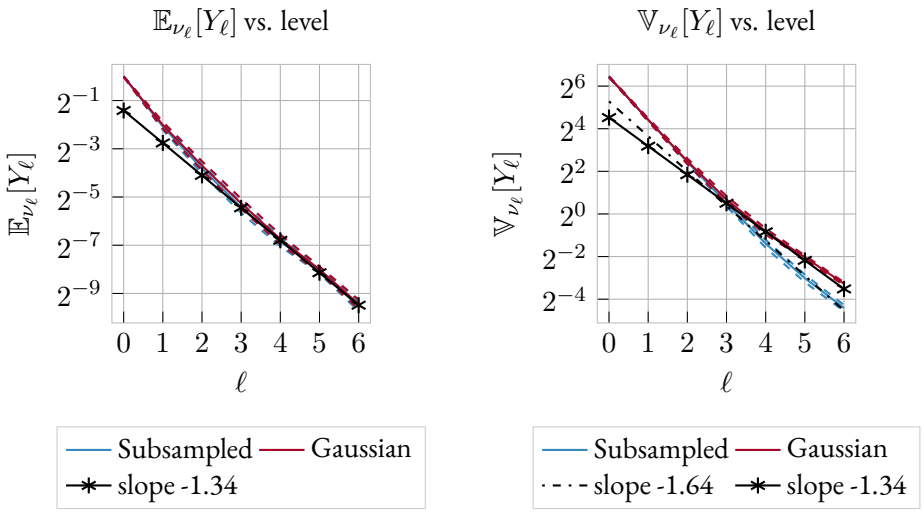

Figure 5.4: (Left) $|\mathbb{E}_{\nu_\ell}[Y_\ell]|$ Vs. level. (Right) $\mathbb{V}_{\nu_\ell}[Y_\ell]$ Vs. level. In both figures, the rates were estimated over 100 independent runs, with 50,000 samples per level, on each run. Solid lines indicate the average value, dashed lines indicate 95% confidence intervals.

It can be seen from Figure 5.5 (left) that the sub-sampling algorithm requires a smaller number of samples per level. From Figure 5.5 (right) we can see that both algorithms tend to a synchronization rate of 1, as expected. It can be seen that the sub-sampling algorithm provides a slightly higher synchronization rate for the problem at hand.

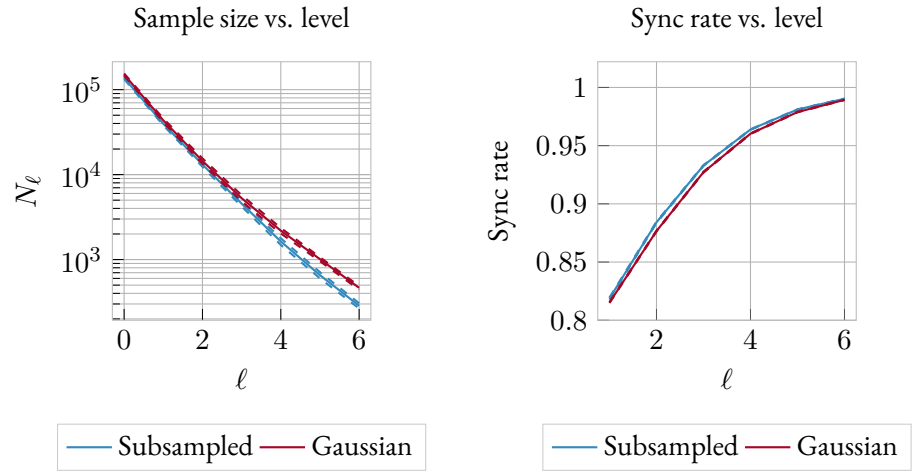

Figure 5.5: (Left) Number of samples, Vs. level for both algorithms. (Right) Synchronization rate vs level for both algorithms.

Lastly, we perform some robustness experiments for our C-ML-MCMC algorithm. To that end, we run Algorithm 9 using the same level independent proposals $Q_\ell = Q = \mathcal{N}(1, 3)$ for three different prescribed tolerances $\mathsf{tol} = \{0.025, 0.05, 0.1\}$. The algorithm is run for a total of 100

independent times. At each run k, we compute the total squared error of the multi-level estimator obtained from the k^{th} run of the C-ML-MCMC algorithm given by

$$\mathsf{er}_k^2 := \left(\widehat{\mathsf{QoI}}^{(k)}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}} - \mu^y(\mathsf{QoI}) \right)^2 \tag{5.29}$$

and plot it in Figure 5.6. As we can see, we obtain estimators whose mean square error is less than the prescribed tolerance, as desired. This result evidences the robustness of Algorithm 9 when computing quantities of interest for a given tolerance.

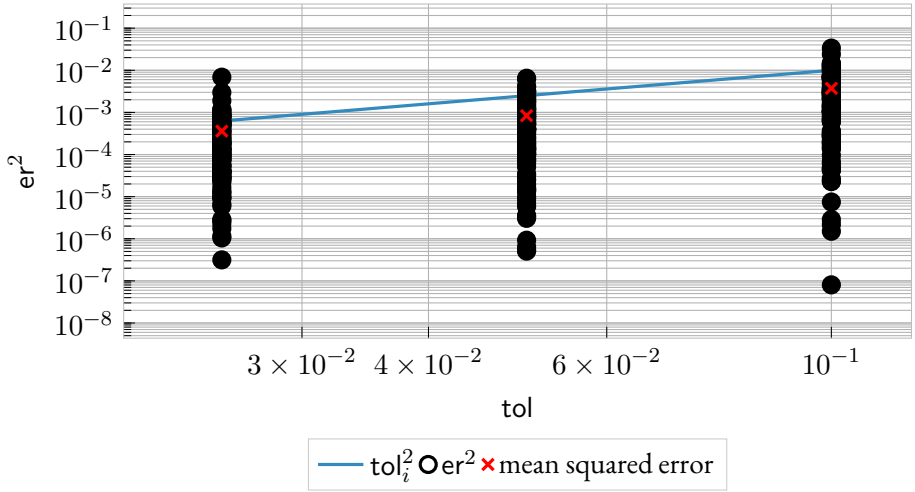

Figure 5.6: Total squared error er^2 vs tol for the nested Gaussians example. Here, we used 100 independent runs of the full C-ML-MCMC algorithm for three different tolerances; $\mathsf{tol} = 0.025, 0.05, 0.1$ (black circles). The red cross denotes the estimated MSE over the 100 runs.

5.6.2 Shifting Gaussians

We now move to a slightly more challenging problem, which is better suited for our proposed method. In this case, we aim at sampling from the family of posteriors $\mu_\ell^y = \mathcal{N}\left(2^{-\ell+2}, 1\right)$, $\ell = 0, 1, 2, \ldots, \mathsf{L}$, which approximate $\mu^y = \mathcal{N}(0, 1)$ as $\ell \to \infty$. Once again, for the ML-MCMC method proposed in the current work, we will use a fixed proposal across all levels given by $Q_\ell = Q = \mathcal{N}(2, 3)$. Such a proposal is chosen to guarantee that Assumption 5.3.1.1 is fulfilled. The posterior and proposal densities are shown in Figure 5.7. Just as in experiment 5.6.1 the proposal distribution at level $\ell = 0$ for both algorithms is a random walk Metropolis proposal $Q_0(u_0^n, \cdot) = \mathcal{N}(u_0^n, 1)$. This proposal is chosen to guarantee an acceptance rate of about 40%.

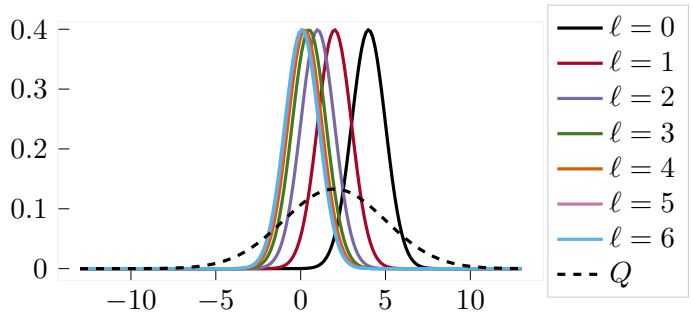

Figure 5.7: Illustration of the posterior densities π_ℓ and the proposal Q for the moving Gaussians example.

Once again, we begin by investigating the correctness of the corresponding marginals. Therefore, we run both algorithms for $\mathsf{L} = 6$, obtaining $50,000$ samples per level and plot the resulting histograms of μ_ℓ^y for levels $\ell = 0, 3, 6$. Such results are presented in Figure 5.8. As it can be seen, the presented ML-MCMC (Figure 5.8, top row) is able to sample from the correct marginals. In contrast, the sub-sampling ML-MCMC algorithm is not able to produce samples from the correct distributions, at least for the number of samples considered, as it can be seen in Figure 5.8 (Bottom row). We believe that this is because Assumption 5.3.1.1 not being satisfied due to the very small *overlap* between the posterior at level 0 and the posteriors at higher levels. Sampling from the wrong marginal distribution results in biased estimators when using the sub-sampling method [45].

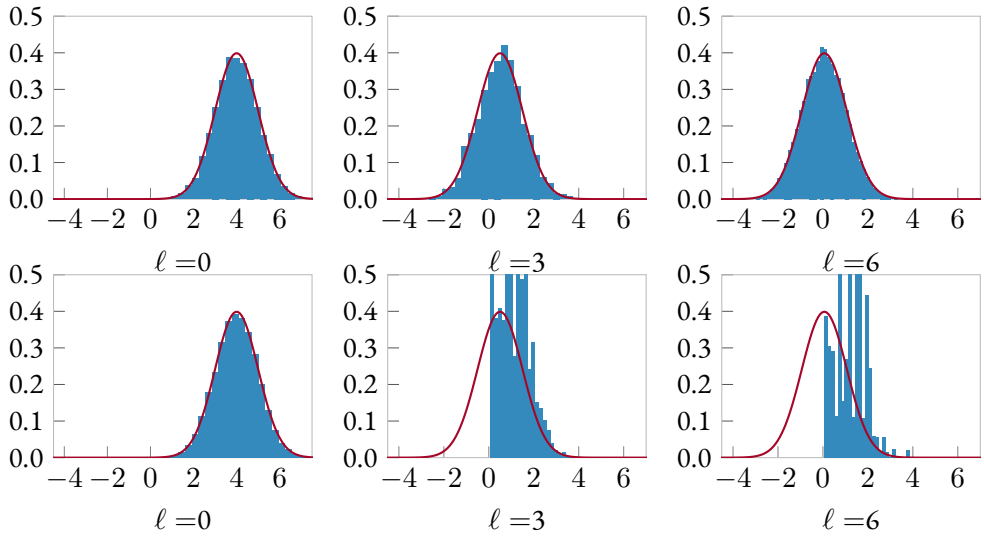

Figure 5.8: Sample histograms for one ML-MCMC run at levels $\ell = 0, 3, 6$ (Top row): Fixed Gaussian proposal . (Bottom row): Sub-sampling approach. As it can be seen, the sub-sampling approach is not able to properly sample from the posterior at higher levels.

Next, we verify the converge rates stated in Theorem 5.4.1. For this particular setting we have $|\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell] - \mathbb{E}_{\mu^y}[\mathsf{QoI}]| = 2^{-\ell+1}$. We run Algorithm 8 100 independent times, obtaining 50,000 samples on each level for every run. The accuracy of the theoretical rates in Theorem 5.4.1 is numerically verified in Figure 5.9. However, the sample mean of QoI_ℓ obtained with the sub-sampling algorithm does not decay as $2^{-\ell}$, confirming the bias of the sub-sampling ML-MCMC algorithm (Figure 5.9, top left). The decay rates α_w and β, corresponding to the decay in weak and strong error, respectively, are verified to be 1 for the ML-MCMC algorithm with fixed proposals, as theoretically expected (Figure 5.9, top right and bottom left). The optimal number of samples per level is presented in Figure 5.9 (bottom left). Again, the sub-sampling ML-MCMC provides a smaller number of samples and variances than the proposed method, but at the cost of a biased estimator. Furthermore, Figure 5.10 reveals that the synchronization rate of both methods tends to 1 with ℓ, as expected.

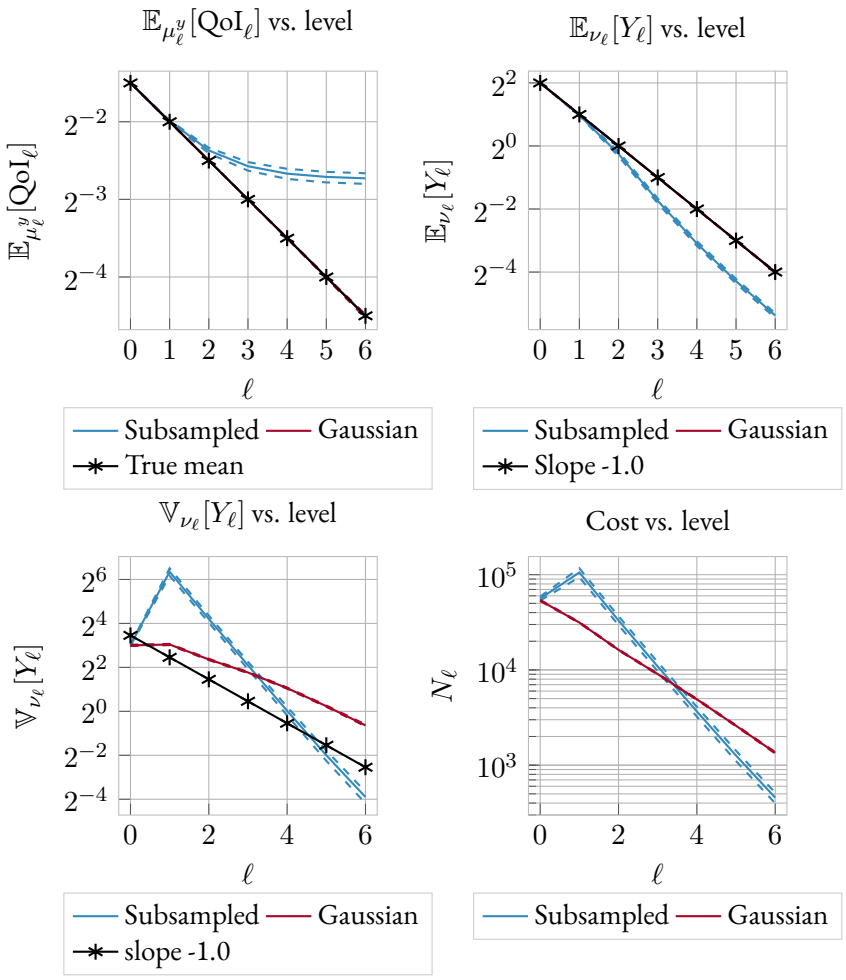

Figure 5.9: (Top left) Estimated expected value of QoI_ℓ for both ML-MCMC algorithms and the true mean of QoI_ℓ for different values of ℓ. (Top right) Expected value of $Y_\ell = \mathsf{QoI}_\ell - \mathsf{QoI}_{\ell-1}$ obtained with both algorithms for different values of ℓ. (Bottom left): Variance of Y_ℓ obtained with both algorithms for different values of ℓ. (Bottom right): Number of samples per level for each method with $\mathsf{tol} = 0.07$. On all plots, dashed lines represent a 95% confidence interval estimated over 100 independent runs of each algorithm.

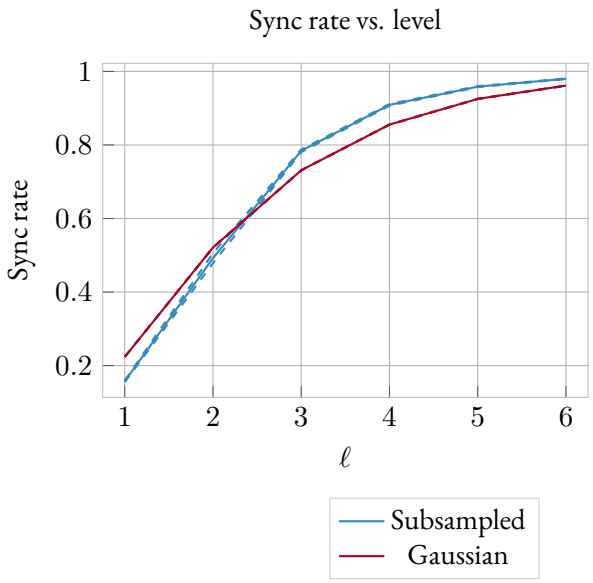

Figure 5.10: Synchronization rate for both algorithms. Dashed lines represent a 95% confidence interval. As expected, the chains become more and more synchronized as the number of levels increases.

Lastly, we once again perform some robustness experiments for our C-ML-MCMC algorithm. We run Algorithm 9 using the same level independent proposals $Q_\ell = Q = \mathcal{N}(2, 3)$ for three different prescribed tolerances $\mathsf{tol} = \{0.1, 0.07, 0.06\}$ for a total of 100 independent runs. Similar as in the previous example, for each independent run k of the C-ML-MCMC algorithm, we compute er_k^2 as in (5.29) and plot it in Figure 5.11. Once again, we obtain estimators whose mean square error is close to the prescribed tolerance, as desired. This further evidences the robustness of Algorithm 9.

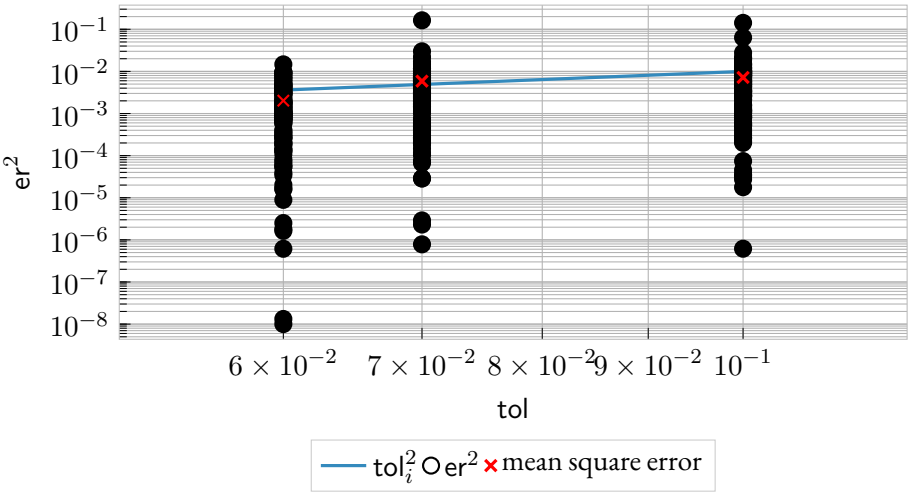

Figure 5.11: Total squared error er^2 vs tolerance tol for the moving Gaussian example.

5.6.3 Subsurface flow

We consider a slightly more challenging problem in which we aim to recover the probability distribution of the stochastic permeability field in Darcy's subsurface flow equation (5.30), based on some noise-polluted measured data. In particular, let $\bar{D} = [0,1]^2$, $\mathsf{X} = \mathbb{R}^4$, $(x_1, x_2) =: x \in \bar{D}$, $\partial D = \Gamma_N \cup \Gamma_D$, with $\mathring{\Gamma}_N \cap \mathring{\Gamma}_D = \emptyset$, where $\Gamma_D := \{(x_1, x_2) \in \partial D, \text{ s.t. } x_1 = \{0, 1\}\}$, and $\Gamma_N = \partial D \backslash \Gamma_D$. Darcy's subsurface equation is given by

$$\begin{cases} -\nabla_x \cdot (\kappa(x,u)\nabla_x \mathsf{p}(x,u)) = 1, & x \in D,\ u \in \mathsf{X}, \\ \mathsf{p}(x,u) = 0 & x \in \Gamma_D,\ u \in \mathsf{X}, \\ \partial_n \mathsf{p}(x,u) = 0 & x \in \Gamma_N,\ u \in \mathsf{X}, \end{cases} \tag{5.30}$$

where p represents the pressure (or hydraulic head), and we model the stochastic permeability $\kappa(x,u)$ for $(u_1, u_2, u_3, u_4) =: u \in \mathsf{X}$, as

$$\kappa(x,u) = \exp\left(u_1 \cos(\pi x) + \frac{u_2}{2}\sin(\pi x) + \frac{u_3}{3}\cos(2\pi x) + \frac{u_4}{4}\sin(2\pi x)\right),$$

with $u_i \sim \mathcal{N}(0,1)$, $i = 1,2,3,4$. Data y is modeled by the solution of Equation (5.30) observed at a grid of 9×9 equally-spaced points in D (hence $\mathsf{Y} = \mathbb{R}^{9\times 9}$) and polluted by a normally-distributed noise $\eta \sim \mathcal{N}(0, \sigma^2_{\text{noise}} I_{81\times 81})$, with $\sigma_{\text{noise}} = 0.004$, which corresponds to approximately 1% noise and $I_{81\times 81}$ is the 81-dimensional identity matrix. At each discretization level $\ell \geq 0$, the solution to Equation (5.30) is numerically approximated using the finite element method on a triangular mesh of $2^\ell \cdot 16 \times 2^\ell \cdot 16$ elements, which is computationally implemented using the FEniCS library [101]. Such a library includes optimal solvers for the forward model, for which γ can be reasonably taken equal to 1. Thus, the map $u \mapsto \mathcal{F}_\ell(u)$ is to be understood as the

numerical solution of Equation (5.30) at a discretization level ℓ, observed at a grid of 9×9 equally spaced points, for a particular value of $u \in \mathsf{X}$. This, in turn, induces a level dependent potential

$$\Phi_\ell(u; y) := \frac{1}{2\sigma^2_{\text{noise}}} \|y - \mathcal{F}_\ell(u)\|^2 ,$$

and prior $\mu_{\mathsf{pr}} = \mathcal{N}(0, I_{4\times 4})$. In the above expressions, $\|\cdot\|$ denotes the Frobenius norm on $\mathbb{R}^{9\times 9}$. Given that we are on a finite-dimensional setting, μ_{pr} has a density with respect to the Lebesgue measure, and as such, we can define the un-normalized posterior density $\tilde{\pi}^y_\ell : \mathsf{X} \mapsto \mathbb{R}_+$ w.r.t the Lebesgue measure given by

$$\tilde{\pi}^y_\ell(u) = \exp\left(-\Phi_\ell(u; y) - \frac{1}{2}u^T u\right).$$

As a quantity of interest we consider the average pressure over the physical domain, that is, $\mathsf{QoI}(u) = \int_D \mathsf{p}(x, u)\mathrm{d}x$. We implement our ML-MCMC algorithm to approximate $\mathbb{E}_{\mu^y}[\mathsf{QoI}]$. In particular, we use RWM at level 0 with Gaussian proposals $\mathcal{N}(0, \sigma^2_{\text{rwm}} I_{4\times 4})$ with step-size $\sigma_{\text{rwm}} = 0.05$, which produces an acceptance rate of about 24%. For the proposal Q_ℓ at higher levels $\ell \geq 1$, we use a mixture between the prior and a KDE obtained from the samples obtained at the previous level $\ell - 1$. This choice of mixture is made so that Assumption 5.3.1 holds.
We begin by numerically verifying the converge rates stated in Theorem 5.4.1. To that end, we run Algorithm 8 20 independent times, obtaining 10,000 samples per run at each level $\ell = 0, 1, 2, 3$. We plot the obtained rates in Figure 5.12. As we can see, we numerically verify that $\alpha_w \approx \beta (\approx 2.0)$, as predicted by our theory; this follows since QoI is smooth, and as such, one should expect the number of moments m to be large, and since $\alpha = 2$ for our FE implementation (see, e.g., [20]).
Lastly, we once again perform some robustness experiments for our C-ML-MCMC algorithm, with $\mathsf{L}_{\text{max}} = 3$. To that end, we first estimate $\mathbb{E}_{\mu^y}[\mathsf{QoI}] \approx \mu^y_4(\mathsf{QoI}_4)$ by performing 50 independent runs of a single-level MCMC algorithm at a discretization level $\ell = 4$, obtaining 2000 samples on each simulation. In particular, each independent run implements a RWM sampler, using proposals given by $\mathcal{N}(0, \sigma^2_{\text{rwm}} I_{4\times 4})$ with step-size $\sigma_{\text{rwm}} = 0.05$, which produces an acceptance rate of about 21%. We run Algorithm 9 using the same mixture of independent proposals as before for different tolerance levels $\mathsf{tol} = \{1.1 \times 10^{-4}, 2.0 \times 10^{-4}, 3.0 \times 10^{-4}\}$. The C-ML-MCMC algorithm is run 20 independent times for each tolerance tol_i. For each independent run $k = 1, 2, \ldots, 20$, let $\widehat{\mathsf{QoI}}^{(k)}_{\mathsf{L}^{(k)}(\mathsf{tol}_i), \{N_\ell\}_{\ell=0}^{\mathsf{L}^{(k)}}, \mathsf{tol}_i}$, with $\mathsf{L}(\mathsf{tol}_i) \leq 3$, denote the ML estimator obtained from the k^{th} run at tolerance tol_i. We compute the (approximate) total error squared $\tilde{\mathsf{er}}^2_{i,k}$ at the k^{th} run with a tolerance tol_i as

$$\tilde{\mathsf{er}}^2_{i,k} = \left(\widehat{\mathsf{QoI}}^{(k)}_{\mathsf{L}^{(k)}(\mathsf{tol}_i), \{N_\ell\}_{\ell=0}^{\mathsf{L}^{(k)}}, \mathsf{tol}_i} - \tilde{\mu}^y_4(\mathsf{QoI}_4)\right)^2, \tag{5.31}$$

and plot it vs a given tolerance in Figure 5.13. As expected, the MSE of the obtained estimators is less than the prescribed tolerance. This further evidences the robustness of Algorithm 9.

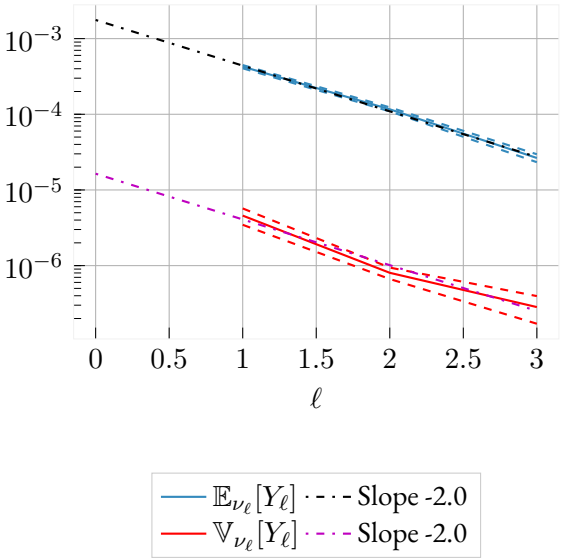

Figure 5.12: Decays of $\mathbb{E}_{\nu_\ell}[Y_\ell]$ and $\mathbb{V}_{\nu_\ell}[Y_\ell]$ vs level ℓ.As we can see, both quantities decay with the same rate, as predicted by the theory.

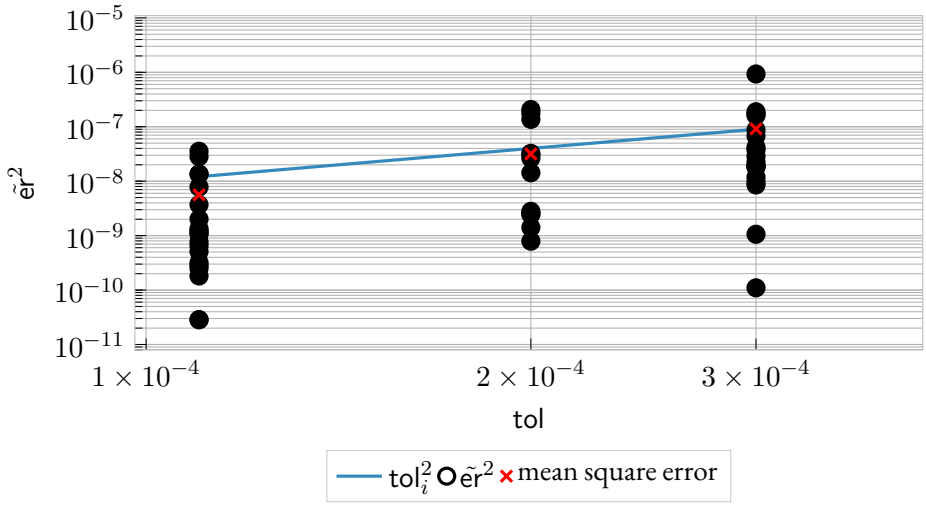

Figure 5.13: Computed squared error er^2 (using Equation 5.31) vs tol for the elliptic PDE example.

5.6.4 High dimensional subsurface flow with Laplace's approximation

Lastly, let us consider a more interesting problem given by a high dimensional example. Consider once again the same setting as in Section 5.6.3 namely the Darcy's subsurface flow (5.30) with

$\kappa(x,u) = e^{u(x)}$, with $u(x) \sim \mathcal{N}(0, \mathcal{A}^{-2}) = \mathcal{N}(0, \mathcal{C}) = \mu_{\mathsf{pr}}$ where the precision operator is the square of the differential operator $\mathcal{A}$ acting on a dense subspace $\mathrm{Dom}(\mathcal{A}) \subset L_2(D)$ of the form

$$\mathcal{A} = -\Delta + \frac{1}{2}I,$$

together with Robbin boundary conditions $\nabla(\cdot)\cdot\hat{n} + \frac{\sqrt{2}}{2}(\cdot) = 0$ (c.f. Sections 2.2.1 and 4.5.7). We assume that the data y is generated by solving equation (5.30), using a realization $u_{\text{true}} \sim \mu_{\mathsf{pr}}$, and observing it at a grid of 10×10 equally spaced points in $[0.1, 0.9]^2$, polluted by some normally distributed noise $\eta \sim \mathcal{N}(0, \sigma^2_{\text{noise}} I_{100\times 100})$ with $\sigma_{\text{noise}} = 9.61 \times 10^{-5}$, corresponding to roughly 1% noise. Denoting by $u \mapsto \mathcal{F}(u)$ the mapping associated to solving Equation (5.30) with $\kappa(x,u) = e^{u(x)}$ and observing the solution at the given grid of points, we can then pose our BIP as sampling from μ^y with

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u) = \frac{1}{Z}\exp\left(-\|y - \mathcal{F}(u)\|^2_\Sigma\right), \quad \Sigma = \sigma^2_{\text{noise}} I_{100\times 100}.$$

Once samples from μ^y have been obtained, we aim at approximating $\mathbb{E}_{\mu^y}[\mathsf{QoI}]$ where the quantity of interest $\mathsf{QoI} : \mathsf{X} \to \mathbb{R}$ is the log-flux through the bottom boundary $\Gamma_b := \{(x_1, x_2) \in \partial D \text{ s.t. } x_2 = 0\}$ defined as

$$\mathsf{QoI}(u) := \log\left(\int_{\Gamma_b} e^{u(x)}\nabla \mathsf{p}\cdot\hat{n}\,\mathrm{d}s\right),$$

where p is the solution to (5.30) and $\hat{n}$ denotes the unit normal vector to Γ_b. In order to implement this, we introduce a sequence of discretization levels $\ell = 0, 1, 2, 3 = \mathsf{L}$ of the forward mapping operator $\mathcal{F}$ by numerically approximating Equation (5.30) using the finite-element method with $16\cdot 2^\ell \times 16 \cdot 2^\ell$ piece-wise-linear finite elements. We denote by $\{\mathsf{X}_\ell\}_{\ell=0}^{\mathsf{L}}$ the sequence of finite-element spaces and by $\{\mathcal{F}_\ell\}_{\ell=0}^{\mathsf{L}}$ the sequence of approximate forward operators. We also introduce a finite-dimensional approximation $u_{\ell,\ell} \in \mathsf{X}_\ell$ of the state variable using the projection operator $\mathcal{P}_\ell^{\mathcal{A}}$ introduced in Section 2.2.3, namely, $u_\ell = \mathcal{P}_\ell^{\mathcal{A}} u$ is such that $\langle \mathcal{A}u_{\ell,\ell}, v_\ell\rangle = \langle \mathcal{A}u, v_\ell\rangle, \forall v_\ell \in \mathsf{X}_\ell$, where $\langle\cdot,\cdot\rangle$ denotes the duality pairing between $H^1(D)$ and its dual, and $\langle\mathcal{A}\cdot,\cdot\rangle$ can be understood as

$$\langle \mathcal{A}u, v\rangle = \int_D \nabla u(x)\cdot\nabla v(x) + \frac{1}{2}u(x)v(x)\mathrm{d}x + \int_{\partial D}\frac{\sqrt{2}}{2}u(x)v(x)\mathrm{d}x \quad \forall u, v \in H^1(D).$$

Together, $\mathcal{F}_\ell$ and $\mathcal{P}_\ell^{\mathcal{A}}$ induce a sequence of approximate potentials $\Phi_\ell(u;y) = \tilde{\Phi}(\mathcal{F}_\ell(\mathcal{P}_\ell^{\mathcal{A}}u); y)$, $\ell = 0, 1, \ldots, \mathsf{L}$, and a corresponding sequence of posterior measures $\{\mu_\ell^y\}_{\ell=0}^{\mathsf{L}}$ defined on the whole state space X, which can however be factorized for $u_{\ell,\ell} = \mathcal{P}_\ell^{\mathcal{A}}u$ and $z_\ell = u - u_{\ell,\ell}$ as

$$\mu_\ell^y(\mathrm{d}u) = \mu_\ell^y(\mathrm{d}u_{\ell,\ell}, \mathrm{d}z_\ell) = \widehat{\mu_\ell^y}(\mathrm{d}u_{\ell,\ell})\widehat{\mu_{\mathsf{pr}}}(u_{\ell,\ell}, \mathrm{d}z_\ell),$$

with $\widehat{\mu_\ell^y}(\mathrm{d}u_{\ell,\ell}) = Z_\ell^{-1} \exp\left(-\tilde{\Phi}(\mathcal{F}_\ell(u_{\ell,\ell}); y)\right) \mu_{\mathsf{pr}_\ell}(\mathrm{d}u_{\ell,\ell})$. We finally consider a sequence of approximations QoI_ℓ of QoI, given by $\mathsf{QoI}_\ell(u) = \mathsf{QoI}(u_{\ell,\ell})$. Notice that

$$\begin{aligned}\mathbb{E}_{\mu_\ell^y}[\mathsf{QoI}_\ell] &= \int_{\mathsf{X}} \mathsf{QoI}(\mathcal{P}_\ell^{\mathcal{A}} u)\mu_\ell^y(\mathrm{d}u) = \int_{\mathsf{X}} \mathsf{QoI}(u_{\ell,\ell})\widehat{\mu_{\mathsf{pr}}}(u_{\ell,\ell}, \mathrm{d}z_\ell)\widehat{\mu_\ell^y}(\mathrm{d}u_{\ell,\ell}) \\ &= \int_{\mathsf{X}_\ell} \mathsf{QoI}(u_{\ell,\ell})\widehat{\mu_\ell^y}(\mathrm{d}u_{\ell,\ell}) = \mathbb{E}_{\widehat{\mu_\ell^y}}[\mathsf{QoI}],\end{aligned}$$

i.e., with the goal of computing the posterior expectation of QoI_ℓ, only the posterior measure $\widehat{\mu_\ell^y}$ on X_ℓ matters, so that we can forget about the conditional distribution $\widehat{\mu_{\mathsf{pr}}}(u_{\ell,\ell}, \mathrm{d}z_\ell)$ and restrict our analysis to just the finite-dimensional space X_ℓ. In view of constructing coupled chains on levels $\ell, \ell-1$, we also remark that $u_{\ell,\ell-1} = \mathcal{P}_{\ell-1}^{\mathcal{A}} u = \mathcal{P}_{\ell-1}^{\mathcal{A}} u_{\ell,\ell}$ so that, to build the posterior measure $\mu_{\ell-1}^y$ at level $\ell-1$ on the full space X reads

$$\begin{aligned}\mu_{\ell-1}^y(\mathrm{d}u) &= \frac{1}{Z_{\ell-1}} \exp\left(-\tilde{\Phi}(\mathcal{F}_{\ell-1}(\mathcal{P}_{\ell-1}^{\mathcal{A}} u); y)\right) \mu_{\mathsf{pr}}(\mathrm{d}u) \\ &= \frac{1}{Z_{\ell-1}} \exp\left(-\tilde{\Phi}(\mathcal{F}_{\ell-1}(\mathcal{P}_{\ell-1}^{\mathcal{A}} u_{\ell,\ell}); y)\right) \mu_{\mathsf{pr}_\ell}(\mathrm{d}u_{\ell,\ell})\widehat{\mu_{\mathsf{pr}}}(u_{\ell,\ell}, \mathrm{d}z),\end{aligned}$$

and we can restrict the measure to X_ℓ giving a posterior

$$\widehat{\mu_{\ell,\ell-1}^y}(\mathrm{d}u_{\ell,\ell}) := \frac{1}{Z_{\ell-1}} \exp\left(-\tilde{\Phi}(\mathcal{F}_{\ell-1}(\mathcal{P}_{\ell-1}^{\mathcal{A}} u_{\ell,\ell}); y)\right) \mu_{\mathsf{pr}_\ell}(\mathrm{d}u_{\ell,\ell}).$$

Our goal is then to construct the coupled chains at levels $\ell, \ell-1,\ \ell \geq 1$ in the higher dimensional space space $\mathsf{X}_\ell \times \mathsf{X}_\ell$, which is achieved by using a *high-dimensional proposal* $z_\ell \in \mathsf{X}_\ell$ for both chains, in such a way that the state z_ℓ is "down-sampled" (i.e., projected onto $\mathsf{X}_{\ell-1}$, deterministically) when evaluating the posterior density and the quantity of interest at level $\ell-1$. Denoting by π_ℓ^y, $\bar{\pi}_{\ell-1}^y$ and Q_ℓ the μ_{pr_ℓ}-densities of $\widehat{\mu_\ell^y}$, $\widehat{\mu_{\ell,\ell-1}^y}$ and of the proposal, respectively, one then has that the MH acceptance ratios are given by

$$\begin{aligned}\alpha_\ell(u_{\ell,\ell}, z_\ell) &= \min\left\{1, \frac{\pi_\ell^y(z_\ell)}{\pi_\ell^y(u_{\ell,\ell})}\frac{Q_\ell(u_{\ell,\ell})}{Q_\ell(z_\ell)}\right\}, \\ \alpha_{\ell-1}(u_{\ell,\ell-1}, z_\ell) &= \min\left\{1, \frac{\bar{\pi}_{\ell-1}^y(z_\ell)}{\bar{\pi}_{\ell-1}^y(u_{\ell,\ell-1})}\frac{Q_\ell(u_{\ell,\ell-1})}{Q_\ell(z_\ell)}\right\}.\end{aligned}$$

Remark 5.6.1: *Alternatively, one could construct an "up-sampled" approach, where one aims at generating coupled chains in the space $\mathsf{X}_{\ell-1} \times \mathsf{X}_{\ell-1}$, using an IMH proposal $Q_{\ell-1}$ on the coarse space $\mathsf{X}_{\ell-1}$ as follows:*

1. *Sample $z_{\ell-1} \sim Q_{\ell-1},\ z_{\ell-1} \in \mathsf{X}_{\ell-1}$.*

2. *Given the coarse state* $z_{\ell-1} =: z_{\ell,coarse}$*, generate its complement on the fine-state by, e.g., a Gaussian process regression conditioned on* $z_{\ell,coarse}$*, i.e.,* $z_{\ell,fine} \sim \mathcal{GP}(\cdot|z_{\ell,coarse})$.

3. *Set* $z_\ell = (z_{\ell,coarse}, z_{\ell,fine})$.

4. *Accept or reject* z_j*,* $j = \ell - 1, \ell$*, with MH acceptance probabilities given by*

$$\alpha_\ell(u_{\ell,\ell}, z_\ell) = \min\left\{1, \frac{\pi^y_\ell(z_\ell)}{\pi^y_\ell(u_{\ell,\ell})} \frac{Q_{\ell-1}(u_{\ell,\ell,coarse})}{Q_{\ell-1}(z_{\ell,coarse})} \frac{\mathcal{GP}(u_{\ell,\ell,fine}|u_{\ell,\ell,coarse})}{\mathcal{GP}(z_{\ell,fine}|z_{\ell,coarse})}\right\},$$
$$\alpha_{\ell-1}(u_{\ell,\ell-1}, z_{\ell-1}) = \min\left\{1, \frac{\pi^y_{\ell-1}(z_{\ell-1})}{\pi^y_{\ell-1}(u_{\ell,\ell-1})} \frac{Q_\ell(u_{\ell,\ell-1})}{Q_\ell(z_{\ell-1})}\right\},$$

where we have set $u_{\ell,\ell} := (u_{\ell,\ell,coarse}, u_{\ell,\ell,fine})$*, with* $u_{\ell,\ell,coarse}$ *in* $\mathsf{X}_{\ell-1}$*, and* $u_{\ell,\ell} \in \mathsf{X}_\ell$.

However, we chose not to investigate this approach further.

Constructing an efficient Laplace approximation

We follow the procedure of [23, 24], where a Laplace-approximation to μ^y_ℓ is constructed using a low-rank covariance matrix. For each level $\ell = 0, 1, \ldots, \mathsf{L}$, we aim at constructing a proposal $\tilde{Q}_\ell = \mathcal{N}(m_{\text{map},\ell}, \mathcal{C}_{\text{Lap},\ell})$, where

$$m_{\text{map},\ell} = \arg\min_{u\in\mathsf{X}} \left(\frac{1}{2}\|y - \mathcal{F}_\ell(u)\|^2_\Sigma + \frac{1}{2}\|u\|^2_{\mathcal{C}_\ell}\right), \tag{5.32}$$
$$\mathcal{C}_{\text{Lap},\ell} = \left(\mathcal{H}_\ell(m_{\text{map},\ell}) + \mathcal{C}_\ell^{-1}\right)^{-1},$$

where $\mathcal{H}_\ell(m_{\text{map},\ell}) \in \mathbb{R}^{K_\ell \times K_\ell}$ is the Hessian of $\Phi(u; y) = \|y - \mathcal{F}_\ell(u)\|^2_\Sigma$ evaluated at $m_{\text{map},\ell}$, and $\mathcal{C}_\ell \in \mathbb{R}^{K_\ell \times K_\ell}$ is the symmetric positive-definite matrix representing the covariance operator $\mathcal{C}$ at discretization level ℓ (i.e., $C_\ell^{-1} = A_\ell^{-1} M_\ell A_\ell^{-1}$, where A_ℓ and M_ℓ are the stiffness and mass matrices defined in 2.2.3. Notice that the optimization problem (5.32) can be understood as minimizing $\Phi(u; y)$ with Tykhonov regularization [158] given by $\frac{1}{2}\|u\|^2_{\mathcal{C}_\ell}$, and as such, such an optimization problem is well-posed, provided that the regularization is strong enough ([85, 158]). The computation of the gradient, together with the Hessian of the misfit $\mathcal{H}_\ell(m_{\text{map},\ell})$ can be computed using adjoint state methods, together with a Lagrangian formulation of the optimization problem (see, e.g., [164]).
It is typically inefficient to construct $\mathcal{C}_{\text{Lap},\ell}$ directly; instead, the work [23] overcomes this issue by proposing a low-rank approximation, summarized as follows (see, e.g., the works [23, 164] for a detailed derivation). Write $\mathcal{H}_\ell = \mathcal{H}_\ell(m_{\text{map},\,\ell})$, and consider the following generalized symmetric eigenproblem

$$\mathcal{H}_\ell v_i = \lambda_i \mathcal{C}_\ell^{-1} v_i, \quad \lambda_1 \geq \lambda_2 \geq \ldots \lambda_{K_\ell}, v_i \in \mathbb{R}^{K_\ell}.$$

It is known [164] that under some technical conditions λ_i decays rapidly. Thus, choosing $r_\ell \ll K_\ell$ such that $\lambda_{r_\ell+1} \ll 1$, and defining

$$V_{r_\ell} := [v_1, v_2, \dots, v_{r_\ell}], \quad \Lambda_{r_\ell} := \text{diag}(\lambda_1, \lambda_2, \dots \lambda_{r_\ell}), \text{ with } V_{r_\ell} \mathcal{C}_\ell^{-1} V_{r_\ell}^T = I_{r_\ell \times r_\ell},$$

it then follows from the Sherman-Morrison-Woodbury formula [161] that one can construct the following low-rank approximation of $\mathcal{C}_{\text{Lap},\ell}$:

$$\begin{aligned}\mathcal{C}_{\text{Lap},\ell} &= \left(\mathcal{H}_\ell(m_{\text{map},\ell}) + \mathcal{C}_\ell^{-1}\right)^{-1} = \mathcal{C}_\ell - V_{r_\ell} D_{r_\ell} V_{r_\ell}^T + O\left(\sum_{i=r_\ell+1}^{K_\ell} \frac{\lambda_i}{\lambda_i + 1}\right) \\ &\approx \mathcal{C}_\ell - V_{r_\ell} D_{r_\ell} V_{r_\ell}^T =: \tilde{\mathcal{C}}_{\text{Lap},\ell},\end{aligned}$$

where $D_{r_\ell} := \text{diag}(\lambda_1/(\lambda_1+1), \dots, \lambda_{r_\ell}/(\lambda_{r_\ell}+1)) \in \mathbb{R}^{r_\ell \times r_\ell}$. It is known that the generalized eigenpairs $(\lambda_i, v_i), i = 1, \dots, K_\ell$ can be efficiently obtained using randomized eigensolvers [67, 148], provided that the spectrum of $\mathcal{H}_\ell$ decays sufficiently fast. From a computational perspective, the minimization procedure, together with the low-rank approximation of $\mathcal{C}_{\text{Lap},\ell}$ is efficiently implemented using the `hIPPYlib` library [164] of the `FEniCS` package [101].

Construction of the sampler

At each iteration of the coupled MCMC algorithm we sample as a proposal

$$z_\ell \sim \mathcal{N}(m_{\text{map},\ell}, \tilde{\mathcal{C}}_{\text{Lap},\ell}),$$

where the efficient sampling from $\mathcal{N}(m_{\text{map},\ell}, \tilde{\mathcal{C}}_{\text{Lap},\ell})$ can also be efficiently implemented using `hIPPYlib` library [164]. We constrct the level-wise MH acceptance probability. Write $H_\ell = \tilde{\mathcal{C}}_{\text{Lap},\ell}^{-1} - \mathcal{C}_\ell^{-1}$. It is shown in [129, Lemma 3.3] and [130, Section 3.2] that $\tilde{Q}_\ell = \mathcal{N}(m_{\text{map},\ell}, \mathcal{C}_{\text{Lap},\ell}) \simeq \mu_{\text{pr},\ell}$ with

$$\frac{\mathrm{d}\tilde{Q}_\ell}{\mathrm{d}\mu_{\text{pr},\ell}}(u_{\ell,\ell}) = \exp\left(\langle u_{\ell,\ell} - m_{\text{map},\ell}, m_{\text{map},\ell}\rangle_{\mathcal{C}_\ell} - \frac{1}{2}\left\|u_{\ell,\ell} - m_{\text{map},\ell}\right\|_{H_\ell^{-1}}^2 + \frac{1}{2}\left\|m_{\text{map},\ell}\right\|_{\mathcal{C}_\ell}^2\right) =: Q_\ell(u).$$

Furthermore, setting $\tilde{Q}_\ell = Q_{\text{ref},\ell} = \tilde{\nu}_\ell$ in the notation of Lemma 3.4.1, it then follows that $\frac{\mathrm{d}\mu_\ell^y}{\mathrm{d}\tilde{\nu}_\ell}(u) = \frac{\mathrm{d}\mu_\ell^y}{\mathrm{d}\tilde{Q}_\ell}(u_{\ell,\ell}) = \frac{\mathrm{d}\mu_\ell^y}{\mathrm{d}\mu_{\text{pr},\ell}}(u_{\ell,\ell}) \cdot \left(\frac{\mathrm{d}\tilde{Q}_\ell}{\mathrm{d}\mu_{\text{pr},\ell}}(u_{\ell,\ell})\right)^{-1}$, and the MH algorithm with target measure μ_ℓ^y induced by taking $\tilde{Q}_\ell$ as an independent proposal is well defined, with

$$\begin{aligned}\alpha_\ell(u_{\ell,\ell}, z_\ell) &= \min\left\{1, \frac{\pi_\ell^y(z_\ell)}{\pi_\ell^y(u_{\ell,\ell})}\frac{Q_\ell(u_{\ell,\ell})}{Q_\ell(z_\ell)}\right\}, \\ \alpha_{\ell-1}(u_{\ell,\ell-1}, z_\ell) &= \min\left\{1, \frac{\bar{\pi}_{\ell-1}^y(z_\ell)}{\bar{\pi}_{\ell-1}^y(u_{\ell,\ell-1})}\frac{Q_\ell(u_{\ell,\ell-1})}{Q_\ell(z_\ell)}\right\},\end{aligned}$$

Furthermore, we assume that for any level ℓ, $\Phi_\ell(u_{\ell,\ell}; y)$ grows such that

$$\operatorname*{ess\,inf}_{u_{\ell,\ell}\in \mathsf{X}_\ell} \Phi_\ell(u_{\ell,\ell}; y) + \left\langle u_{\ell,\ell} - m_{\text{map},\ell}, m_{\text{map},\ell}\right\rangle_{\mathcal{C}} - \frac{1}{2}\left\|u_{\ell,\ell} - m_{\text{map},\ell}\right\|^2_{H^{-1}} + \frac{1}{2}\left\|m_{\text{map},\ell}\right\|^2_{\mathcal{C}} > a > -\infty,$$

thus, satisfying Assumption 5.3.1.1. Intuitively, one would expect this to happen whenever the posterior measure is more concentrated than the Laplace approximation proposal.

Results

We follow a similar procedure as in previous examples, and proceed to numerically verify the converge rates stated in Theorem 5.4.1. To that end, we run Algorithm 8 50 independent times, obtaining 2,000 samples per run at each level $\ell = 0, 1, 2, 3$, using as a proposal the Laplace-approximation of the posterior at level $\ell = 0, 1, 2, 3$ to construct the coupled chain, obtaining an acceptance rate and synchronization rate shown in Figure 5.17, where $\dim(\mathsf{X}_0) = 289$, $\dim(\mathsf{X}_1) = 1089$, $\dim(\mathsf{X}_2) = 4225$ and $\dim(\mathsf{X}_3) = 16641$. As we can see, the synchronization rate increases rapidly with ℓ, while the (marginal) acceptance rates converge quickly to the same dimension-independent value of around 0.56. As an illustration, we plot the MAP $m_{\text{map},\ell}$ at each level in Figure 5.15, where the difference in dimensionality between spaces is clearly appreciable. Notice that the MAP at each level is able to capture the main features of u_{true}. We remark that we use `hIPPYlib` [164] and `FEniCS` [101] to efficiently construct the Laplace approximation (i.e., to solve the minimization problem (5.32) and to construct the low-rank approximation of the covariance, taking $r_\ell = 100$ for all $\ell = 0, 1, 2, 3$). We depict three samples from μ^y_ℓ, $\ell = 0, 1, 2, 3$ obtained with our method in Figure 5.16. We plot the obtained rates for $\mathbb{E}_{\nu_\ell}[Y_\ell]$ and $\mathbb{V}_{\nu_\ell}[Y_\ell]$ in Figure 5.17, where transparent colors represent a 95% confidence interval. As we can see, we have that $\alpha_w \approx 1.4$, $\beta \approx 1.4$. Furthermore, we plot the joint distribution of $(\mathsf{QoI}_{\ell-1}, \mathsf{QoI}_\ell)$ for $\ell = 1, 2, 3$ on Figure 5.18. It is clear then that the samples become increasingly concentrated in the diagonal, as expected. Lastly, once again under the assumption that $\gamma = 1$, we estimate the number of required samples per level for different levels of tolerance, together with the total computational cost of the algorithm and plot them in Figure 5.19. The number of required samples per level at a given tolerance are shown in Figure 5.19 (left). As it can be seen, the amount of samples required decreases with ℓ. Notice that there is a rather small decrease between the number of samples at level $\ell = 0$ and level $\ell = 1$; this suggests that coarsest discretization level was, perhaps, too coarse. in Figure 5.19 (right), we plot the complexity of the ML-MCMC method, compared to that of the standard, single level MH's algorithm. As it can be seen, our proposed method has a much better complexity than its single level counterpart.

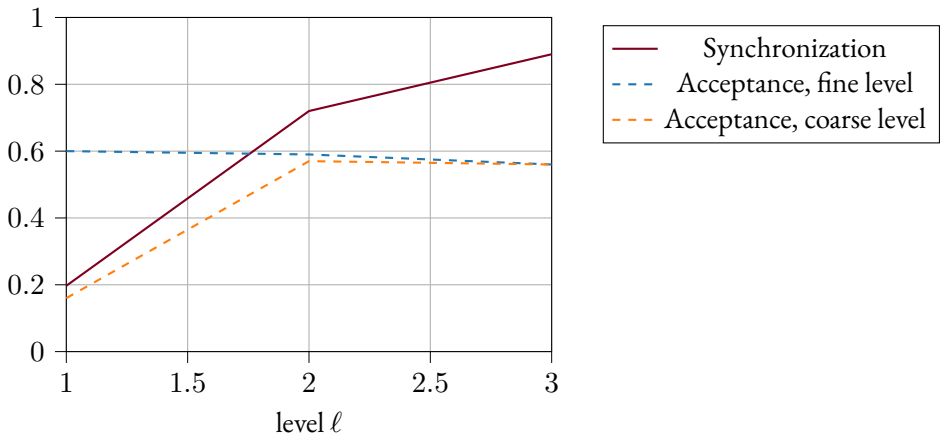

Figure 5.14: Plots of synchronization and acceptance rates using the Laplace-approximation proposal

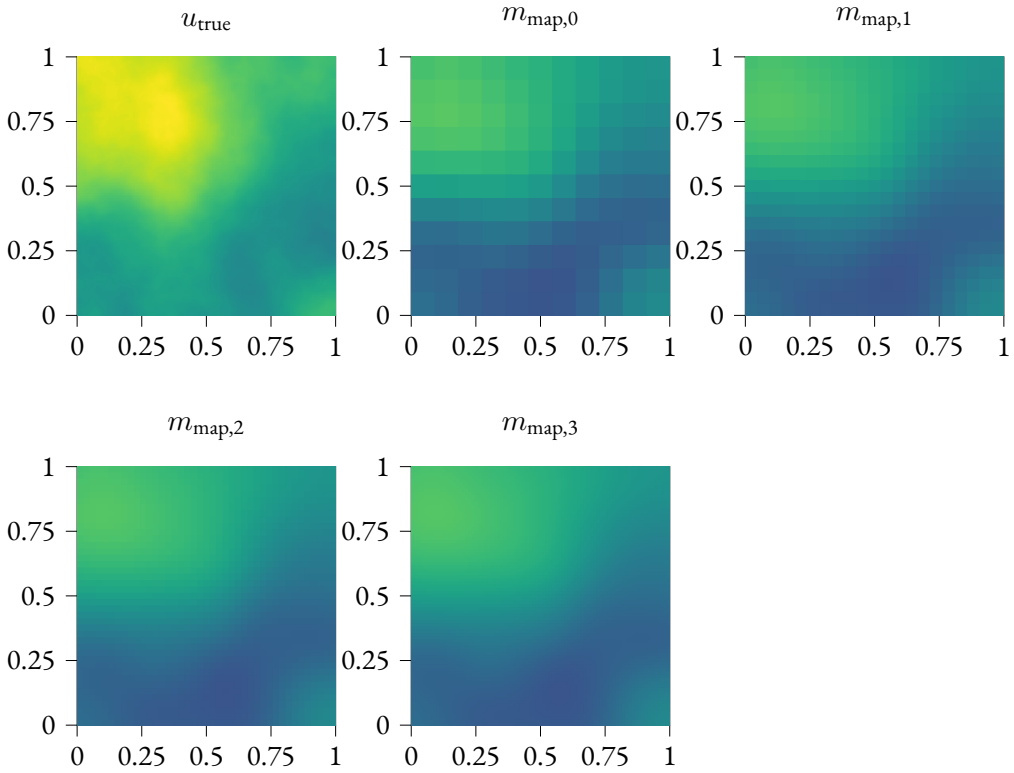

Figure 5.15: Realization of u_{true} and the MAP $m_{\text{map},\ell}$ at each level.

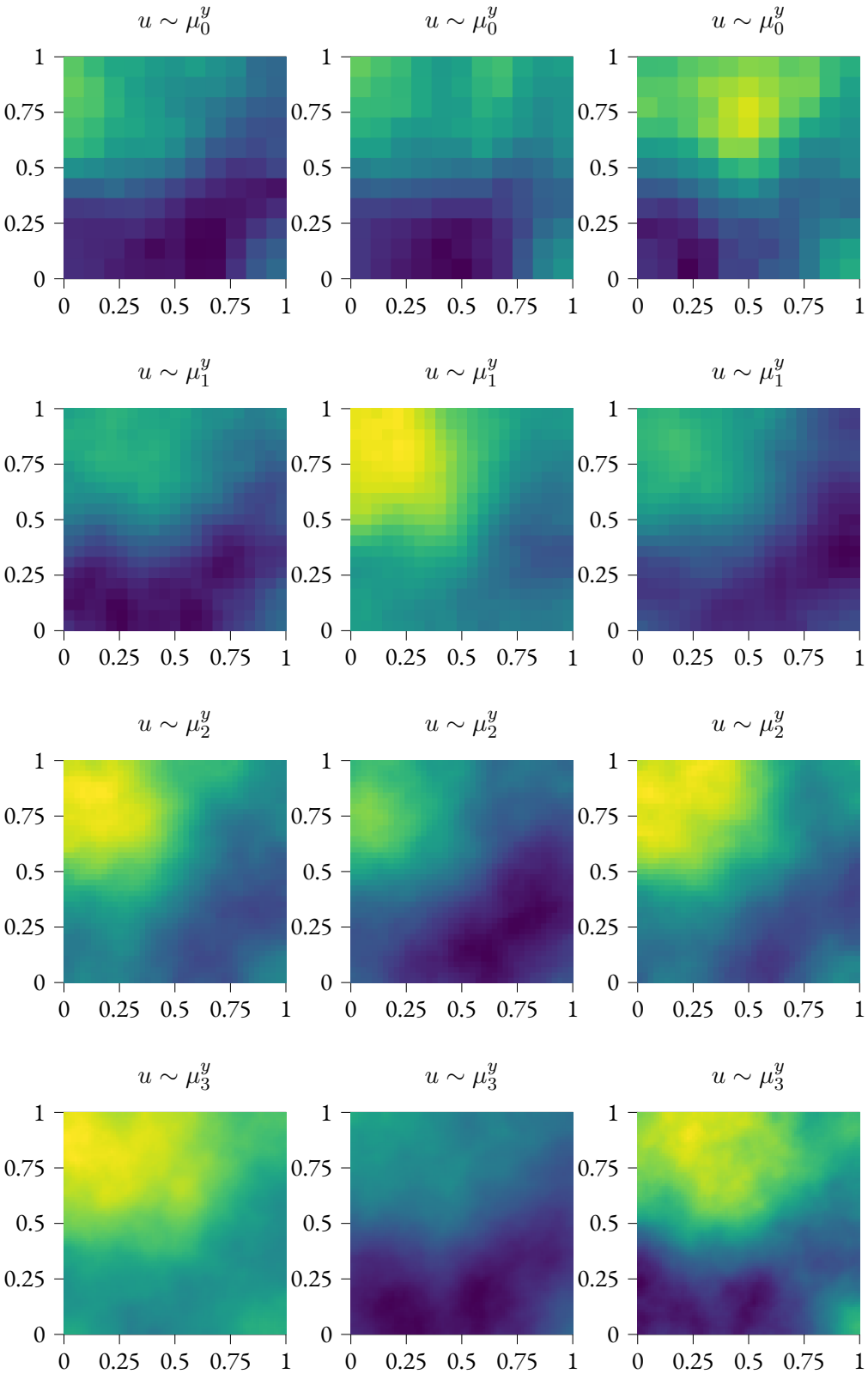

Figure 5.16: Three samples from $u \sim \mu_\ell^y$ per each level; from top to bottom $\ell = 0, 1, 2, 3$.

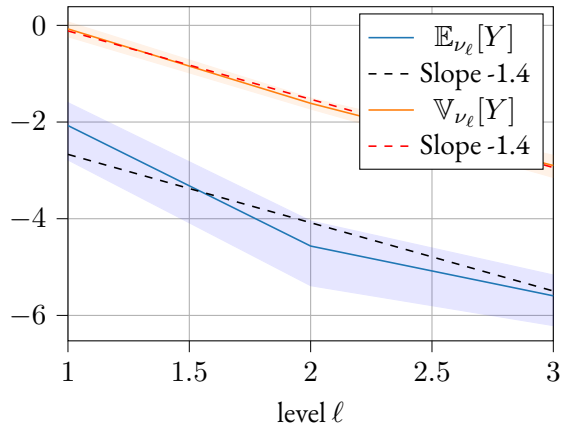

Figure 5.17: Plots of $\mathbb{E}_{\nu_\ell}[Y_\ell]$ and $\mathbb{V}_{\nu_\ell}[Y_\ell]$ (in $\log_2$-scale) vs level for the high-dimensional example.

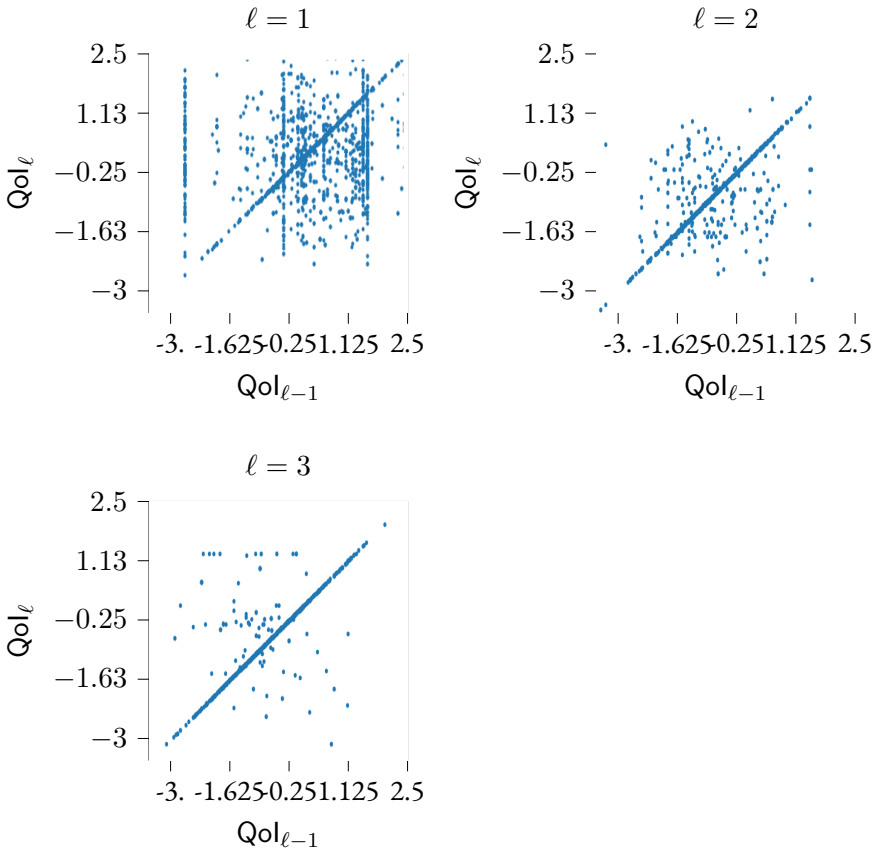

Figure 5.18: Joint samples of $(\mathsf{QoI}_{\ell-1}, \mathsf{QoI}_\ell)$ for different levels $\ell = 1, 2, 3$.

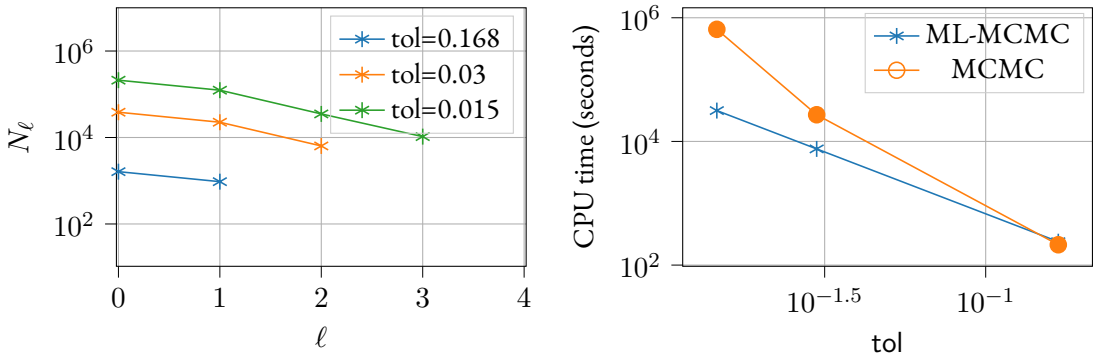

Figure 5.19: Costs for the high-dimensional example. Left: number of samples vs level for different tolerances. Right: complexity of ML-MCMC vs a single-level MCMC estimator.

6 Multi-level Markov chain Monte Carlo method based on maximally coupled proposals

6.1 Introduction

As discussed in Chapter 5, the crux of the ML-MCMC algorithm relies upon introducing a correlation between the chains $\{u^i_{\ell,\ell}\}_{i=1}^{N_\ell}$, $\{u^i_{\ell,\ell-1}\}_{i=1}^{N_\ell}$, at each given level ℓ. In the previous chapter, we introduced such a correlation by proposing the same state for both chains at each step in the MH algorithm, using an IMH sampler. In this chapter we present a novel type of ML-MCMC algorithms for which the correlation between chains is introduced by using a Metropolis-Hastings algorithm for each marginal chain $\{u^i_{\ell,\ell}\}_{i=1}^{N_\ell}$, $\{u^i_{\ell,\ell-1}\}_{i=1}^{N_\ell}$, in such a way that the proposal distributions for each chain are coupled using a so-called *maximal coupling* of the proposals. Such an algorithm allows for state-dependent proposals in the context of ML-MCMC. Being able to construct state-dependent proposals (like, e.g., RWM or pCN) in the context of ML-MCMC algorithms is particularly useful in those cases in which constructing a suitable IMH proposal (as discussed in the previous chapter), is difficult in some sense. This can occur, e.g., in the subsampling algorithm, whenever Assumption 5.4.1 is not satisfied by the posterior at the previous level (or more precisely, by the empirical measure approximating it), or when the Gaussian measure arising from the Laplace's approximation to the posterior measure is not sufficiently accurate, due to having few measurements or extremely noisy data. Also, our proposed methodology can also be of interest when solving the optimization problem associated to the construction of such an approximating measure (i.e., finding the MAP and the Hessian) is not feasible, which could be the case, e.g., when the forward mapping is computationally implemented using a so-called "black-box" and no information on the gradient of the cost-functional for the minimization problem is available.

In short, our method uses the following procedure. Supposing the chains are at a state $(u^n_{\ell,\ell-1}, u^n_{\ell,\ell})$, and each marginal chain is being constructed using possibly state-dependent proposals $Q_\ell(u^n_{\ell,\ell}, \cdot)$ and $R_{\ell-1}(u^n_{\ell,\ell-1}, \cdot)$, our proposed method samples a coupled state (v', u'), from a maximal cou-

pling of $Q_\ell(u^n_{\ell,\ell}, \cdot)$ and $R_{\ell-1}(u^n_{\ell,\ell-1}, \cdot)$. In practice, this means that (v', u') are sampled in such a way that $u' \sim Q_\ell(u^n_{\ell,\ell}, \cdot)$ and $v' \sim R_{\ell-1}(u^n_{\ell,\ell-1}, \cdot)$, with $\mathbb{P}(u' \neq v') = \left\| Q_\ell(u^n_{\ell,\ell}, \cdot) - R_{\ell-1}(u^n_{\ell,\ell-1}, \cdot) \right\|_{\text{tv}}$, where $\|\cdot\|_{\text{tv}}$ is the *total variation distance*. Each candidate state then gets accepted or rejected by its respective chain with the usual MH acceptance-rejection step. This procedure, allows us to use more flexible, non-necessarily independent proposals for each chain, such as Random Walk Metropolis (RWM) or preconditioned Crank Nicholson (pCN) (c.f. Section 3.4.1), while at the same time, creating chains that are highly correlated, as required by the ML-MCMC algorithm (see discussion on Section 5.2). More importantly, this type of ML-MCMC based on maximally coupled proposals, allows for more "flexibility" in the choice of proposals, while at the same time being easy to implement and for which marginal chains are geometrically ergodic under mild conditions. We show by numerical experimentation the effectiveness of our approach. Moreover, we present some elements of analysis, proving, in particular, the existence of a unique invariant measure for the level-ℓ coupled sampler.

The rest of this Chapter is organized as follows. We begin Section 6.2 by introducing an algorithm to sample from a maximal coupling between Gaussian probability measures (c.f. Algorithm 11), and then proceed to introduce our proposed method in Section (c.f. Algorithm 12). We present an analysis of the existence of and convergence to an invariant measure for the proposed algorithm in Section 6.3, and present some numerical experiments in Section 6.4.

6.2 ML-MCMC based on Maximal Coupling

We begin this section by recalling some basic concepts on the coupling of probability measures. We follow closely some of the theory presented in [19, 53, 76, 99]. Let X be a separable Banach space with associated Borel σ-algebra $\mathcal{B}(\mathsf{X})$, define the product space $\mathsf{X}^2 := \mathsf{X} \times \mathsf{X}$, and denote by $\mathcal{M}(\mathsf{X})$ the set of probability measures on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$. For any two probability measures $Q, R \in \mathcal{M}(\mathsf{X})$, we say that a measure $\gamma' \in \mathcal{M}(\mathsf{X}^2)$ is a *coupling* of Q and R if for any set $A \in \mathcal{B}(\mathsf{X})$,

$$\gamma'(A \times \mathsf{X}) = Q(A) \quad \text{and} \quad \gamma'(\mathsf{X} \times A) = R(A).$$

In words, we say that a probability measure γ' is a coupling of Q and R if its marginals are Q and R. It is known (see, e.g., [159]) that for any such coupling γ' it holds that

$$\|Q - R\|_{\text{tv}} \leq \mathbb{P}_{\gamma'}(\xi \neq \zeta) \quad (\xi, \zeta) \sim \gamma'. \tag{6.1}$$

We say that γ' is a *maximal coupling* of Q and R if γ' is a coupling such that equality holds for equation (6.1). It is always possible to find such a coupling under the assumption that X is a Polish space, as shown in [99, Theorem 5.2]. We remark, however, that such a maximal coupling is not necessarily unique.

Suppose, for the time being, that there exists an algorithm to efficiently sample from a maximal coupling between two (possibly state-dependent) proposal kernels $Q(u_{\ell,\ell}, \cdot)$ and $R(u_{\ell,\ell-1}, \cdot)$

(such as, e.g., RWM or pCN), in such a way that $R(u_{\ell,\ell-1},\cdot), Q(u_{\ell,\ell},\cdot)$ are the proposal kernels for a MH algorithm with invariant measure $\mu^y_{\ell-1}, \mu^y_\ell$, respectively. Denoting this sampling procedure as `Coupled-chain-MCMC`, one can then use such a coupling to create a ML-MCMC algorithm as shown in Algorithm 10. We will discuss next a possible way to generate such a coupling.

Algorithm 10 Multi-level MCMC

procedure ML-MCMC($\{\mu^y_\ell\}_{\ell=0}^{\mathsf{L}}, \{N_\ell\}_{\ell=0}^{\mathsf{L}}, \mu_{\mathsf{pr}}, \{Q_\ell, R_\ell\}_{\ell=0}^{\mathsf{L}}$)
  if $\ell = 0$ **then**
    # Create a chain at level $\ell = 0$ using any suitable MCMC algorithm
    $\{u_0^n\}$ =`MCMC`$(\mu_0^y, N_0, \dots)$. Set $\chi_{0,0} = \{u_0^n\}$.
  end if
  for $\ell = 1, \dots, \mathsf{L}$ **do**
    Sample $u^0_{\ell,\ell-1} \sim \mu_{\mathsf{pr}}$, and set $u^0_{\ell,\ell} = u^0_{\ell,\ell-1}$
    for $n = 0, \dots, N_\ell - 1$ **do**
      # Create a coupled chain using some coupling
      $\{u^{n+1}_{\ell,\ell-1}, u^{n+1}_{\ell,\ell}\}$ = `Coupled-chain-MCMC`$(\{\mu^y_{\ell-1}, \mu^y_\ell\}, \{u^n_{\ell,\ell-1}, u^n_{\ell,\ell}\}, \{R_\ell, Q_\ell\})$
    end for
    Set $\chi_{\ell,\ell} = \{u^n_{\ell,\ell}\}_{n=0}^{N_\ell}$, and $\chi_{\ell,\ell-1} = \{u^n_{\ell,\ell-1}\}_{n=0}^{N_\ell}$.
  end for
  Output $\chi_{0,0} \cup \{\chi_{\ell,\ell-1}, \chi_{\ell,\ell}\}_{\ell=0}^{\mathsf{L}}$ and $\widehat{\mathsf{QoI}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}$.
end procedure

6.2.1 Reflection maximal coupling for Gaussian proposals

We recall a technique used to sample from a given maximal coupling [76, 99, 100] between two Gaussian distributions. Further coupling strategies are presented in, e.g., [76]. We will focus our attention to couplings in finite-dimensional Hilbert spaces. This setting applies, e.g., when:

Case I. X itself is a finite-dimensional space, i.e., $\mathsf{X} = \mathbb{R}^K$, for some $K \geq 1$.

Case II. X is an infinite-dimensional Hilbert space that can be decomposed as $\mathsf{X}_\mathsf{L} \oplus \mathsf{X}_\mathsf{L}^\perp$ with $\mathsf{X}_\mathsf{L}^\perp \perp \mathsf{X}_\mathsf{L}$, where X_L is a K_L-dimensional subspace of X, for some $K_\mathsf{L} \geq 1$. This is the case, e.g., where we model $u_{\ell,\ell}, u_{\ell,\ell-1}$ in terms of their Karhunen-Loeve expansion, and aim at only coupling the proposal distribution of the first K_L terms in the expansion.

Case III. A similar case occurs when $\mathsf{X}_{\ell-1} \subset \mathsf{X}_\ell$ and one tries to couple the proposals for the first $K_{\ell-1}$ components of $u_{\ell,\ell}$ with all the $K_{\ell-1}$ components of $u_{\ell,\ell-1}$, while the fine modes of $u_{\ell,\ell}$ evolve independently of the coarse modes of $u_{\ell,\ell-1}$.

For notational simplicity and with a slight abuse of notation, hereafter we will denote by X either the $K-$-dimensional space of Case I, the K_L-dimensional subspace of Case II, or the $K_{\ell-1}$-dimensional subspace of Case III. For any $\ell = 1, 2 \dots, \mathsf{L}$ and any given step $n \in \mathbb{N}$, let

$(u^n_{\ell,\ell-1}, u^n_{\ell,\ell})$ be the current state of the joint chain constructed by Algorithm 10 with marginals $\mu^y_{\ell-1}, \mu^y_\ell$. Furthermore, suppose that each marginal chain is being constructed following the usual Metropolis-Hastings algorithm with proposal measures $Q^n := Q_\ell(u^n_{\ell,\ell}, \cdot), R^n := R_{\ell-1}(u^n_{\ell,\ell-1}, \cdot)$ in $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$. We aim at coupling proposals Q^n and R^n whenever these proposals are of the general form

$$\begin{aligned} u'_{\ell,\ell} &\sim \mathcal{N}(m(u^n_{\ell,\ell}), \tilde{\mathcal{C}}) = Q^n, \\ u'_{\ell,\ell-1} &\sim \mathcal{N}(m(u^n_{\ell,\ell-1}), \tilde{\mathcal{C}}) = R^n, \end{aligned} \tag{6.2}$$

where $m : \mathsf{X} \to \mathsf{X}$, is some $\mathcal{B}(\mathsf{X})$-measurable function and $\tilde{\mathcal{C}}$ is some symmetric, positive-definite covariance matrix in $\mathbb{R}^{K\times K}$. Proposals of this form are commonly used in MCMC; for some $z \in \mathsf{X}$, one could have, e.g., $m(z) = z$, if the corresponding proposal scheme corresponds to a RWM, or in the case where $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$, one can set $m(z) = \sqrt{1-\rho^2}z$, with some $\rho \in (0,1)$ and $\tilde{\mathcal{C}} = \rho^2\mathcal{C}$ if one is using pCN proposals instead. Let $\varphi_0 = \mathcal{N}(0, I)$, and with a slight abuse of notation, denote by $\varphi_0 : \mathsf{X} \to \mathbb{R}$ its Lebesgue density. Clearly, one can generate coupled samples with marginals (6.2) by sampling $(\xi, \zeta) \sim \gamma'$, where γ' is a coupling of φ_0 with φ_0, and setting

$$u'_{\ell,\ell} = m(u^n_{\ell,\ell}) + \tilde{\mathcal{C}}^{1/2}\xi, \quad u'_{\ell,\ell-1} = m(u^n_{\ell,\ell-1}) + \tilde{\mathcal{C}}^{1/2}\zeta, \quad \xi, \zeta \sim \mathcal{N}(0, I).$$

Thus, by carefully choosing how ξ, ζ are generated, one can generate maximally coupled proposals $(u'_{\ell,\ell-1}, u'_{\ell,\ell})$ with the desired distributions; indeed, one could (trivially) generate a maximal coupling of φ_0 with itself by sampling $\xi \sim \varphi_0$, and then setting $\zeta = \xi$, which produces (ξ, ζ) as sample from a maximal coupling. However, this will be a maximal coupling of φ_0 with itself, but will not lead, in general, to a maximal coupling of Q^n, R^n. To that end, for any $u^n_{\ell,\ell-1}, u^n_{\ell,\ell} \in \mathsf{X}$, let $z^n := \tilde{\mathcal{C}}^{-1/2}(m(u^n_{\ell,\ell}) - m(u^n_{\ell,\ell-1}))$ and define

$$e^n := \begin{cases} z^n / \|z^n\|_{\mathsf{X}} & \text{if } z^n \neq 0, \\ 0 & \text{otherwise.} \end{cases} \tag{6.3}$$

The reflection maximal coupling algorithm proceeds by first sampling $\xi \sim \varphi_0$ and then setting

$$\zeta = \begin{cases} \xi + z^n, & \text{with probability } \min\left\{1, \frac{\varphi_0(\xi+z^n)}{\varphi_0(\xi)}\right\} \quad \textbf{(case I)}, \\ \xi - 2\langle e^n, \xi\rangle_{\mathsf{X}} e^n, & \text{otherwise} \quad \textbf{(case II)}. \end{cases}$$

Thus, intuitively, if $z^n \approx 0$, meaning that $m(u^n_{\ell,\ell}), m(u^n_{\ell,\ell-1})$ are *relatively* close to each other, then, with high probability, $\zeta = \xi + z^n$, and as such, $u'_{\ell,\ell-1} = u'_{\ell,\ell}$. Otherwise, the algorithm produces ζ which is a reflection of ξ with respect to the plane orthogonal to e^n defined in (6.3). Theorem 6.2.1 states that Algorithm 11 samples $(u'_{\ell,\ell-1}, u'_{\ell,\ell})$ from a maximal coupling of Q^n, R^n.

Algorithm 11 Reflection maximal coupling.

procedure REFLECTION-COUPLING($\varphi_0, m(u^n_{\ell,\ell-1}), m(u^n_{\ell,\ell})$)
  Set $z^n = m(u^n_{\ell,\ell}) - m(u^n_{\ell,\ell-1})$ and set e^n from (6.3).
  Sample $\xi \sim \varphi_0$, and $w \sim \mathcal{U}([0,1])$.
  if $w \leq \min\left\{1, \frac{\varphi_0(\xi+z^n)}{\varphi_0(\xi)}\right\}$ **then**
    Set $\zeta = \xi + z^n$. ▷ case I
  else
    Set $\zeta = \xi - 2\langle e^n, \xi\rangle_{\mathsf{X}} e^n$. ▷ case II
  end if
  Set $u'_{\ell,\ell} = m(u^n_{\ell,\ell}) + \tilde{\mathcal{C}}^{1/2}\xi$ and $u'_{\ell,\ell-1} = m(u^n_{\ell,\ell-1}) + \tilde{\mathcal{C}}^{1/2}\zeta$
  Output $(u'_{\ell,\ell-1}, u'_{\ell,\ell})$.
end procedure

Theorem 6.2.1: *Let $u^n_{\ell,\ell-1}, u^n_{\ell,\ell} \in \mathsf{X}$. Algorithm 11 produces a coupled sample $(u'_{\ell,\ell-1}, u'_{\ell,\ell}) \sim \gamma^n$ where γ^n is a maximal coupling of $Q^n = \mathcal{N}(m(u^n_{\ell,\ell}), \tilde{\mathcal{C}})$ and $R^n = \mathcal{N}(m(u^n_{\ell,\ell-1}), \tilde{\mathcal{C}})$, i.e., $\mathbb{P}_{\gamma^n}(u'_{\ell,\ell} \neq u'_{\ell,\ell-1}) = \|R^n - Q^n\|_{\text{tv}}$, with $u'_{\ell,\ell} \sim Q^n$ and $u'_{\ell,\ell-1} \sim R^n$.*

Proof. See [76]. □

We reiterate that this coupling is induced by the coupling between the spherically symmetric measure φ_0 with itself. This technique can only be used to couple spherically symmetric proposals, such as (but not limited to) Gaussians [76]. Many commonly-used MCMC algorithms for PDE-based BIPs utilize proposals that arise from a spherically symmetric measure . Thus, we will primarily focus on this type of coupling for the work presented herein. An additional coupling technique based on rejection sampling is presented in [76, 159], but we will not investigate it in this work, since we believe it is less efficient than the reflection coupling in the ML-MCMC context.

6.2.2 Generating coupled chains

Let γ^n be the reflection maximal coupling of Q^n, R^n induced by Algorithm 11. Once such a coupling has been constructed, one can use γ^n as a proposal in a Metropolis-Hastings algorithm with marginals $\mu^y_{\ell-1}, \mu^y_\ell$, with $\ell \in \{0, 1, 2, \ldots, \mathsf{L}\}$. The procedure is relatively straight forward; given a joint state $(u^n_{\ell,\ell-1}, u^n_{\ell,\ell}) =: \boldsymbol{u}_\ell{}^n$ and (possibly state-dependent) marginal proposals probability measures Q^n and R^n, the algorithm begins by generating a joint proposal $\boldsymbol{u}_\ell{}' := (u'_{\ell,\ell-1}, u'_{\ell,\ell}) \sim \gamma^n = \gamma^n$ (where γ^n can possibly depend on the joint state $\boldsymbol{u}_\ell$, i.e., $\gamma^n(\boldsymbol{u}_\ell, \cdot)$) together with a random number $w \sim \mathcal{U}([0,1])$. and it then proceeds to accept or reject $u'_{\ell,\ell-1}$ and $u'_{\ell,\ell}$ as the new states of the respective marginal chains, following the usual MH accept-reject rule where, for each marginal chain, the MH acceptance probability $\alpha_\ell(u^n_{\ell,\ell}, u'_{\ell,\ell})$ and $\alpha_{\ell-1}(u^n_{\ell,\ell-1}, u'_{\ell,\ell-1})$ are compared to the same random number w. Furthermore, since γ^n is

a maximal coupling of Q^n, R^n, meaning, in particular, that Q^n and R^n are the marginals of γ^n, the MH acceptance rate $\alpha_j, j = \ell, \ell-1$, is of the form

$$\alpha_{\ell-1}(u_{\ell,\ell-1}, u'_{\ell,\ell-1}) = \min\left\{1, \frac{\pi^y_{\ell-1}(u'_{\ell,\ell-1})r(u'_{\ell,\ell-1}, u^n_{\ell,\ell-1})}{\pi^y_{\ell-1}(u^n_{\ell,\ell-1})r(u^n_{\ell,\ell-1}, u'_{\ell,\ell-1})}\right\}, \tag{6.4}$$

$$\alpha_{\ell}(u_{\ell,\ell}, u'_{\ell,\ell}) = \min\left\{1, \frac{\pi^y_{\ell}(u'_{\ell,\ell})q(u'_{\ell,\ell}, u^n_{\ell,\ell})}{\pi^y_{\ell}(u^n_{\ell,\ell})q(u^n_{\ell,\ell}, u'_{\ell,\ell})}\right\}, \tag{6.5}$$

where $\pi^y_j : \mathsf{X} \to \mathbb{R}_+, j = \ell-1, \ell$ are the densities of the posterior at level j with respect to some suitable reference measure (e.g., the prior), and similarly, $r : \mathsf{X}^2 \to \mathbb{R}_+$ and $q : \mathsf{X}^2 \to \mathbb{R}_+$ are the densities of the proposal measures R^n, Q^n with respect to some suitable reference probability measure (c.f. Section 3.4.1). This procedure is depicted in Algorithm 12. Once again (c.f. Chapter 5), we emphasize that such an algorithm also couples the Metropolisation step by comparing the acceptance probabilities α_j, $j = \ell-1, \ell$, with respect to the same uniform random number w. It is important to remark that, even though we use a maximal coupling as a proposal, the resulting joint Markov chain has marginals that are not, in general, maximally coupled, because of the Metropolization step.

Algorithm 12 Coupled chain MCMC.

procedure COUPLED-CHAIN-MCMC($\mu^y_\ell, \mu^y_{\ell-1}, \boldsymbol{u}_\ell{}^n, R^n, Q^n, m, \varphi$)
  # Produces one sample $\boldsymbol{u}_\ell{}^{n+1} = (u^n_{\ell,\ell-1}, u^n_{\ell,\ell}) \sim \nu_\ell$, with $u^{n+1}_{\ell,\ell} \sim \mu^y_\ell$ and $u^{n+1}_{\ell,\ell-1} \sim \mu^y_{\ell-1}$, given some current state $\boldsymbol{u}_\ell{}^n = (u^n_{\ell,\ell-1}, u^n_{\ell,\ell})$.
  Sample $\boldsymbol{u}_\ell{}'$ =`reflection-coupling`$(\varphi_0, m(u^n_{\ell,\ell-1}), m(u^n_{\ell,\ell}))$.
  Sample $w \sim \mathcal{U}([0,1])$
  Compute $\alpha_{\ell-1}(u^n_{\ell,\ell-1}, u'_{\ell,\ell-1})$, and $\alpha_\ell(u^n_{\ell,\ell}, u'_{\ell,\ell})$, as in Equations (6.4), (6.5).
  if $w \le \alpha_\ell(u^n_{\ell,\ell}, u'_{\ell,\ell})$ **then**
    Set $u^{n+1}_{\ell,\ell} = u'_{\ell,\ell}$.
  else
    Set $u^{n+1}_{\ell,\ell} = u^n_{\ell,\ell}$.
  end if
  if $w \le \alpha_{\ell-1}(u^n_{\ell,\ell-1}, u'_{\ell,\ell-1})$ **then**
    Set $u^{n+1}_{\ell,\ell-1} = u'_{\ell,\ell-1}$.
  else
    Set $u^{n+1}_{\ell,\ell-1} = u^n_{\ell,\ell-1}$.
  end if
  Output $\boldsymbol{u}_\ell{}^{n+1} = (u^{n+1}_{\ell,\ell}, u^{n+1}_{\ell,\ell-1})$.
end procedure

For any $A \in \mathcal{B}(\mathsf{X}^2)$, Algorithm 12 induces a Markov transition kernel $\boldsymbol{p}_\ell : \mathsf{X}^2 \times \mathcal{B}(\mathsf{X}^2) \to [0,1]$ given by

$$
\begin{aligned}
&\boldsymbol{p}_\ell(\boldsymbol{u}_\ell{}^n, A) \\
&= \int_{\mathsf{X}^2} \min\{\alpha_\ell(u^n_{\ell,\ell}, u'_{\ell,\ell}), \alpha_{\ell-1}(u^n_{\ell,\ell-1}, u'_{\ell,\ell-1})\} \mathbb{1}_{\{\boldsymbol{u}'_\ell \in A\}} \gamma^n(\boldsymbol{u}^n_\ell, \mathrm{d}\boldsymbol{u}_\ell{}') \\
&+ \int_{\mathsf{X}^2} \left(\alpha_\ell(u^n_{\ell,\ell}, u'_{\ell,\ell}) - \alpha_{\ell-1}(u^n_{\ell,\ell-1}, u'_{\ell,\ell-1})\right)^+ \mathbb{1}_{\left\{(u'_{\ell,\ell}, u^n_{\ell,\ell-1}) \in A\right\}} \gamma^n(\boldsymbol{u}^n_\ell, \mathrm{d}\boldsymbol{u}_\ell{}') \\
&+ \int_{\mathsf{X}^2} \left(\alpha_{\ell-1}(u^n_{\ell,\ell-1}, u'_{\ell,\ell-1}) - \alpha_\ell(u^n_{\ell,\ell}, u'_{\ell,\ell})\right)^+ \mathbb{1}_{\left\{(u^n_{\ell,\ell}, u'_{\ell,\ell-1}) \in A\right\}} \gamma^n(\boldsymbol{u}^n_\ell, \mathrm{d}\boldsymbol{u}_\ell{}') \\
&+ \mathbb{1}_{\{\boldsymbol{u}_\ell{}^n \in A\}} \left(1 - \int_{\mathsf{X}^2} \max\{\alpha_\ell(u^n_{\ell,\ell}, u'_{\ell,\ell}), \alpha_{\ell-1}(u^n_{\ell,\ell-1}, u'_{\ell,\ell-1})\} \gamma^n(\boldsymbol{u}^n_\ell, \mathrm{d}\boldsymbol{u}_\ell{}')\right),
\end{aligned}
\tag{6.6}
$$

where we have used $(b)^+ = \frac{b+|b|}{2}$, $\forall b \in \mathbb{R}$. In words, (6.6) gives the probability of moving to a set A given a joint state $\boldsymbol{u}^n_\ell$. We associate a Markov transition operator $\boldsymbol{P}_\ell : L_2(\mathsf{X}, \nu_\ell) \to L_2(\mathsf{X}, \nu_\ell)$ to $\boldsymbol{p}_\ell$, where ν_ℓ is the invariant probability measure of the Algorithm, provided it exists.

6.2.3 Re-synchronizing the chains

So far, we have proposed a method to generate coupled MCMC chains by sampling from a maximal coupling of the proposals of each marginal chain. However, it is still possible for the chains to uncouple and stay de-synchronized for a long period of time. In the setting for which one constructs each marginal chain using localized Gaussian proposals (such as RWM and pCN), where each proposal is a Gaussian centered at some $m(u_{\ell,\ell}), m(u_{\ell,\ell-1}) \in \mathsf{X}$, such a prolonged stage of de-synchronization is likely to occur when $m(u_{\ell,\ell})$ and $m(u_{\ell,\ell-1})$ are sufficiently *far apart*. This is a situation that could potentially happen whenever the posterior is e.g., multi-modal or high-dimensional, as suggested by the numerical experiments in [76]. Ideally, we would like for the chains generated by our algorithm to avoid such a situation, since long periods of de-synchronizations will, in general, reduce the correlation between chains at level ℓ and $\ell - 1$, and could eventually result in having $\mathbb{V}_{\nu_\ell}[Y_\ell] = O(1)$. One possible way to avoid this undesirable situation is to construct the coupled chains at level ℓ, using the following convex combination of Markov transition operators:

$$
\hat{\boldsymbol{P}}_\ell := (1 - \omega_\ell)\boldsymbol{P}_\ell + \omega_\ell \boldsymbol{P}_\ell{}^{\mathsf{sync}}, \quad \omega_\ell \in (0,1),\ \ell = 0, 1, \ldots, \mathsf{L}.
$$

Here, $\boldsymbol{P}_\ell$ denotes one step of Algorithm 12 and $\boldsymbol{P}_\ell{}^{\mathsf{sync}}$ is a synchronization Markov transition operator, which at each step, proposes to both chains a common candidate state, which then gets accepted or rejected by each marginal chain with the usual Metropolisation step. Thus, for some fixed ω_ℓ, $\hat{\boldsymbol{P}}_\ell$ can be understood as sampling from Algorithm 12 with probability $1 - \omega_\ell$, and otherwise sampling from $\boldsymbol{P}_\ell{}^{\mathsf{sync}}$. This synchronization operator can be understood as a one step of IMH as in Chapter 5, proposing a candidate state from, e.g., the prior μ_{pr} or a KDE of the

previous samples, a sub-sampled approximation of $\mu^y_{\ell-1}$ as in [45], or a sample from μ^y_ℓ or $\mu^y_{\ell-1}$ obtained from a chain with invariant measure μ^y_ℓ or $\mu^y_{\ell-1}$ run independently and in parallel. One could, alternatively, propose to synchronize the chains (i.e., attempt to have $u^{n+1}_{\ell,\ell} = u^{n+1}_{\ell,\ell-1}$ with some probability) by performing one full iteration of the multi-level delayed-acceptance algorithm presented in [105].
This re-synchronization procedure yields the following alternative one-step re-synchronized coupled chain MCMC algorithm which can also be used inside Algorithm 8.

Algorithm 13 Resync. Coupled-chain-MCMC.

procedure RESYNC. COUPLED CHAIN MCMC($\mathbf{p}_\ell, \mathbf{p}^{\mathsf{sync}}_\ell, (u^n_{\ell,\ell-1}, u^n_{\ell,\ell}), \omega_\ell$)
    # Produces one sample $\boldsymbol{u}_\ell{}^{n+1} = (u^{n+1}_{\ell,\ell-1}, u^{n+1}_{\ell,\ell}) \sim \nu_\ell$, with $u^{n+1}_{\ell,\ell} \sim \mu^y_\ell$ and $u^{n+1}_{\ell,\ell-1} \sim \mu^y_{\ell-1}$, given some current state $\boldsymbol{u}_\ell{}^n = (u^n_{\ell,\ell-1}, u^n_{\ell,\ell})$.
    Sample $U \sim \mathcal{U}(0,1)$
    if $U \geq \omega_\ell$ **then**
        Sample $\boldsymbol{u}_\ell{}^{n+1} \sim \mathbf{p}_\ell(\boldsymbol{u}_\ell{}^n, \cdot)$
    else
        Sample $\boldsymbol{u}_\ell{}^{n+1} \sim \mathbf{p}^{\mathsf{sync}}_\ell(\boldsymbol{u}_\ell{}^n, \cdot)$, i.e., do one step of IMH as described above.
    end if
    Output $\boldsymbol{u}_\ell{}^{n+1} = (u^{n+1}_{\ell,\ell-1}, u^{n+1}_{\ell,\ell})$.
end procedure

6.3 Convergence of the level-wise coupled pCN chain

We now proceed to analyze the convergence of the level-wise coupled chains generated by Algorithm 12. The main result of this section is stated in Theorem 6.3.1, which provides conditions for the existence and uniqueness of an invariant measure ν_ℓ. We will limit ourselves to the particular case where $\mathsf{X} = \mathbb{R}^K$, $\mu_{\mathsf{pr}} = \mathcal{N}(0, \mathcal{C})$ and $Q_\ell(u, \cdot) = \mathcal{N}(\sqrt{1-\rho^2}u, \rho^2\mathcal{C})$. Here $K \geq 1$ is independent of the level, $\mathcal{C}$ is a symmetric positive definite matrix in $\mathbb{R}^{K\times K}$ and $\rho \in (0,1)$. This setting corresponds to a pCN algorithm in a finite-dimensional space. In this case, the acceptance rate for the marginal chain at level ℓ is of the form $\alpha_\ell(u, u'_{\ell,\ell}) = \min\{1, e^{\Phi_\ell(u;y) - \Phi_\ell(u'_{\ell,\ell};y)}\}$, $\forall u, u'_{\ell,\ell} \in \mathsf{X}$. We believe that our setting can be easily extended to other MCMC algorithms, however, we choose not to pursue such an analysis in this work.
Similarly as in the previous chapter, our goal is then to show that, under some technical assumptions, our ML-MCMC algorithm satisfies the assumptions of Theorem 3.2.4. To that end, we will require the following assumptions to hold:

Assumption 6.3.1 (Assumptions on the potential): *The following conditions hold for all* $\ell = 1, \ldots, \mathscr{L}$*:*

6.3.1.1. $\Phi_\ell(u; y)$ *is strictly positive* $\forall u \in \mathsf{X}$, $y \in \mathsf{Y}$.

6.3.1.2. *There exists* $\mathscr{R} > 0$, $\mathscr{A}_{\lambda_\ell} > -\infty$ *and a function* $r : \mathbb{R}^+ \to \mathbb{R}^+$ *satisfying* $r(s) = \hat{r}s^a$ *for all* $|s| \geq \mathscr{R}$, $\hat{r} > 0$, $a \in (\frac{1}{2}, 1)$, *such that for all* $u \in B_{\mathscr{R}}(0)^c$, *it holds that*

$$\inf_{u'_{\ell,\ell} \in B_{r(\|u\|)}(\sqrt{1-\rho^2}u)} \alpha_\ell(u, u'_{\ell,\ell}) > \exp(\mathscr{A}_{\lambda_\ell}).$$

Assumption 6.3.1.1 is mild and easy to satisfy. Assumption 6.3.1.2 is more complicated to verify, however, it is needed to establish the convergence of the marginal pCN algorithm (see [65]). We now present the main result of this section.

Theorem 6.3.1: (Existence of a unique invariant measure and geometric ergodicity) *Suppose that Assumption 6.3.1 holds. Then, for any level* $\ell = 1, 2, \ldots, \mathsf{L}$,

1. *The joint Markov chain generated by* $\boldsymbol{P}_\ell$ *in* (6.6) *has a unique stationary probability measure* ν_ℓ *on* $(\mathsf{X}^2, \mathcal{B}(\mathsf{X}^2))$.

2. *The joint Markov chain generated by* $\boldsymbol{P}_\ell$ *is geometrically ergodic. That is, there exists a* ν_ℓ*-integrable, bi-variate Lyapunov function* $V_{\ell-1,\ell} : \mathsf{X}^2 \mapsto [1, \infty]$, *an* $r \in (0, 1)$ *and* $M \in \mathbb{R}^+$, *such that*

$$\sup_{|f| \leq V_{\ell-1,\ell}} \left| \int_{\mathsf{X}^2} f(\boldsymbol{u}_\ell{}') \boldsymbol{p}^n_\ell(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell{}') - \int_{\mathsf{X}^2} f(\boldsymbol{u}_\ell) \nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \right| \leq M V_{\ell-1,\ell}(\boldsymbol{u}_\ell) r^n,$$

   $\forall \boldsymbol{u}_\ell \in \mathsf{X}^2$, $n \in \mathbb{N}$, *where the supremum is taken over all* ν_ℓ*-measurable functions* $f : \mathsf{X}^2 \mapsto \mathbb{R}$ *satisfying* $|f(\boldsymbol{u}_\ell)| \leq V_{\ell-1,\ell}(\boldsymbol{u}_\ell)$.

We postpone the proof of Theorem 6.3.1 to the end of the this subsection. We begin by showing the irreducibly of the chain generated by (6.6). For notational simplicity, for the remainder of this subsection we will write $\boldsymbol{u}_\ell = (u_{\ell,\ell-1}, u_{\ell,\ell})$, $\boldsymbol{u}_\ell{}' = (u'_{\ell,\ell-1}, u'_{\ell,\ell})$, and $q(u_{\ell,\ell}, u'_{\ell,\ell})$ and $r(u_{\ell,\ell-1}, u'_{\ell,\ell-1})$ as the Lebesgue densities of $Q_\ell(u_{\ell,\ell}, \cdot)$, $R_{\ell-1}(u_{\ell,\ell-1}, \cdot)$ respectively, evaluated at $u'_{\ell,\ell-1}, u'_{\ell,\ell} \in \mathsf{X}$. Furthermore, we denote by $\gamma(\cdot) = \gamma(\boldsymbol{u}_\ell, \cdot)$ the maximal coupling of $Q_\ell(u_{\ell,\ell}, \cdot)$ and $R_{\ell-1}(u_{\ell,\ell-1}, \cdot)$ obtained with Algorithm 11.

Lemma 6.3.1: (Irreducibility) *Suppose Assumption 6.3.1 holds. Then, for any* $\ell = 1, 2, \ldots, \mathsf{L}$, *the joint Markov transition kernel* $\boldsymbol{p}_\ell$ *defined in* (6.6) *is* ψ*-irreducible.*

Proof. Take any compact set $K \in \mathcal{B}(\mathsf{X})$ with non-zero Lebesgue measure, set $K^2 = K \times K$ and, for any set $A \in \mathcal{B}(\mathsf{X}^2)$, denote $A_K = A \cap K^2$. For $\boldsymbol{u}_\ell \in \mathsf{X}^2$, one has that

$$\begin{aligned} &\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, A) \geq \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, A_K) \\ &\geq \int_{\mathsf{X}^2} \min\{\alpha_\ell(u, u'_{\ell,\ell}), \alpha_{\ell-1}(v, u'_{\ell,\ell-1})\} \mathbb{1}_{\{\boldsymbol{u}_\ell{}' \in A_K\}} \gamma(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell{}'), \end{aligned} \tag{6.7}$$

Since we are using pCN kernels on both marginals, we obtain:

$$\begin{aligned}
&\min\{\alpha_\ell(u_{\ell,\ell}, u'_{\ell,\ell}), \alpha_{\ell-1}(u_{\ell,\ell-1}, u'_{\ell,\ell-1})\} \\
&= \min\left\{1, \frac{e^{-\Phi_\ell(u'_{\ell,\ell};y)}}{e^{-\Phi_\ell(u_{\ell,\ell};y)}}, \frac{e^{-\Phi_{\ell-1}(u'_{\ell,\ell-1};y)}}{e^{-\Phi_{\ell-1}(u_{\ell,\ell-1};y)}}\right\} \\
&\geq \min\left\{e^{-\Phi_\ell(u'_{\ell,\ell};y)}, e^{-\Phi_{\ell-1}(u'_{\ell,\ell-1};y)}\right\} =: \hat{\alpha}_\ell(\boldsymbol{u}_\ell{}'). \qquad (6.8)
\end{aligned}$$

Furthermore, since A_K is a compact set and $\Phi_j, j = \ell-1, \ell$ is a continuous and positive function, then, there exists $\delta_\ell > 0$ such that $\hat{\alpha}_\ell(\boldsymbol{u}_\ell{}') \geq \delta_\ell, \quad \forall \boldsymbol{u}_\ell{}' \in A_K$. Thus, we obtain

$$\begin{aligned}
(6.7) &\geq \delta_\ell \int_{\mathsf{X}^2} \mathbb{1}_{\{\boldsymbol{u}_\ell{}' \in A_K\}} \gamma(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell{}') \\
&= \delta_\ell \int_{\mathsf{X}^2 \cap \Delta} \mathbb{1}_{\{\boldsymbol{u}_\ell{}' \in A_K\}} \gamma(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell{}') + \int_{\mathsf{X}^2 \cap \Delta^c} \mathbb{1}_{\{\boldsymbol{u}_\ell{}' \in A_K\}} \gamma(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell{}') \\
&\geq \delta_\ell \int_{\Delta} \mathbb{1}_{\{\boldsymbol{u}_\ell{}' \in A_K\}} \gamma(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell{}'), \qquad (6.9)
\end{aligned}$$

where $\Delta = \{(u_{\ell,\ell-1}, u_{\ell,\ell}) \in \mathsf{X}^2 : \ u_{\ell,\ell-1} = u_{\ell,\ell}\}$. Notice that the integral in Equation (6.9) is over the diagonal set of X^2, i.e., over the set $\{\boldsymbol{u}_\ell{}' \in \mathsf{X}^2 : u_{\ell,\ell-1} = u_{\ell,\ell}\}$, which can only occur when Algorithm 11 finalizes in case I. Thus, writing $u'_{\ell,\ell-1} = u'_{\ell,\ell} = u'$, and observing that since $u' = m(u_{\ell,\ell}) + \tilde{\mathcal{C}}^{1/2}\xi, \xi \sim \varphi_0$, we have

$$\xi = \tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell})), \quad \xi + \tilde{\mathcal{C}}^{-1/2}(m(u_{\ell,\ell}) - m(u_{\ell,\ell-1}) = \tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell-1})),$$

it then follows that

$$\begin{aligned}
(6.9) &= \delta_\ell \int_{\mathsf{X}} \mathbb{1}_{\{(u'(\xi), u'(\xi)) \in A_K\}} \min\left\{\varphi_0(\xi), \varphi_0(\xi + \tilde{\mathcal{C}}^{-1/2}(m(u_{\ell,\ell}) - m(u_{\ell,\ell-1})))\right\} \mathrm{d}\xi \\
&= |\det \tilde{\mathcal{C}}^{-1/2}| \delta_\ell \int_{\mathsf{X}} \mathbb{1}_{\{(u', u') \in A_K\}} \min\left\{\varphi_0(\tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell}))), \varphi_0(\tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell-1})))\right\} \mathrm{d}u'.
\end{aligned}$$

Furthermore, since $\left\|\tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell}))\right\|_{\mathsf{X}}^2 \leq 2\left\|\tilde{\mathcal{C}}^{-1/2}u'\right\|_{\mathsf{X}}^2 + 2\left\|\tilde{\mathcal{C}}^{-1/2}m(u_{\ell,\ell})\right\|_{\mathsf{X}}^2$, it then follows from the previous equation that

$$(6.9) \geq |\det \tilde{\mathcal{C}}^{-1/2}| \min\{e^{-\|m(u_{\ell,\ell-1})\|_{\tilde{\mathcal{C}}}^2}, e^{-\|m(u_{\ell,\ell})\|_{\tilde{\mathcal{C}}}^2}\} \delta_\ell \int_{\mathsf{X}} \mathbb{1}_{\{(u', u') \in A_K\}} \varphi_0\left(\sqrt{2}\tilde{\mathcal{C}}^{-1/2}u'\right) \mathrm{d}u'$$

where $\|\cdot\|_{\tilde{\mathcal{C}}} = \left\|\tilde{\mathcal{C}}^{-1/2}\cdot\right\|_{\mathsf{X}}$. Setting

$$\psi(A) := \int_{\mathsf{X}} \mathbb{1}_{\{(u', u') \in A \cap K^2\}} \varphi_0\left(\sqrt{2}\tilde{\mathcal{C}}^{-1/2}u'\right) \mathrm{d}u'$$

gives the desired result. □

We now proceed to show that Assumption 6.3.1 implies that all compact subsets of X^2 are small sets.

Lemma 6.3.2: (Existence of small sets) *Let $\hat{S}_\ell \in \mathcal{B}(\mathsf{X}^2)$ be a compact subset and suppose* Assumption 6.3.1 *holds. Then, $\hat{S}_\ell$ is a small set for the Markov kernel $\boldsymbol{p}_\ell$.*

Proof. We proceed similarly to the proof of Lemma 6.3.1. Notice that for any $A \in \mathcal{B}(\mathsf{X}^2)$, it holds that

$$\begin{aligned}\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, A) &\geq \int_{\mathsf{X}^2} \min\{\alpha_\ell(u, u'_{\ell,\ell}), \alpha_{\ell-1}(v, u'_{\ell,\ell-1})\}\mathbb{1}_{\{\boldsymbol{u}_\ell' \in A\}}\gamma(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell') \\ &\geq \int_{\mathsf{X}^2} \hat{\alpha}_\ell(\boldsymbol{u}_\ell')\mathbb{1}_{\{\boldsymbol{u}_\ell' \in A\}}\gamma(\boldsymbol{u}_\ell, \mathrm{d}\boldsymbol{u}_\ell'),\end{aligned}$$

with $\hat{\alpha}_\ell(\boldsymbol{u}_\ell')$ as in (6.8). Minorizing once again by the probability of Algorithm 11 finishing on the first case (i.e., proposing the same state for both chains), we obtain:

$$\begin{aligned}&\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, A) \geq |\det \tilde{\mathcal{C}}^{-1/2}| \\ &\times \int_{\mathsf{X}} \hat{\alpha}_\ell(\boldsymbol{u}_\ell')\mathbb{1}_{\{(u',u')\in A\}} \min\left\{\varphi_0(\tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell}))), \varphi_0(\tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell-1})))\right\} \mathrm{d}u'.\end{aligned}$$

Moreover, since $\hat{S}_\ell$ is compact and $\varphi_0(\cdot)$ (when seen as a density) is a positive, continuous, and bounded function, then, there exists a continuous and bounded function $\hat{\delta} : \mathsf{X} \to \mathbb{R}_+$ such that

$$0 < \hat{\delta}'(u') \leq \min\left\{\varphi_0(\tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell}))), \varphi_0(\tilde{\mathcal{C}}^{-1/2}(u' - m(u_{\ell,\ell-1})))\right\} \quad \forall \boldsymbol{u}_\ell \in \hat{S}_\ell.$$

Thus, setting $\nu(A) := \int_{\mathsf{X}} \hat{\delta}'\hat{\alpha}_\ell(u')\mathbb{1}_{\{(u',u')\in A\}}\varphi_0\left(\sqrt{2}\tilde{\mathcal{C}}^{-1/2}u'\right)\mathrm{d}u'$ gives the desired result. □

Aperiodicity follows from Lemmata 6.3.1 and 6.3.2 since $\nu(\hat{S}_\ell) > 0$ as long as $\hat{S}_{\ell,\Delta} = \{u \in \mathsf{X} : (u, u) \in \hat{S}_\ell\}$ has non-zero φ_0-measure.
We now focus on the existence of a drift condition (c.f. Definition 3.2.8) for our ML-MCMC kernel. We recall the following auxiliary result from [65], which states the existence of such a drift condition for the marginal pCN algorithm.

Lemma 6.3.3: (convergence of the marginal pCN kernel) *Suppose Assumption* 6.3.1 *holds. Then, the pCN algorithm with invariant measure μ_ℓ^y is geometrically ergodic and satisfies the drift condition as in* (3.14) *with a Lyapunov function $V_\ell : \mathsf{X} \to [1, \infty)$ of the form $V_\ell(u_{\ell,\ell}) = \exp(k_\ell \|u_{\ell,\ell}\|_{\mathsf{X}})$, for some $k_\ell > 0$.*

Proof. See [65, Theorem 2.12] □

Remark 6.3.1: *The work [65] states the previous result in terms of Lyapunov function V satisfying the slightly different drift condition:*

$$(P_\ell V_\ell)(u_{\ell,\ell}) \leq \lambda'_\ell V_\ell(u_{\ell,\ell}) + K_\ell, \quad \lambda'_\ell \in (0,1),\ K_\ell < \infty,$$

i.e., without an explicit dependency on a small set. It can be shown, however, that the function V in the previous Lemma also satisfies our drift condition. To that end, set $S_\ell = \{u \in \mathsf{X} : V_\ell(u) \leq L\}$, for some L sufficiently large so that $\lambda_\ell := \lambda'_\ell + K_\ell/L < 1$. Notice that since V_ℓ has compact level sets and, by Lemma 6.3.2, compact sets are small, S_ℓ is then a small set. We then have that

$$\begin{aligned}
(P_\ell V_\ell)(u_{\ell,\ell}) &\leq \lambda'_\ell V_\ell(u_{\ell,\ell}) + K_\ell = \lambda'_\ell V_\ell + K_\ell \mathbb{1}_{\{u \in S_\ell\}} + K_\ell \mathbb{1}_{\{u \notin S_\ell\}} \\
&\leq \lambda'_\ell V_\ell + K_\ell \mathbb{1}_{\{u_{\ell,\ell} \in S_\ell\}} + \left(\frac{K_\ell}{L} \mathbb{1}_{\{u_{\ell,\ell} \notin S_\ell\}}\right) V_\ell(u_{\ell,\ell}) \\
&= \left(\lambda'_\ell + \frac{K_\ell}{L}\right) V_\ell(u_{\ell,\ell}) + K_\ell \mathbb{1}_{\{u_{\ell,\ell} \in S_\ell\}} \\
&= \lambda_\ell V_\ell(u_{\ell,\ell}) + K_\ell \mathbb{1}_{\{u_{\ell,\ell} \in S_\ell\}}.
\end{aligned}$$

Notice that from the previous theorem, it follows that, given some $\Gamma \in \mathbb{R}_+$, the set $\hat{S}_\ell = \{u_{\ell,\ell} \in \mathsf{X} : V_\ell(u_{\ell,\ell}) \leq \Gamma\}$ is compact (and hence, a small set, from Lemma 6.3.2). We will use this in the proof of Lemma 6.3.4. We define the joint function $V_{\ell-1,\ell} : \mathsf{X}^2 \to [1,\infty)$ by

$$V_{\ell-1,\ell}(\boldsymbol{u}_\ell) := \frac{1}{2}(V_\ell(u_{\ell,\ell}) + V_{\ell-1}(u_{\ell,\ell-1})). \tag{6.10}$$

The next Lemma shows that $V_{\ell-1,\ell}$ is a Lyapunov function for the joint kernel $\boldsymbol{p}_\ell$. Since the marginal kernels of $\boldsymbol{p}_\ell$ are p_ℓ and $p_{\ell-1}$ respectively, one then has, for $\tilde{V}_\ell(\boldsymbol{u}_\ell) = V_\ell(u_{\ell,\ell})$ that $\boldsymbol{P}_\ell \tilde{V}_\ell = P_\ell V_\ell = \int_{\mathsf{X}} V_\ell(u'_{\ell,\ell}) p_{\ell,\ell}(u_{\ell,\ell}, \mathrm{d}u'_{\ell,\ell})$ (where $p_{\ell,\ell}$ is the Markov transition kernel corresponding to the marginal chain with invariant measure μ_ℓ^y) and similarly $\boldsymbol{P}_\ell \tilde{V}_{\ell-1} = P_\ell V_{\ell-1}$, that is, the joint kernel acts on the marginal Lyapunov function exactly as the marginal kernel does. In light of this, we now show that our joint kernel $\boldsymbol{p}_\ell$ satisfies a drift condition of the form (3.14).

Lemma 6.3.4 (Drift condition): *Suppose Assumption 6.3.1 holds. Then, the kernel $\boldsymbol{p}_\ell$ in (6.6) satisfies a drift condition of the form (3.14) with Lyapunov function $V_{\ell-1,\ell}(\boldsymbol{u}_\ell) = \frac{1}{2}(V_\ell(u_{\ell,\ell}) + V_{\ell-1}(u_{\ell,\ell-1}))$.*

Proof. From Lemma 6.3.3, it follows that for $j = \ell - 1, \ell$, the marginal kernel $p_{\ell,j}$, satisfies a drift condition of the form

$$\int_X V_j(u'_{\ell,j}) p_{\ell,j}(u_{\ell,j}, \mathrm{d}u'_{\ell,j}) \leq \lambda_j V_j(u_{\ell,j}) + \kappa_j \mathbb{1}_{\left\{u_{\ell,j} \in \hat{S}_j\right\}},\ \forall u_{\ell,j} \in \mathsf{X},$$

for some $\lambda_j \in (0,1), \kappa_j \in (0,\infty)$ and for a small set $\hat{S}_j := \{u_{\ell,j} \in \mathsf{X} : V_j(u_{\ell,j}) \leq \Gamma\}, \hat{S}_j \subset \mathsf{X}$, with $\Gamma > 0$ sufficiently large such that $\max\{\lambda_\ell, \lambda_{\ell-1}\} + \frac{2\max\{\kappa_\ell, \kappa_{\ell-1}\}}{1+\Gamma} < 1$. Notice that this

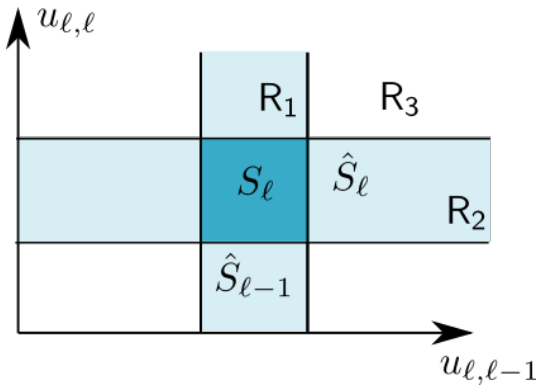

Figure 6.1: Depiction of a two-dimensional small set.

choice of small set is motivated since, by Lemma 6.3.3, the Lyapunov function $V_j : \mathsf{X} \to [1, \infty)$, $j = \ell - 1, \ell$, has compact level sets. Furthermore, define $\mathcal{S}_\ell := \hat{S}_\ell \times \hat{S}_{\ell-1}$. We remark that $\hat{S}_j$ (and hence $\mathcal{S}_\ell$) is compact since the function V_j given in Lemma 6.3.3 has compact level sets, and as such $\mathcal{S}_\ell$ is a small set. We will now show that $\boldsymbol{P}_\ell$ satisfies a drift condition with Lyapunov function given by (6.10) and small set $\mathcal{S}_\ell$. Consider first the case where $\boldsymbol{u}_\ell \in \mathcal{S}_\ell^c$. Notice that the set $\mathcal{S}_\ell^c$ can be written as the union of three non-overlapping regions; $\mathcal{S}_\ell^c = \mathsf{R}_1 \cup \mathsf{R}_2 \cup \mathsf{R}_3$, where $\mathsf{R}_1 = \{\boldsymbol{u}_\ell \in \mathsf{X}^2 : u_{\ell,\ell} \notin \hat{S}_\ell, u_{\ell,\ell-1} \in \hat{S}_{\ell-1}\}$, $\mathsf{R}_2 = \{\boldsymbol{u}_\ell \in \mathsf{X}^2 : u_{\ell,\ell} \in \hat{S}_\ell, u_{\ell,\ell-1} \notin \hat{S}_{\ell-1}\}$, and $\mathsf{R}_3 = \mathcal{S}_\ell^c \backslash (\mathsf{R}_1 \cup \mathsf{R}_2)$, as depicted in Figure 6.1.
For $\boldsymbol{u}_\ell \in \mathsf{R}_1$ we have that

$$(\boldsymbol{P}_\ell V_{\ell-1,\ell})(\boldsymbol{u}_\ell) \leq \frac{1}{2}\left(\lambda_\ell V_\ell(u_{\ell,\ell}) + \lambda_{\ell-1} V_{\ell-1}(u_{\ell,\ell-1})\right) + \frac{\kappa_{\ell-1}}{2}. \tag{6.11}$$

Since $\boldsymbol{u}_\ell \in \mathsf{R}_1$, it then holds that $V_\ell \geq \Gamma$. Furthermore, since $V_j \geq 1$, we then have that $V_{\ell-1,\ell}(\boldsymbol{u}_\ell) = \frac{1}{2}\left(V_\ell(u_{\ell,\ell}) + V_{\ell-1}(u_{\ell,\ell-1})\right) \geq \frac{1}{2}(1+\Gamma)$, which in turn implies that $\frac{1}{2} \leq \frac{V_{\ell-1,\ell}(\boldsymbol{u}_\ell)}{1+\Gamma}$. Thus, from (6.11), one obtains that

$$\begin{aligned}(6.11) &\leq \frac{1}{2}\left(\lambda_\ell V_\ell(u_{\ell,\ell}) + \lambda_{\ell-1} V_{\ell-1}(u_{\ell,\ell-1})\right) + \frac{\kappa_{\ell-1}}{1+\Gamma} V_{\ell-1,\ell}(\boldsymbol{u}_\ell) \\ &\leq \underbrace{\left(\max\{\lambda_\ell, \lambda_{\ell-1}\} + \frac{\kappa_{\ell-1}}{1+\Gamma}\right)}_{<1} V_{\ell-1,\ell}(\boldsymbol{u}_\ell).\end{aligned} \tag{6.12}$$

Similarly, for $\boldsymbol{u}_\ell \in \mathsf{R}_2$ it holds that

$$\boldsymbol{P}_\ell V_{\ell-1,\ell}(\boldsymbol{u}_\ell) \leq \underbrace{\left(\max\{\lambda_\ell, \lambda_{\ell-1}\} + \frac{\kappa_\ell}{1+\Gamma}\right)}_{<1} V_{\ell-1,\ell}(\boldsymbol{u}_\ell), \tag{6.13}$$

and for $\boldsymbol{u}_\ell \in \mathsf{R}_3$ we simply have:

$$\boldsymbol{P}_\ell V_{\ell-1,\ell}(\boldsymbol{u}_\ell) \leq (\max\{\lambda_\ell, \lambda_{\ell-1}\})\, V_{\ell-1,\ell}(\boldsymbol{u}_\ell). \tag{6.14}$$

Thus, from (6.12), (6.13), and (6.14) it follows that

$$\boldsymbol{P}_\ell V_{\ell-1,\ell}(\boldsymbol{u}_\ell) \leq \underbrace{\left(\max\{\lambda_\ell, \lambda_{\ell-1}\} + \frac{\max\{\kappa_\ell, \kappa_{\ell-1}\}}{1+\Gamma}\right)}_{=:\,\Lambda_\ell < 1} V_{\ell-1,\ell}(\boldsymbol{u}_\ell), \quad \forall \boldsymbol{u}_\ell \notin \mathcal{S}_\ell. \tag{6.15}$$

Lastly, for $\boldsymbol{u}_\ell \in \mathcal{S}_\ell$ we have

$$\begin{aligned}\boldsymbol{P}_\ell V_{\ell-1,\ell}(\boldsymbol{u}_\ell) &\leq \frac{1}{2}\left(\lambda_\ell V_\ell(u_{\ell,\ell}) + \lambda_{\ell-1} V_{\ell-1}(u_{\ell,\ell-1})\right) + \frac{\kappa_{\ell-1}}{2} + \frac{\kappa_\ell}{2} \\ &\leq \Lambda_\ell V_{\ell-1,\ell}(\boldsymbol{u}_\ell) + \hat{\kappa}_\ell,\end{aligned} \tag{6.16}$$

with $\hat{\kappa}_\ell := \frac{1}{2}(\kappa_{\ell-1} + \kappa_\ell)$. Thus, from (6.15) and (6.16), we have that the joint kernel satisfies a drift condition of the form (3.14), namely:

$$\boldsymbol{P}_\ell V_{\ell-1,\ell}(\boldsymbol{u}_\ell) \leq \Lambda_\ell V_{\ell-1,\ell}(\boldsymbol{u}_\ell) + \hat{\kappa}_\ell \mathbb{1}_{\{\boldsymbol{u}_\ell \in \mathcal{S}_\ell\}}.$$

□

We now have all the required results to prove Theorem 6.3.1.

Proof of Theorem 6.3.1. The proof of Theorem 6.3.1 follows immediately from Theorem 3.2.4 and Lemmata 6.3.1, 6.3.2, and 6.3.4. □

Corollary 6.3.1: *Under the same assumptions as in Theorem 6.3.1, the ML-MCMC algorithm induced by a maximal coupling of the* gpCN *method of [144] also has a unique invariant joint measure; this follows form the fact that both pCN and the gpCN samplers have the same MH acceptance probability* $\alpha(u, v)$.

Remark 6.3.2 (On the proof of the complexity result of [45]): *We remark that we are currently unable to prove that conditions T2 and T3 of Theorem 5.4.1 in Chapter 5 for the complexity result of Dodwell et. al. ([45, Theorem 3.5]) hold true for the currently proposed method, under reasonable assumptions (one can show that T1 holds true under similar conditions to that of Lemma 5.4.3, however, this condition alone is not sufficient for the complexity result of Dodwell. et. al. to hold true). We expect however, that, by including a re-synchronization kernel, as in Algorithm 13 the convergence properties of the ML-MCMC algorithm are "inherited" from the IMH part. Furthermore, the numerical results in the following sections suggest that T2 holds, indeed, true (with and without re-synchronization). Furthermore, it is a consequence of Theorem 5.3.4 that T3 also holds true provided that the (joint) chain is mixing sufficiently fast. We investigate this in further detail in the Appendix of this chapter.*

6.4 Numerical experiments

6.4.1 Moving Gaussians, revisited

We return to the *sanity check* experiment of Section 5.6.2 to verify the capabilities of our proposed method. In this case, we aim at sampling from the family of probability measures $\mu_\ell^y = \mathcal{N}\left(2^{-\ell+2}, 1\right), \ell = 0, 1, 2, \ldots, \mathsf{L}$, which approximate $\mu^y = \mathcal{N}(0, 1)$ as $\ell \to \infty$. As discussed in Chapter 5, such a family of probability measures poses a problem to some ML-MCMC algorithms due to the relatively small overlap between the posterior measure at level $\ell = 0$ and those at higher accuracy levels. In particular, we aim at comparing our method with the sub-sampling algorithm of [45]. To that end, we first implement our ML-MCMC Algorithm 8 from Chapter 5 together with Algorithm 12 (that is, we are not using a re-synchronization kernel). At any given level ℓ, the proposed coupled state at the $(n+1)^{\text{th}}$ step is given by $(u'_{\ell,\ell-1}, u'_\ell) \sim \gamma_\ell^n$, where $u'_{\ell,\ell-1} \sim \mathcal{N}(u^n_{\ell,\ell-1}, \sigma^2), u'_{\ell,\ell} \sim \mathcal{N}(u^n_{\ell,\ell}, \sigma^2)$, and $\mathbb{P}(u'_{\ell,\ell-1} \neq u'_\ell) = \left\|\mathcal{N}(u^n_{\ell,\ell-1}, \sigma^2) - \mathcal{N}(u^n_{\ell,\ell}, \sigma^2)\right\|_{\text{tv}}$. Here, $(u^n_{\ell,\ell-1}, u^n_{\ell,\ell})$ denotes the current state of each chain with invariant measure $\mu^y_{\ell-1}, \mu^y_\ell$ respectively. At each level, the step-size of the RWM algorithm, $\sigma^2 = 1$ is chosen such that each chain has an acceptance rate of about 40%. We compared our proposed approach to the methods and experimental setting of Section 5.6.2, i.e., (a) the sub-sampling ML-MCMC algorithm of [45] with a level-independent sub-sampling rate $t_\ell = \max\left\{1 + 2\sum_{k=0}^N \hat{\varrho}_\ell(k), 5\right\}$, where $1 + 2\sum_{k=0}^N \varrho_\ell(k)$ is the integrated auto-correlation time of $Y_\ell(\boldsymbol{u}_\ell)$ at level ℓ, and (b) our IMH algorithm with a level-independent proposal $Q_\ell = Q = \mathcal{N}(2, 3)$. For all methods, the proposal distribution at level $\ell = 0$ is a random walk Metropolis proposal $Q_0(u_0^n, \cdot) = \mathcal{N}(u_0^n, 1)$, which yields an acceptance rate of about 40%.

We begin by investigating the correctness of the corresponding marginals obtained with our method, and compare such results to those obtained with the the previously discussed ML-MCMC algorithms. To that end, we run all algorithms with $\mathsf{L} = 7$, obtaining $20,000$ samples per level, and present the histograms of the samples obtained with all methods at levels $\ell = 2, 4, 7$ in Figure 6.2. The left column of Figure 6.2 shows the histograms of the samples from $\mu^y_{\ell-1}, \mu^y_\ell$ obtained with the maximal coupling algorithm, the middle column of Figure 6.2 shows the histograms of the samples from $\mu^y_{\ell-1}, \mu^y_\ell$ obtained with the sub-sampling algorithm, and the right column corresponds to the histograms of $\mu^y_{\ell-1}, \mu^y_\ell$ obtained with the IMH algorithm. For either column, each row represents a different level $\ell = 2$ (top), $\ell = 4$ (middle) and $\ell = 7$ (bottom). As it can be seen from Figure 6.2 (left), the maximal coupling algorithm is able to target the right marginal distributions at any level. This should not be a surprising fact, since each marginal chain is being created using RWM proposals.

We now investigate the coupling between chains, i.e., *how often* are the chains coupled when using our method. To that end, we define the *synchronization rate* at level ℓ by $\mathscr{S}_\ell := \frac{1}{N_\ell}\sum_{n=0}^{N_\ell - 1} \mathbb{1}_{\left\{u^n_{\ell,\ell} = u^n_{\ell,\ell-1}\right\}}$, which is an ergodic estimator of $\mathbb{P}_{\nu_\ell}(u_{\ell,\ell} = u_{\ell,\ell-1})$ (with ν_ℓ the joint probability measure with marginals $\mu^y_{\ell-1}, \mu^y_\ell$, induced by the ML-MCMC algorithm with maximal couplings). We run our

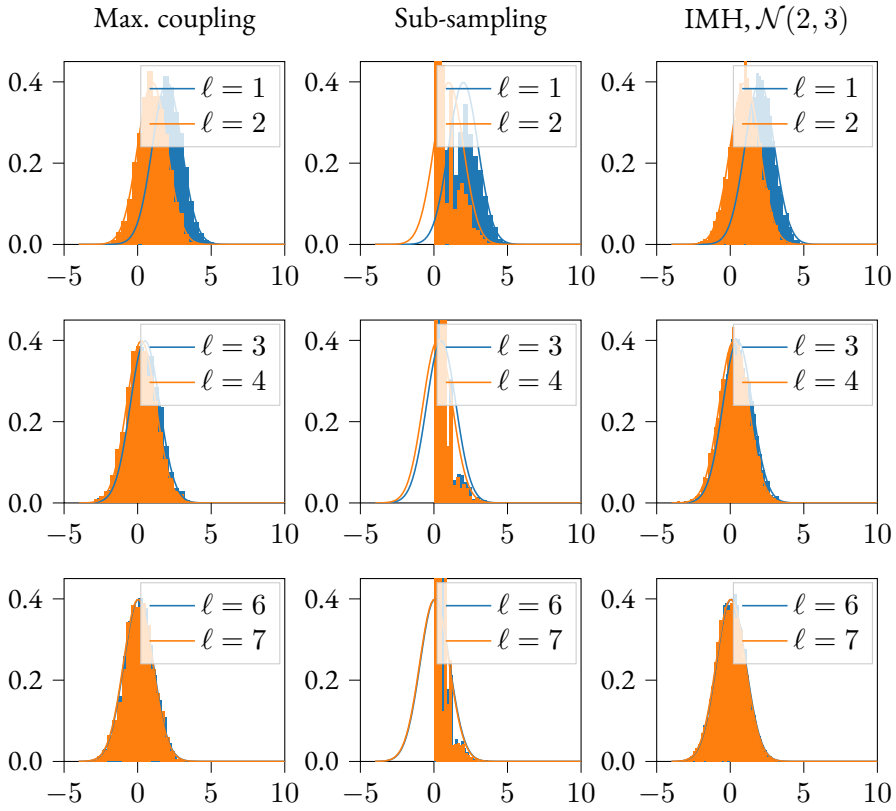

Figure 6.2: Histograms of samples obtained with a (left) maximal coupling of the proposals (center) with a sub-sampling algorithm and (right) with the independent Metrpolis-Hastings algorithm, for different pairs of accuracy levels. Each histogram is obtained with 20000.

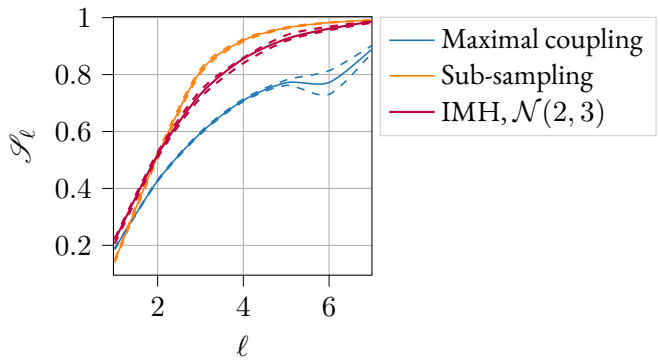

Figure 6.3: Average synchronization rate for the chains generated with maximally coupled proposals (blue) those generated by the sub-sampling algorithm (orange), and those obtained with IMH (burgundy). 95% confidence intervals are shown in dashed lines.

ML-MCMC Algorithm 8 together with Algorithm 12 $M = 50$ independent times, where each independent run has $\mathsf{L} = 7$, $N_\ell = 20,000$, $\ell = 0, 1, \ldots, \mathsf{L}$. Furthermore, for each independent run $k = 1, 2, \ldots, M$, we compute $\mathscr{S}_\ell^{(k)}$. We plot the estimated $\overline{\mathscr{S}_\ell} := \frac{1}{M}\sum_{k=1}^{M} \mathscr{S}_\ell^{(k)}$ at each level, together with a 95% confidence interval, in Figure 6.3. As we can see, the synchronization rates of all algorithms increase with ℓ, with the synchronization rate of the sub-sampling algorithm increasing faster.

Sampling with a re-synchronization kernel

The (simple) numerical experiments conducted so far show that our ML-MCMC method (a) is able to sample from the right marginal probability measures at each level ℓ and (b) the synchronization rate increases with ℓ. However, as evidenced in Figure 6.3, the synchronization rate for our method increases at slower rate than that of the sub-sampling or IMH algorithm. Ideally, one would like to have a coupled sampler that fulfills (a), while at the same time having a higher synchronization rate. As discussed in Section 6.2.3, this can be achieved by using a convex combination of our coupled sampler together with a re-synchronization kernel.

Remark 6.4.1 (On the use of re-synchronization kernel): *One should not expect to introduce an additional bias on the marginal chains when using a combination of joint Markov operators of the form* $\tilde{P} = \omega P_\ell^1 + (1-\omega)P_\ell^2, \omega \in (0,1)$, *provided that* $\mu_j^y P_\ell^i = \mu_j^y$, $j = \ell - 1, \ell$, $i = 1, 2$. *Indeed, since for* $i = 1, 2$, P_ℓ^i *is a Markov operator, it follows that* $\left\|P_\ell^i\right\|_{L_2} = 1$ *thus for* $\omega \in (0,1)$ $\tilde{P} = wP_\ell^1 + (1-\omega)P_\ell^2 \implies \left\|\tilde{P}\right\|_{L_2^0} \leq \omega + (1-\omega)\left\|P_\ell^2\right\|_{L_2^0} = a$, *with* $a < 1$ *provided* $\left\|P_\ell^2\right\|_{L_2^0} < 1$ *(i.e., the marginal max. coupling has a positive* L_2 *spectral gap). The choice of weight* ω *does affect the convergence rate though.*

Motivated by this, we shift our attention to the performance of our method when combined with either the sub-sampling algorithm of [45] or our IMH algorithm presented in the previous Chapter

as a re-synchronization kernel. To that end, we implement our ML-MCMC Algorithm 8 together with Algorithm 13. This induces two different Markov operators, $\mathbf{P}_{\mathrm{SS}_\ell}^{\mathsf{sync}}$, corresponding to the Markov operator which induces the re-synchronization via the sub-sampling algorithm (by using the emperical distribution form the samples obtained at level $\ell-1$ as a proposal), and $\mathbf{P}_{\mathrm{IMH}_\ell}^{\mathsf{sync}}$, which induces it using the IMH algorithm.

We consider a level independent weight $\omega_\ell = \omega$, for different values of $\omega \in \{0, 0.1, 0.2, 0.5, 0.7, 0.9\}$, where the level ℓ synchronization operator $\mathbf{P}_{(\cdot)_\ell}^{\mathsf{sync}}$ is to be understood as a step of the either the sub-sampling algorithm of [45] or our IMH algorithm presented in the previous section, with $Q_\ell = Q = \mathcal{N}(2,3)$. We first verify the accuracy of a ML-MCMC estimator of the form

$$\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}} := \frac{1}{N_0}\sum_{n=0}^{N_0}[\mathsf{Qol}_0(u_{0,0}^n)] + \sum_{\ell=1}^{\mathsf{L}}\frac{1}{N_\ell}\sum_{n=0}^{N_\ell}\underbrace{\left(\mathsf{Qol}_\ell(u_{\ell,\ell}^n) - \mathsf{Qol}_{\ell-1}(u_{\ell,\ell-1}^n)\right)}_{:=Y_\ell^n}. \tag{6.17}$$

(6.17) obtained with our mixed algorithm. We begin by first focusing only in the results generated by $\mathbf{P}_{\mathrm{SS}_\ell}^{\mathsf{sync}}$, since, as previously discussed, the sub-sampling algorithm by itself tends to give biased results for this particular hierarchy of posteriors. Setting $\mathsf{Qol} = u$, we clearly have that $\mathbb{E}_{\mu_\ell^y}[\mathsf{Qol}_\ell] = \mathbb{E}_{\mu_\ell^y}[u] = 2^{-\ell+2}$. Thus, one can investigate the accuracy of the ML-MCMC estimator, by computing $\widehat{\mathsf{Qol}}_{\mathsf{L},\{N_\ell\}_{\ell=0}^{\mathsf{L}}}$ and comparing it to $\mathbb{E}_{\mu_{\mathsf{L}}^y}[u]$. To that end, for each value of ω, we run our ML-MCMC algorithm $M = 50$ independent times, where each independent simulation is run with $\mathsf{L} = 7$ levels, using $N_\ell = 20,000$, $\ell = 0, 1, \ldots, \mathsf{L}$, samples per level, per run. For each ω we compute M independent level ℓ estimators $\widehat{\mathsf{Qol}}_{\ell,\{N_{\ell'}\}_{\ell'=0}^{\ell}}^{(\omega,k)}$, $k = 1, 2, \ldots, M$, and compute, for each value of ω, $\overline{\mathsf{Qol}}_\ell^{(\omega)} := \frac{1}{M}\sum_{k=1}^{M}\widehat{\mathsf{Qol}}_{\ell,\{N_{\ell'}\}_{\ell'=0}^{\ell}}^{(\omega,k)}$. For illustrative purposes, we plot $\overline{\mathsf{Qol}}_\ell^{(\omega)}$ v.s ℓ for each value of ω in Figure 6.4. As we can see, for values of $\omega \leq 0.7$, there does not seem to be any noticeable bias with respect to the true estimator. This is further confirmed in Figure 6.5, where the histograms of the resulting samples from our ML-MCMC algorithm are presented for $\omega = \{0.1, 0.5, 0.7\}$ (top, middle, and bottom row, respectively). As we can see, the histograms match the densities of μ_ℓ^y for different levels ℓ.

Lastly we plot, for both synchronization kernels (subsampling and IMH), the synchronization rate for each ω in Figure 6.6. As we can see from Figure 6.6 (left), for all values of $\omega > 0$, we obtain synchronization rates that go to 1 much faster than the one corresponding to $\omega = 0$. This result, together with the ones presented in Figures 6.4 and 6.5, suggest that one can combine the sub-sampling approach of [45] with our ML-MCMC based on maximal couplings using the convex combination of kernels presented in Section 6.4.1 with some carefully-chosen weights. Similarly, we can see from Figure 6.6 (Right) that introducing such a re synchronization kernel noticeably improves the synchronization between chains at two consecutive levels. Although, for this particular case there does not seem to be much difference between the synchronization rate

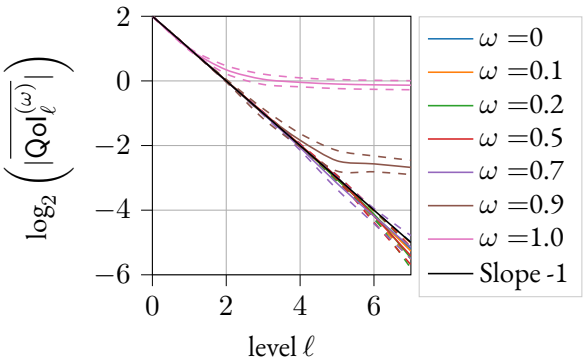

Figure 6.4: Mean multi-level estimator $\log_2\left(|\overline{\mathsf{QoI}_\ell^{(\omega)}}|\right)$, using the sub-sampling re-synchronization kernel. Estimates where computed for each value of ω, from 50 independent runs.

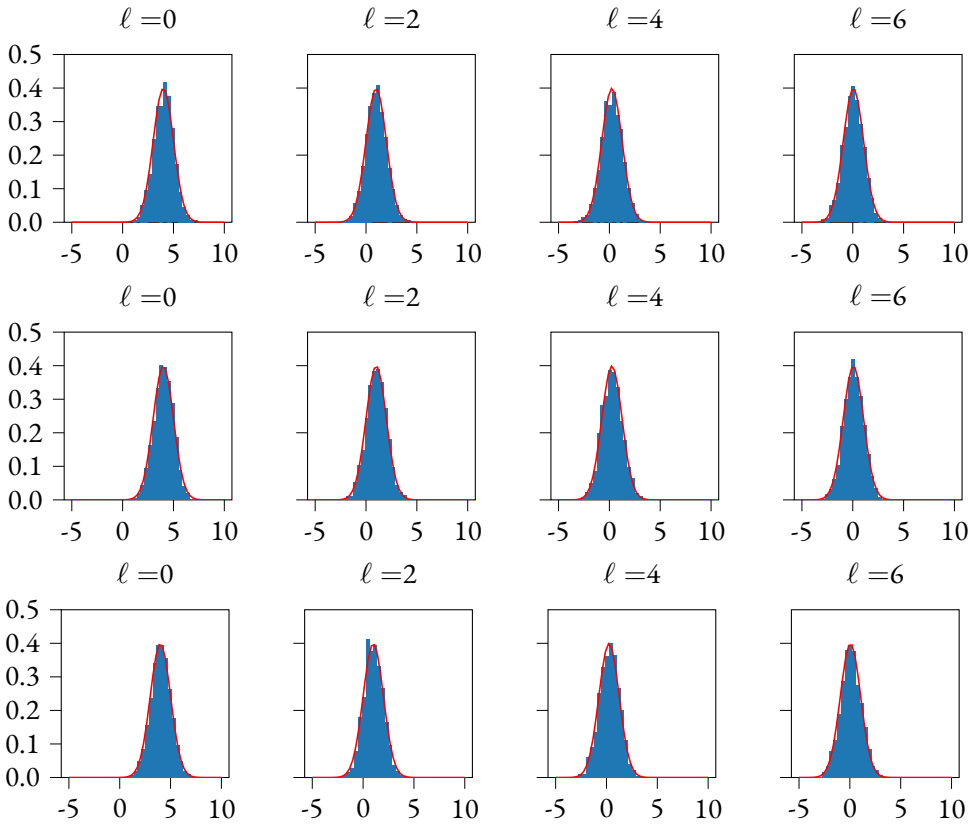

Figure 6.5: Histograms of the samples from μ_ℓ^y obtained with our Algorithm. Each row corresponds to a different value of $\omega = 0.1$ (top), $\omega = 0.5$ (middle), $\omega = 0.7$ (bottom).

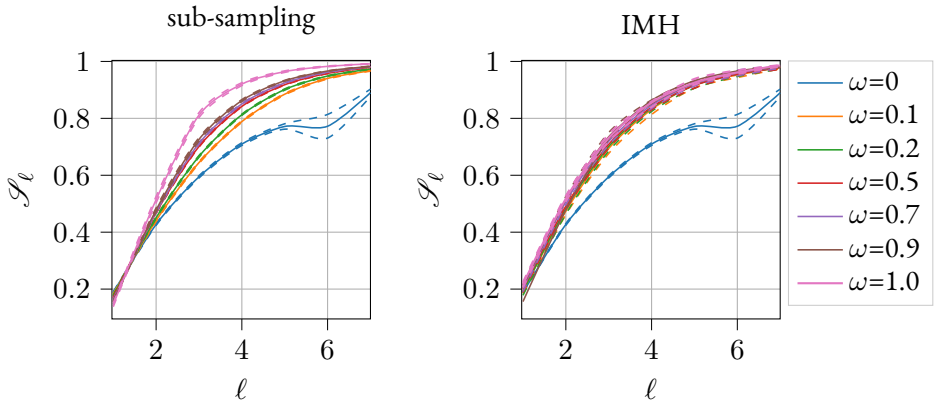

Figure 6.6: Mean synchronization rate for different values of ω and different synchronization kernels. (Left) re-sync. via sub-sampling kernel. (Right) re-sync using IMH kernel. Dashed lines represent 95% confidence intervals.

induced by two values $\omega_1, \omega_2 \geq 0.1$, we believe this to be problem dependent, and affected by several factors, such as dimensionality, multi-modality and choice of Q.

6.4.2 Subsurface flow: moderate-dimensions

We consider a more interesting example for which we try to recover the probability distribution of the permeability field κ in Darcy's subsurface flow equation, given some measurements of the hydraulic head on the physical domain. Let $\bar{D} = [0,1]^2$, $\mathsf{X} = \mathbb{R}^K$, $(x_1, x_2) =: x \in \bar{D}$, $\partial D = \Gamma_N \cup \Gamma_D$, with $\mathring{\Gamma}_N \cap \mathring{\Gamma}_D = \emptyset$, where $\Gamma_D := \{(x_1, x_2) \in \partial D, \text{ s.t. } x_1 = \{0, 1\}\}$, and $\Gamma_N = \partial D \backslash \Gamma_D$. Darcy's subsurface equation is given by

$$\begin{cases} -\nabla_x \cdot (\kappa(x,u)\nabla_x \mathsf{p}(x,u)) = 1, & x \in D,\ u \in \mathsf{X}, \\ \mathsf{p}(x,u) = 0 & x \in \Gamma_D,\ u \in \mathsf{X}, \\ \partial_n \mathsf{p}(x,u) = 0 & x \in \Gamma_N,\ u \in \mathsf{X}, \end{cases} \tag{6.18}$$

where p represents the pressure (or hydraulic head), and we simulate the stochastic permeability $\kappa(x,u)$ as a mean-zero stationary Gaussian field written in terms of its Karhunen-Loève expansion as

$$\log(\kappa(x,u)) = \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} \sqrt{\lambda_{m,n}}\phi_{m,n}(x)u_{m,n}, \quad u_{n,m} \sim \mathcal{N}(0,1) \tag{6.19}$$

with $\lambda_{m,n} = \frac{1}{\pi m^2}\frac{1}{\pi n^2}$, and $\phi_{m,n}(x) = \sin(m\pi x_1)\sin(n\pi x_2), \forall x \in \bar{D}, m, n \in \mathbb{N}$. Notice that the random permeability field can be recovered given the set of random parameters $\{u_{m,n}\}_{m,n\in\mathbb{N}}$. For computational purposes, we reorder Equation (6.19) in terms of a single index j in such a way

that $\lambda_{m,n} =: \lambda_j \leq \lambda_{j+1}$ (in case of equal values the order is chosen arbitrarily), and truncate such an expansion after K terms, thus obtaining the following approximation:

$$\log(\kappa(x,u)) \approx \sum_{j=1}^{K} \sqrt{\lambda_j}\phi_j(x)u_j, \quad u_j \sim N(0,1),\ j = 1,\dots,K. \tag{6.20}$$

Equation (6.18) can then be numerically approximated at a discretization level $\ell \geq 0$ using the finite element method on a triangular mesh of $2^\ell \cdot 22 \times 2^\ell \cdot 22$ piece-wise linear elements. Such a numerical approximation is done using the FEniCS package [101].
Data y is generated from the numerical solution of (6.18) observed at a grid of 4×4 uniformly spaced points inside D, polluted by a normally-distributed noise $\varepsilon \sim \mathcal{N}(0, \sigma^2_{\text{noise}} I_{16\times 16})$, where $I_{16\times 16}$ is the 16-dimensional identity matrix. In particular, the solution to Equation (6.18) is numerically approximated at a discretization level $\mathsf{L}^* = 6$, using a truncation parameter $K^* = 150$ in Equation (6.20), with a true set of parameters $u_j^* \sim N(0,1),\ j = 1,2,\dots,K^*$. Moreover, we set $\sigma_{\text{noise}} = 0.01$, which corresponds to, roughly, 1% measurement noise.
We set $K = 50$, $\ell \in \{0,1,2,3,4\}$, $\mathsf{L} = 4$, and define the map $u \mapsto \mathcal{F}_\ell(u)$ as the numerical approximation of the solution to Equation (6.18) at a discretization level ℓ, using a log-permeability field modeled by (6.20), and observed at a grid of 4×4 equally spaced points inside D, for a particular value of $u \in \mathsf{X}_\ell = \mathbb{R}^K$. Thus, the level-dependent potential is given by

$$\Phi_\ell(u;y) = \frac{1}{2\sigma^2_{\text{noise}}} \left\| y - \mathcal{F}_\ell(u) \right\|^2.$$

Furthermore, setting $\mu_{\mathsf{pr}} = \bigotimes_{i=1}^K \mathcal{N}(0,1)$, we can then define the level-ℓ posterior μ_ℓ^y in terms of its Radon-Nykodim derivative with respect to the prior as:

$$\frac{\mathrm{d}\mu_\ell^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u) = \frac{1}{Z_\ell} \exp\left(-\Phi_\ell(u;y)\right), \quad Z_\ell = \int_{\mathsf{X}} \exp\left(-\Phi_\ell(u;y)\right) \mu_{\mathsf{pr}}(\mathrm{d}u).$$

The BIP thus consists of sampling from the probability distribution $\log(\kappa(x,u))$ (parameterized in terms of $\{u_j\}_{j=1}^K$) conditioned on the noise-polluted observed data y.
We implement our ML-MCMC algorithm to sample from μ_{L}^y and compute posterior expectations at level L of a quantity of interest

$$\mathsf{QoI}(u) := \ln\left(-\int_{\Gamma_{D_1}} \kappa(x,u) \nabla \mathsf{p}(x,u) \cdot \mathbf{n}\, \mathrm{d}s \right), \tag{6.21}$$

where Γ_{D_1} denotes the rightmost boundary of the domain and $\mathbf{n}$ is the unit normal vector to Γ_{D_1}.
We will denote by QoI_ℓ as (6.21) computed with a level ℓ approximation of p.
We implement our method together with a re-synchronization kernel, where, similar to Section 6.4.1, we use the sub-sampling algorithm of [45] as a re-synchronizing kernel, with a level-

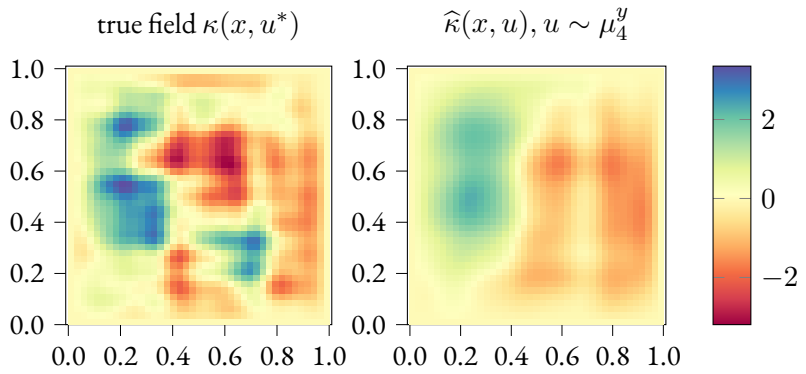

Figure 6.7: (Left) true log-field $\log(\kappa(x, u^*))$. (Right) Posterior mean estimator of $\log(\kappa(x, u))$ at level $\mathsf{L} = 4$.

dependent sub-sampling rate of $t_\ell = \max\left\{1 + 2\sum_{k=0}^{N} \hat{\varrho}_\ell(k), 5\right\}$, where $1 + 2\sum_{k=0}^{N} \varrho_\ell(k)$ is the integrated auto-correlation time of $\{\mathsf{QoI}_{\ell-1}(u^n_{\ell,\ell-1})\}_{n=0}^{N}$ at level $\ell - 1$. We do not compare our results to those of algorithm [45] since such a method can produce biased results when used by itself (c.f. Section 5.6.2). We implement the re-synchronization kernel with level-dependent weights ω_ℓ given by $\omega_1 = 0.1$, and $\omega_2 = \omega_3 = \omega_4 = 0.5$. This choice of weights was made in such a way that, at the coarsest level, the coupling is mostly driven by localized proposals, while the choice of higher weights for the higher levels is made in such a way that it favors "desirable" properties for the chains, such as having a rapidly decaying ACF for the marginal chains, or a rapidly increasing synchronization rate for the joint chains (c.f. Figure 6.9). An alternative (and certainly more systematic) approach for the selection of $\{\omega_\ell\}_{\ell=0}^{\mathsf{L}}$ could be to include a Bayesian update on their value inside a continuation-type ML-MCMC algorithm. Further investigation on the choice of these values will likely be the subject of future work.

As a verification of our method, we run our ML-MCMC algorithm 50 independent times, where each simulation is run with $N_\ell = 5000$ samples per level, for each level $\ell = 0, 1, \ldots, 4$. The true log-permeability field, together with the level L posterior mean are shown in Figure 6.7. As it can be seen, the level-L estimator is able to capture some of the more representative features of the true permeability field.

We begin by investigating the synchronization of the chains. In Figure 6.8 we plot $\mathsf{QoI}_{\ell-1}$ Vs QoI_ℓ for $\ell = 1, 2$ (top row, from left to right) and $\ell = 3, 4$ (bottom row, from left to right). As expected, samples become increasingly more concentrated on the diagonal as ℓ increases. Furthermore, we consider once again the synchronization rate at level ℓ given by $\mathscr{S}_\ell = \frac{1}{N_\ell}\sum_{n=0}^{N_\ell - 1} \mathbb{1}_{\left\{u^n_{\ell,\ell} = u^n_{\ell,\ell-1}\right\}}$, and compute $\mathscr{S}_\ell^{(k)}$ for each independent run $k = 1, \ldots, 50$. We plot $1 - \overline{\mathscr{S}_\ell}$, with $\overline{\mathscr{S}_\ell} = \frac{1}{M}\sum_{k=1}^{M} \mathscr{S}_\ell^{(k)}$, at each level, together with a 95% confidence interval, in Figure 6.9 (left). As we can see, the synchronization rates increase with ℓ. Notice that $1 - \overline{\mathscr{S}_\ell}$ can be understood as an estimator of $\mathsf{P} = \mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1})$. Furthermore, under the same setting, we plot the autocorrelation function (ACF) of QoI_ℓ fo each level $\ell = 0, 1, 2, 3, 4$ on Figure 6.9. The weights

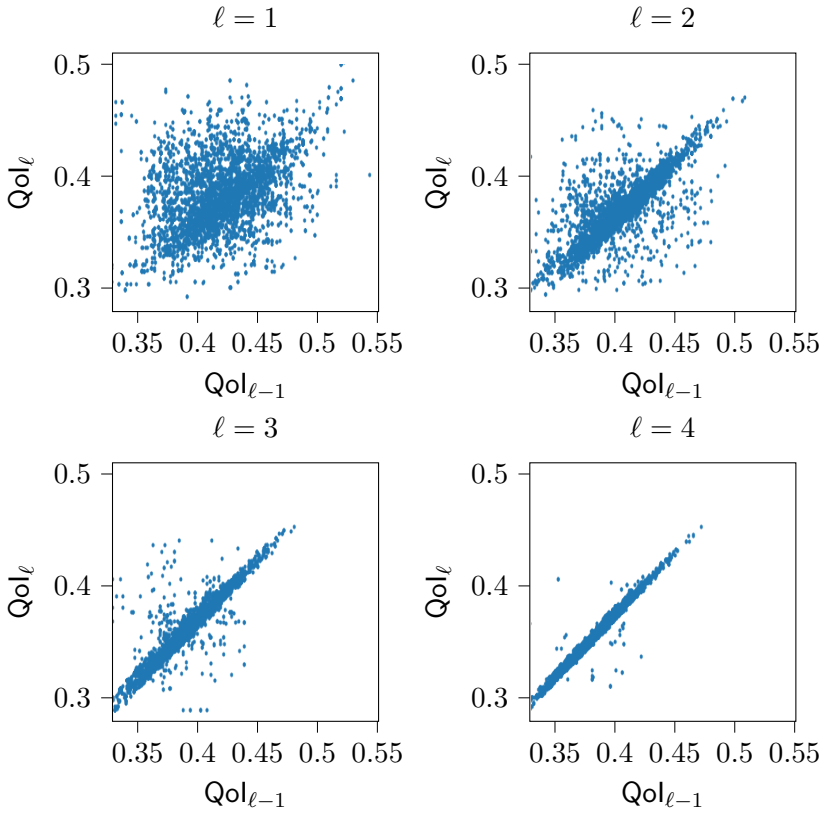

Figure 6.8: Diagonal plots of QoI_ℓ vs $\mathsf{QoI}_{\ell-1}$ for $\ell = 1, 2, 3, 4$.

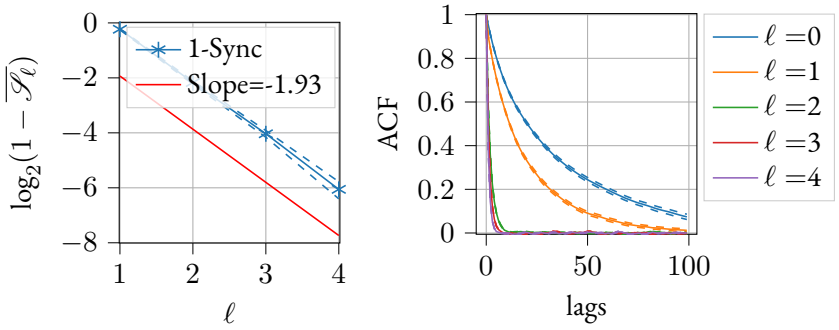

Figure 6.9: (Left) Estimated synchronization rate vs level. (Right) Mean autocorrelation plot (ACF) at lag 100. In both plots, dashed lines represent a 95% confidence interval obtained over 50 independent runs.

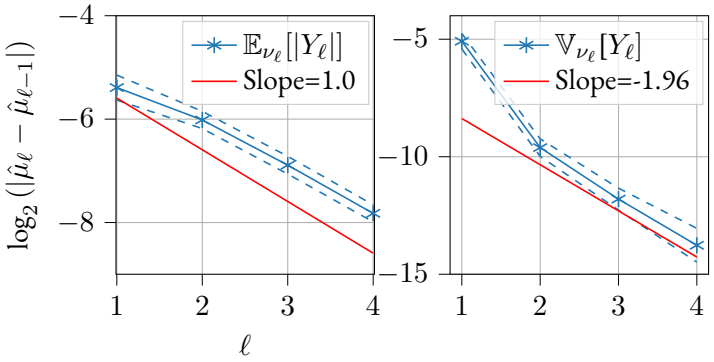

Figure 6.10: Convergence vs level. (Left) weak convergence, (right) strong convergence (log_2 scale in y axis). Dashed lines represent a 95% confidence interval obtained over 50 independent runs.

are chosen so that they produce a decay on the ACF with respect to ℓ, while at the same time conserving the explorability associated to our maximal coupling algorithm.

We now proceed to numerically investigate the rates α and β in the complexity theorem reported in Chapter 5 (c.f. Theorem 5.4.1). To that end, we estimate α and β using the same set-up as before (i.e., 50 independent runs with $N_\ell = 5000$ per level, per run) and plot estimates of $|\mathbb{E}_{\mu^y_{\ell-1}}[\mathsf{Qol}_{\ell-1}] - \mathbb{E}_{\mu^y_\ell}[\mathsf{Qol}_\ell]|$ and $\mathbb{V}_{\nu_\ell}[\mathsf{Qol}_\ell - \mathsf{Qol}_{\ell-1}]$ versus ℓ, together with a 95% confidence interval, in Figure 6.10 (right). As we can see, we obtain a weak decay rate $\alpha_w \approx 1.0$ and a strong decay $\beta \approx 1.96$. This in turn verifies numerically Assumptions T1 and T2 in Theorem 5.4.1.

Lastly, following the same discussion as in Section 5.5, we have that the optimal hierarchy of samples in our ML-MCMC algorithm for a given tolerance tol is given by

$$N_\ell = \left\lceil 2\mathsf{tol}^{-2}\sqrt{\frac{\sigma_\ell^2}{\mathsf{C}_\ell}}\left(\sum_{j=0}^{\mathsf{L}}\sqrt{\sigma_j^2\mathsf{C}_j}\right)\right\rceil,$$

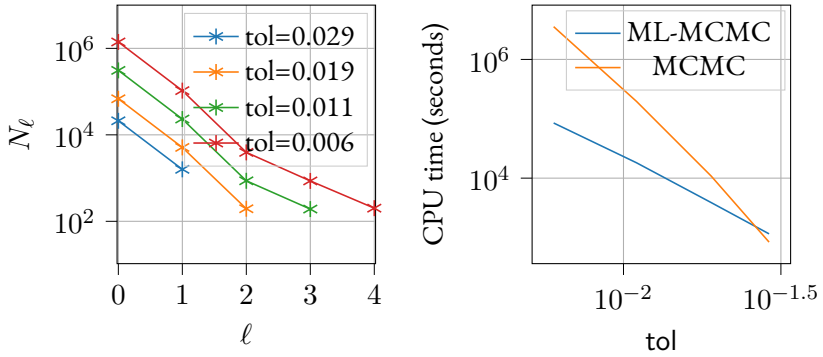

Figure 6.11: (Left). Number of samples $N_\ell(\mathsf{tol}_i)$ for a given tolerance tol_i, $i = 1, 2, 3, 4$. (Right). Comparison of the cost against a single-level MCMC algorithm.

with $\sigma_\ell^2 = \mathbb{V}_{\nu_\ell}[Y_\ell]$ which can be estimated, for instance, by a batched means estimator (see e.g., [54]) and where C_ℓ is the cost of obtaining one independent sample of $(\mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1}), \mathsf{QoI}_\ell(u_{\ell,\ell}))$ using our ML-MCMC algorithm. We plot the hierarchy of samples $\{N_\ell(\mathsf{tol}_i)\}_{\ell=0}^{\mathsf{L}}$ for different tolerances $\mathsf{tol}_i \in \{0.029, 0.019, 0.011, 0.006\}$ in Figure 6.11 (left), and the total computational cost (in seconds) vs tolerance of our ML-MCMC estimator vs that of its single-level counterpart in Figure 6.11 (right) . Figure 6.11 (left) suggests the computational advantage of using ML-MCMC; indeed for a tolerance of $\mathsf{tol} = 0.007$, a single-level MCMC algorithm would need over 10^6 (correlated) samples at level $\mathsf{L} = 4$. Meanwhile, the ML-MCMC algorithm, requires around 10^2 (correlated) samples at this level, since most of the computational effort is being done at the low discretization levels, where samples are inexpensive to obtain. These results are consistent with those of [45]. This is further corroborated in Figure 6.11 (right), where we plot the cost (in seconds) vs tolerance for both our ML-MCMC and a single level MCMC, where the number of samples necessary for the single-level (at level $\ell = \ell(\mathsf{tol})$) chain to achieve an error smaller than tol was estimated from 10 independent pCN runs of the single-level MCMC algorithm using 5000 samples per run. As it can be seen from such a figure, the ML-MCMC has a much smaller complexity than its single-level counterpart.

6.5 Appendix

6.5.1 A.1. Higher-dimensional subsurface flow revisited: some numerical results

Lastly, we revisit the same problem studied in in Section 5.6.4. In particular, we are interested in testing our maximal coupling algorithm without re-synchronization for this large-dimensional problem using a ν-pCN algorithm (c.f. Section 3.4.1) as a proposal. As we will see on this example, the maximal coupling algorithm can show promise on problems where the coupling is done in rather large dimensions. We remark however that this last section is rather *exploratory* in nature, and further investigation is needed.

We recall the formulation of the problem at hand for convenience. Consider once again the same setting as in Section 6.4.2, namely the Darcy's subsurface flow (6.18) with $\kappa(x,u) = e^{u(x)}$, with $u(x) \sim \mathcal{N}(0, \mathcal{A}^{-2}) = \mathcal{N}(0, \mathcal{C}) = \mu_{\mathsf{pr}}$ where the precision operator is the square of the differential operator $\mathcal{A}$ acting on a dense subspace $\mathrm{Dom}(\mathcal{A}) \subset L_2(D)$ of the form

$$\mathcal{A} = -\Delta + \frac{1}{2}I,$$

together with Robbin boundary conditions $\nabla(\cdot)\cdot\hat{n} + \frac{\sqrt{2}}{2}(\cdot) = 0$ (c.f. Sections 2.2.1 and 4.5.7). We assume that the data y is generated by solving equation (6.18), using a realization $u_{\mathsf{true}} \sim \mu_{\mathsf{pr}}$, and observing it at a grid of 10×10 equally spaced points in $[0.1, 0.9]^2$, polluted by some normally distributed noise $\eta \sim \mathcal{N}(0, \sigma^2_{\mathrm{noise}} I_{100\times 100})$ with $\sigma_{\mathrm{noise}} = 9.61 \times 10^{-5}$, corresponding to roughly 1% noise. Denoting by $u \mapsto \mathcal{F}(u)$ the mapping associated to solving Equation (6.18) with $\kappa(x,u) = e^{u(x)}$ and observing the solution at the given grid of points, we can then pose our BIP as sampling from μ^y with

$$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_{\mathsf{pr}}}(u) = \frac{1}{Z}\exp\left(-\|y - \mathcal{F}(u)\|^2_\Sigma\right), \quad \Sigma = \sigma^2_{\mathrm{noise}} I_{100\times 100}.$$

Once samples from μ^y have been obtained, we aim at approximating $\mathbb{E}_{\mu^y}[\mathsf{QoI}]$ where the quantity of interest $\mathsf{QoI} : \mathsf{X} \to \mathbb{R}$ is the log-flux through the bottom boundary $\Gamma_b := \{(x_1, x_2) \in \partial D \text{ s.t. } x_2 = 0\}$ defined as

$$\mathsf{QoI}(u) := \log\left(\int_{\Gamma_b} e^{u(x)} \nabla \mathsf{p}\cdot\hat{n}\,\mathrm{d}s\right).$$

Following Section 5.6.4, we introduce a sequence of discretization levels $\ell = 0, 1, 2, 3 = \mathsf{L}$ of the forward mapping operator $\mathcal{F}$ by numerically approximating Equation (6.18) using the finite-element method with $16\cdot 2^\ell \times 16 \cdot 2^\ell$ piece-wise-linear finite elements. This hierarchy of $\{\mathcal{F}_\ell\}_{\ell=0}^{\mathsf{L}}$ induces the family of of potential $\{\Phi_\ell(u;y)\}_{\ell=0}^{\mathsf{L}}$, with $\Phi_\ell(u;y) = \tilde{\Phi}(\mathcal{F}_\ell(\mathcal{P}_\ell^{\mathcal{A}}u); y)$, which in turn induces the family of posteriors $\{\widehat{\mu^y_\ell}\}_{\ell=0}^{\mathsf{L}}$, witch each posterior defined on $(\mathsf{X}_\ell, \mathcal{B}(\mathsf{X}_\ell))$, approximating μ^y as $\ell \to \infty$, with

$$\widehat{\mu^y_\ell}(\mathrm{d}u_{\ell,\ell}) = \frac{1}{Z_\ell}\exp\left(-\tilde{\Phi}(\mathcal{F}_{\ell-1}(u_{\ell,\ell}); y)\right)\mu_{\mathsf{pr}_\ell}(\mathrm{d}u_{\ell,\ell})$$

with μ_{pr_ℓ} the discretized prior, induced by the approximation of the operator $\mathcal{A}$, as discussed in Section 5.6.4. We are interested in investigating how, (or whether) we can use our proposed algorithm in such a (high-dimensional) case. To reiterate, this is of interest if, e.g., a sufficiently accurate Laplace approximation can not be built in this rather large number of dimensions, or if the sub-sampling approach of [45] cannot be applied. At each level $\ell = 0, 1, 2, 3$, we generate a

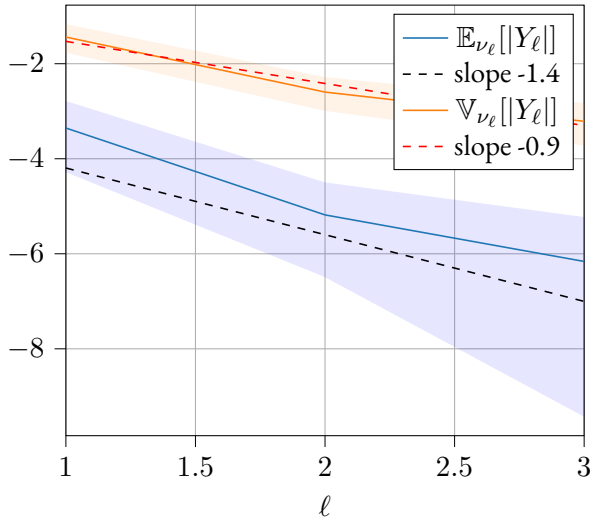

Figure 6.12: Rates for cost-tolerance theorem for the high-dimensional example with maximal-coupling. ($\log_2$-scale in the y-axis)

coupling of two proposal measures $Q_\ell(u_{\ell,\ell},\cdot)$ $R_\ell(u_{\ell,\ell-1},\cdot)$ in X_ℓ (i.e., the higher dimensional space induced by the FE discretization of the operator $\mathcal{A}$), where

$$Q_\ell(u_{\ell,\ell},\cdot) = \mathcal{N}(m_{\text{map},\ell} + \sqrt{1-\rho^2}(u_{\ell,\ell} - m_{\text{map},\ell}), \rho^2\mathcal{C}_{\text{Lap},\ell})$$
$$R_\ell(u_{\ell,\ell-1},\cdot) = \mathcal{N}(m_{\text{map},\ell} + \sqrt{1-\rho^2}(u_{\ell,\ell-1} - m_{\text{map},\ell}), \rho^2\mathcal{C}_{\text{Lap},\ell}),$$

We implement our proposed ML-MCMC algorithm for $M = 10$ independent runs, obtaining $N = 2000$ samples per level, per run. We obtain the following results. We estimate the rates for the cost-tolerance result of [45] (c.f. Theorem 5.4.2) in Figure 6.12. As we can see, although it is clear that there is indeed a decay on the rates for the variance and expected value of $Y_\ell(\boldsymbol{u}_\ell) = \mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1})$, this decay is not as fast as in the case of IMH. We plot $\mathsf{QoI}_{\ell-1}$ vs QoI_ℓ in Figure 6.13. We remark that, although there is no actual coupling between the chains such that $u_{\ell,\ell-1} = u_{\ell,\ell}$ (indeed, the proposed maximal coupling algorithm is designed to work in finite-dimensions), there is still a clear correlation between samples, which become increasingly more concentrated around the diagonal Δ. Lastly, we plot the number of samples and the computational cost of our ML-MCMC method compared to its single level counter part in Figure 6.14. As we can see, even in this case where we pay the extra price of not using an IMH, we can see that the total computational cost associated to the ML-MCMC algorithm has a much better complexity than its single-level counterpart.

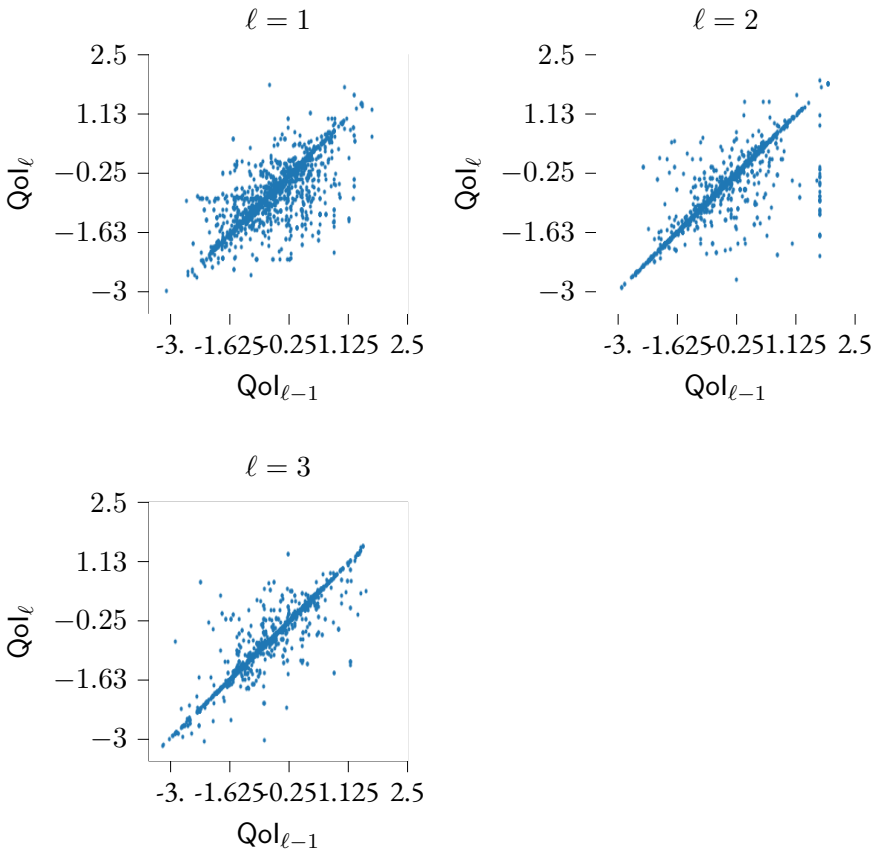

Figure 6.13: Diagonal plots of $\mathsf{QoI}_\ell(u_{\ell,\ell}) - \mathsf{QoI}_{\ell-1}(u_{\ell,\ell-1})$

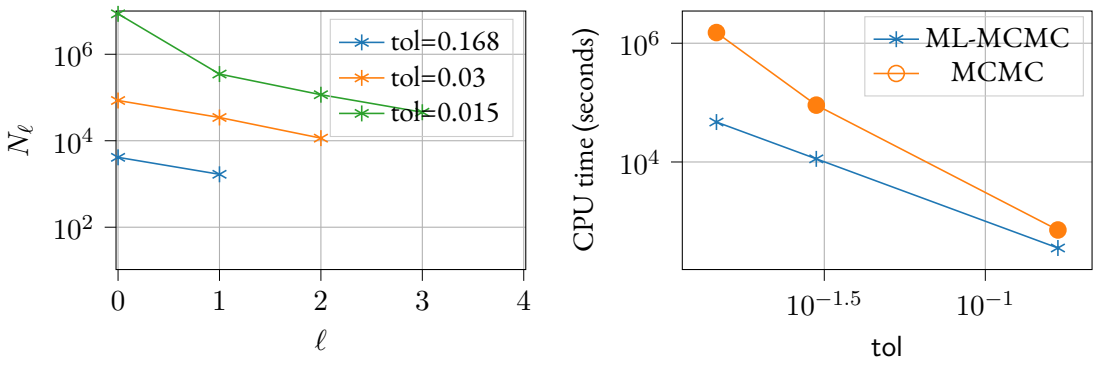

Figure 6.14: (Left) Number of samples per level for different tolerances. (Right). Total computational cost vs tolerance of ML-MLMC and single-level MCMC.

6.5.2 A.2. Some results towards the complexity study of the maximal coupling algorithm

As mentioned in Section 6.3 we were, at the time of the writing of this thesis, unable to show that the conditions required for [45, Theorem 3.5] to hold are satisfied under reasonable assumptions. It is known from Theorem 3.3.2 that Assumption T3 holds true provided that the join chain is mixing *sufficiently fast* (i.e., that it has finite *mixing time*, see [127] for more details), however, verifying this is highly technical, and as such, we will consider it to hold true. We will limit ourselves to verifying that Assumptions T1 and T2 hold for our setting, the latter of which would require some additional, potentially restrictive conditions. As we will see, in our formulation verifying this latter condition will require that $\int_{\Delta^c} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \not\to 0$ as $\ell \to \infty$, roughly understood as the chains always having a positive probability of re-synchronizing,and which we will assume to hold true. We formalize this discussion below.

Assumption 6.5.1: *For any $\ell \geq 0$, the following hold: There exist positive functions $C_{\mathcal{F}}, C_Q, C_\Phi : \mathsf{X} \to \mathbb{R}_+$ independent of ℓ, a positive constant C'_e, α independent of u and ℓ, and a positive constant $c_\ell \underset{\ell\to\infty}{\not\to} 0$ such that*

1. $\int_{\mathsf{X}} (C_{\mathcal{F}}(u')C_\Phi(u'))Q_\ell(u, \mathrm{d}u') \leq C_Q(u) < \infty,$
2. $\int_{\mathsf{X}} C_Q(u)\mu_{\mathsf{pr}}(\mathrm{d}u) \leq C'_e < \infty.$
3. $\int_{\Delta^c} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) > c_\ell.$

We remark that while Assumption 6.5.1.1-2 are relatively mild, Assumption 6.5.1.3 is difficult to verify and perhaps, too strong. For this reason we decided to include the following results as an appendix, as they possibly need further investigation. Our results will also rely upon Assumption 5.4.1, introduced in Chapter 5, on the potential Φ_ℓ, the forward operator $\mathcal{F}_\ell$ and the quantity of interest QoI_ℓ for $\ell \geq 0$.

Theorem 6.5.1: *Suppose Assumptions 5.4.1 and 6.5.1 hold. Then, Assumptions T1 and T2 are satisfied.*

Corollary 6.5.1: *Suppose additionally that Assumption T3 holds true. Then, the ML-MCMC algorithm based on maximally coupled proposals satisfies the conditions for Theorem 5.4.2 to hold.*

The proof of Theorem 6.5.1 is decomposed in a series of results and is given at the end of the section. It has been shown in Lemma 5.4.3 that T1 holds under Assumption 5.4.1. We recall such a result, for convenience.

Lemma 6.5.1: *Suppose Assumption 5.4.1 holds. Then, for any $\ell = 0, 1, \ldots \mathsf{L}$, there exists a positive constant $C_w \in \mathbb{R}_+$, independent of ℓ, such that:*

$$|\mathbb{E}_{\pi_\ell^y}[\mathsf{QoI}_\ell(u)] - \mathbb{E}_{\pi^y}[\mathsf{QoI}(u)]| \leq C_w s^{-\alpha_w \ell},$$

with $\alpha_w = \min\{\alpha_q, \alpha\}$ and α_q, α as in Assumption 5.4.1.

Recall that for any given level $\ell = 0, 1, \ldots, \mathsf{L}$, we say that the joint chains created by the ML-MCMC algorithm are *synchronized* at step n if $u^n_{\ell,\ell} = u^n_{\ell,\ell-1} = u$. We say that they are *unsynchronized* otherwise. Notice that, in our setting, if the chains are synchronized, Algorithm 11 will propose the same candidate state u' to both chains with probability 1. Thus, assuming that the chains are synchronized at step n, they will become unsynchronized at step $n+1$ with probability $|\alpha_\ell(u,u') - \alpha_{\ell-1}(u,u')|$. We now show that such a probability approaches 0 as $\ell \to \infty$, using a simplified version of Lemma 5.4.4 in Chapter 5.

Lemma 6.5.2: *Suppose Assumptions 5.4.1.1 hold. Then, the following bound holds*

$$\left|\alpha_\ell(u,u') - \alpha_{\ell-1}(u,u')\right| \leq h'_\ell(u,u') s^{-\alpha\ell}, \quad u, u' \in \mathsf{X},$$

with

$$h'_\ell(u,u') := \left[\frac{e^{-\Phi_\ell(u';y)} + e^{-\Phi_{\ell-1}(u';y)}}{e^{-\Phi_\ell(u;y)}} C_\Phi(u') C_{\mathcal{F}}(u') \right.$$
$$\left. + e^{-\Phi_{\ell-1}(u';y)} \left(e^{\Phi_{\ell-1}(u;y)} + e^{\Phi_{\ell-1}(u;y)}\right) C_\Phi(u) C_{\mathcal{F}}(u)\right] s^{-\alpha\ell}$$

Proof. From the definition of α_ℓ, and the fact that $\psi(x) := \min\{1, x\}$ is Lipschitz continuous with a constant of 1, it can be seen that

$$\left|\alpha_\ell(u,u') - \alpha_{\ell-1}(u,u')\right| \leq \left|\frac{e^{-\Phi_\ell(u';y)}}{e^{-\Phi_\ell(u;y)}} - \frac{e^{-\Phi_{\ell-1}(u';y)}}{e^{-\Phi_{\ell-1}(u;y)}}\right|$$
$$\leq e^{\Phi_\ell(u;y)} \left|e^{-\Phi_\ell(u';y)} - e^{-\Phi_{\ell-1}(u';y)}\right| + \frac{e^{-\Phi_{\ell-1}(u';y)}}{e^{-\Phi_\ell(u;y)} e^{-\Phi_{\ell-1}(u;y)}} \left|e^{-\Phi_\ell(u;y)} - e^{-\Phi_{\ell-1}(u;y)}\right|. \tag{6.22}$$

Assuming $\Phi_\ell(u') \leq \Phi_{\ell-1}(u')$, a straightforward application of the mean value theorem gives

$$\left|e^{-\Phi_\ell(u';y)} - e^{-\Phi_{\ell-1}(u';y)}\right| \leq e^{-\Phi_\ell(u';y)} \left|\Phi_\ell(u';y) - \Phi_{\ell-1}(u';y)\right|. \tag{6.23}$$

Similarly, for the case $\Phi_\ell(u') \geq \Phi_{\ell-1}(u')$, we obtain

$$\left|e^{-\Phi_\ell(u';y)} - e^{-\Phi_{\ell-1}(u';y)}\right| \leq e^{-\Phi_{\ell-1}(u';y)} \left|\Phi_\ell(u';y) - \Phi_{\ell-1}(u';y)\right|. \tag{6.24}$$

Thus, from (6.23)-(6.24) it follows that

$$\left|e^{-\Phi_\ell(u';y)} - e^{-\Phi_{\ell-1}(u';y)}\right| \leq \left(e^{-\Phi_{\ell-1}(u';y)} + e^{-\Phi_\ell(u';y)}\right) \left|\Phi_\ell(u';y) - \Phi_{\ell-1}(u';y)\right|. \tag{6.25}$$

Thus, from (6.22), (6.25) and Assumption 5.4.1, we obtain:

$$(6.22) \leq \left[\frac{e^{-\Phi_\ell(u';y)} + e^{-\Phi_{\ell-1}(u';y)}}{e^{-\Phi_\ell(u;y)}} C_\Phi(u') C_{\mathcal{F}}(u') \right.$$
$$\left. + e^{-\Phi_{\ell-1}(u';y)} \left(e^{\Phi_{\ell-1}(u;y)} + e^{\Phi_{\ell-1}(u;y)} \right) C_\Phi(u) C_{\mathcal{F}}(u) \right] s^{-\alpha\ell}$$

□

Lemma 6.5.3: *Suppose Assumptions 5.4.1 and 6.5.1 hold, and denote the diagonal set of* X^2 *as* $\Delta = \{(u, u') \in \mathsf{X}^2 \text{ s.t. } u = u'\}$. *The transition probability to* Δ^c *for the coupled chain of Algorithm 12 is such that*

$$\boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c) \leq h_\ell(u) s^{-\alpha\ell}, \quad \forall \boldsymbol{u}_\ell = (u, u) \in \Delta,$$

with

$$h_\ell(u) = \left[2C_Q(u) e^{\Phi_\ell(u;y)} + \left(e^{\Phi_{\ell-1}(u;y)} + e^{\Phi_\ell(u;y)} \right) C_\Phi(u) C_{\mathcal{F}}(u) \right].$$

Proof. Since $\boldsymbol{u}_\ell \in \Delta$, we set $u_{\ell,\ell} = u_{\ell,\ell-1} = u$. Furthermore, in this case, we have that $u'_{\ell,\ell} = u'_{\ell,\ell-1}$ since only Case I in Algorithm 11 will happen. It then follows from Lemma 6.5.2 that:

$$\begin{aligned}
\boldsymbol{p}_\ell((u,u), \Delta^c) &= \int_{\mathsf{X}} |\alpha_{\ell-1}(u, u') - \alpha_\ell(u, u')| Q_\ell(u, \mathrm{d}u') \\
&\leq s^{-\alpha\ell} \int_{\mathsf{X}} \frac{e^{-\Phi_\ell(u';y)} + e^{-\Phi_{\ell-1}(u';y)}}{e^{-\Phi_\ell(u;y)}} C_\Phi(u') C_{\mathcal{F}}(u') Q_\ell(u, \mathrm{d}u') \\
&+ s^{-\alpha\ell} \int_{\mathsf{X}} e^{-\Phi_{\ell-1}(u';y)} \left(e^{\Phi_{\ell-1}(u;y)} + e^{\Phi_\ell(u;y)} \right) C_\Phi(u) C_{\mathcal{F}}(u) Q_\ell(u, \mathrm{d}u').
\end{aligned} \tag{6.26}$$

Since from Assumption 6.5.1 we have that $\int_{\mathsf{X}} e^{-\Phi_\ell(u';y)} C_\Phi(u') C_{\mathcal{F}}(u') Q_\ell(u, \mathrm{d}u') \leq C_Q(u)$, it then follows that

$$(6.26) \leq s^{-\alpha\ell} \left[2C_Q(u) e^{\Phi_\ell(u;y)} + \left(e^{\Phi_{\ell-1}(u;y)} + e^{\Phi_\ell(u;y)} \right) C_\Phi(u) C_{\mathcal{F}}(u) \right].$$

Setting $h(u) = \left[2C_Q(u) e^{\Phi_\ell(u;y)} + \left(e^{\Phi_{\ell-1}(u;y)} + e^{\Phi_{\ell-1}(u;y)} \right) C_\Phi(u) C_{\mathcal{F}}(u) \right]$ gives the desired result. □

Lemma 6.5.4: *Suppose* Assumptions 5.4.1 *and* 6.5.1 *hold. Then, there exists a positive constant* C_h *independent of the level such that*

$$\int_{\Delta} h_\ell(\boldsymbol{u}_\ell)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \leq C_h, \boldsymbol{u}_\ell = (u,u) \in \Delta,$$

Proof. Since $\boldsymbol{u}_\ell \in \Delta$, we have

$$\begin{aligned}\int_{\Delta} h_\ell(\boldsymbol{u}_\ell)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) = &\underbrace{\int_{\mathsf{X}^2} e^{\Phi_{\ell-1}(u_{\ell,\ell-1};y)} C_\Phi(u_{\ell,\ell-1}) C_{\mathcal{F}}(u_{\ell,\ell-1})\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell)}_{=I_1} \\ &+ \underbrace{\int_{\mathsf{X}^2} e^{\Phi_{\ell}(u_{\ell,\ell};y)} C_\Phi(u_{\ell,\ell}) C_{\mathcal{F}}(u_{\ell,\ell})\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell)}_{=I_2} \\ &+ \underbrace{2\int_{\mathsf{X}^2} e^{\Phi_{\ell}(u_{\ell,\ell};y)} C_Q(u_{\ell,\ell})\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell)}_{=I_3}\end{aligned}$$

Since $u \in \Delta$, we can marginalize over each component on both I_1 and I_2. We begin with I_1, integrating over $u_{\ell,\ell}$:

$$\begin{aligned}I_1 &= \int_{\mathsf{X}^2} e^{\Phi_{\ell-1}(u_{\ell,\ell-1};y)} C_\Phi(u_{\ell,\ell-1}) C_{\mathcal{F}}(u_{\ell,\ell-1})\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\ &= Z_{\ell-1}^{-1} \int_{\mathsf{X}} C_\Phi(u_{\ell,\ell-1}) C_{\mathcal{F}}(u_{\ell,\ell-1})\mu_{\mathsf{pr}}(\mathrm{d}u_{\ell,\ell-1}) \\ &\leq C_e c_I^{-1} \quad \text{(from Assumption 5.4.1 and Lemma 5.4.1).}\end{aligned}$$

A similar procedure for I_2, also yields $I_2 \leq C_e C_I^{-1}$. Lastly, we focus on I_3. Integrating over $u_{\ell,\ell-1}$ gives

$$\begin{aligned}I_3 &= \int_{\mathsf{X}^2} e^{\Phi_{\ell}(u_{\ell,\ell};y)} C_Q(u_{\ell,\ell})\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) = Z_\ell^{-1} \int_{\mathsf{X}} C_Q(u_{\ell,\ell})\mu_{\mathsf{pr}}(\mathrm{d}u_{\ell,\ell}) \\ &\leq C_e' c_I^{-1} \quad \text{(from Assumption 6.5.1 and 5.4.1).}\end{aligned}$$

Taking $C_h = c_I^{-1}(C_e' + 2C_e)$ gives the desired result. □

Lemma 6.5.5: *Suppose Assumptions* 5.3.1, 5.4.1 *and* 6.5.1 *hold. Then, for all* $\ell = 1, 2, \ldots, \mathsf{L}$, *there exist a positive constant* C_r *independent of* ℓ *such that*

$$\mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1}) \leq C_r s^{-\alpha\ell}, \quad \forall n \in \mathbb{N}.$$

Proof. Since $\boldsymbol{P}_\ell$ is ν_ℓ-invariant, we can write

$$\begin{aligned}
&\mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1}) = \int_{\Delta^c} \nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) = \int_{\mathsf{X}^2} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&= 1 - \int_{\mathsf{X}^2} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) \\
&= 1 - \int_{\Delta} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) - \int_{\Delta^c} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell).
\end{aligned}$$

$$\begin{aligned}
&\leq 1 - \int_{\Delta} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) - c_\ell \mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1}) \quad \text{(from Assumption 6.5.1)} \\
&= \int_{\Delta} \boldsymbol{p}_\ell(\boldsymbol{u}_\ell, \Delta^c)\nu_\ell(\mathrm{d}\boldsymbol{u}_\ell) - c_\ell \mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1}).
\end{aligned}$$

It then follows from Lemmata 6.5.3 and 6.5.4 that

$$\mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1}) \leq \frac{C_h}{c_\ell} s^{-\alpha_w \ell} \leq \frac{C_h}{c'} s^{-\alpha_w \ell},$$

where $0 < c' := \inf_{\ell \in \mathbb{N}}\{c_\ell\}$, by Assumption 6.5.1. □

Lemma 6.5.6: *Suppose Assumptions 5.4.1, 5.3.1 and 6.5.1 hold. Then, for any $\ell \geq 1$, there exists a positive constant C_v such that*

$$\mathbb{V}_{\nu_\ell}[Y_\ell] \leq C_v s^{-\beta\ell},$$

where $\beta = \min\{2\alpha_q, \alpha(1 - 2/m)\}$, and α, α_q, m as in Assumption 5.4.1.

Proof. Having shown Lemma 6.5.5, the proof of this Lemma becomes the same as that of Lemma 5.4.7. □

7 Finalizing remarks

The last chapter of this thesis is divided into two parts. In the first one, we summarize and draw some conclusions from the material presented in Chapters 4 through 6. In the second part we identify and discuss several possible research directions extending the work presented in previous chapters.

7.1 Summary and conclusions

In this thesis, we have developed, analyzed and implemented different novel hierarchical MCMC techniques with the aim of alleviating some of the computational challenges arising in modern, large scale Bayesian inverse problems.

The first hierarchical method we presented was the *Generalized Parallel Tempering* method, an extension of the well-known *parallel tempering* algorithm [52], used primarily to sample from probability distributions that are multi-modal or that concentrate around a lower-dimensional, non-linear manifold. Inspired by the *infinite swapping* methodology of Doll et. al., [47] (who propose an algorithm aimed at improving the efficiency of continuous-time Markov chains arising in the field of molecular dynamics), we introduced two tempering techniques based on state-dependent kernel swaps. We provided a thorough convergence analysis of these methods; indeed, we were able to show that under some technical conditions on the marginal Markov transition kernels and marginal (in the context of tempering) probability measures, both of our proposed methods are reversible and convergent with respect to their own invariant measure. Furthermore, we implemented and successfully applied these methodologies to sample from several multi-modal probability distributions arising in the context of BIP. In addition, we presented an extensive discussion on the implementation and potential shortcomings of these methods. We were able to see that, at the experimental level, our proposed methodologies clearly outperform (in terms of total computational cost VS. variance of a given estimator) several competing algorithms. An additional advantage of our proposed algorithms is that they can be seen, to some extent, as "self-tuning", since the choice of swaps between chains (in UGPT) or kernels (in WGPT), is done automatically, eliminating the need of fixing this swapping schedule apriori, as it has been typically done in the parallel-tempering literature. Lastly, we also implemented these methods for the solution of a high-dimensional BIP based on a hyperbolic (i.e., acoustic wave) PDE. To the best of the author's knowledge, tempering techniques have seldom been applied to tackle such high-dimensional problems, and even more so to those arising from wave phenomena, for which the literature on BIP is rather scarce.

We remark that the framework considered in Chapter 4 can be combined with other, more advanced MCMC algorithms, such as, e.g., the Metropolis-adjusted Langevin algorithm (MALA) (c.f. Section 3.4.1), or the Delayed Rejection Adaptive Metropolis (DRAM) [63]. Furthermore, in principle, such a method can also be combined with geometry-informed, dimension-independent samplers such as the ones presented in [12, 36].

The second hierarchical method we presented in this work was a class of multi-level MCMC algorithms based on independent Metropolis-Hastings proposals. We presented several contributions to the emerging sub-field of ML-MCMC.

From a methodological perspective, we extended the seminal work of [45], by devising a ML-MCMC method based on a class of independent Metropolis Hastings proposals fulfilling certain technical conditions. This is an important contribution in the sense that, previous ML-MCMC algorithms based solely on sub-sampling the posterior distribution at the previous accuracy level, could lead to biased results for a certain class of problems. In addition, we presented a continuation-type ML-MCMC algorithm in the spirit of [31, 132], in the hope of making the ML-MCMC procedure both efficient and robust.

From a theoretical perspective we investigated the existence and uniqueness of a joint invariant measure for this class of techniques, and presented conditions on the level dependent posteriors and proposal kernels under which such a joint invariant measure exists. Furthermore, we were able to show that the joint ML-MCMC algorithm has a uniformly ergodic convergence to such a probability measure, a generally desirable attribute for MCMC samplers. In addition, we extended the complexity results of [45] to our setting; indeed, their result was formulated specifically for their choice of proposal. Lastly, we implemented our proposed methodology on an array of BIP, of both low and high dimensionality, where we validated some of our theoretical results. Once again, we can see that there is a clear computational and methodological advantage to the methods we advocate in this work.

In the last part of this thesis we presented a novel ML-MCMC based on maximally coupled proposals. This setting can be though of as a generalization of our previous methodology, in the sense that it allows for both state-dependent and state-independent proposals; indeed, the way coupled chains are being generated in Chapter 5 (and in [45]) can be thought of as a (rather trivial) maximal coupling of an independent proposal kernel $Q_\ell(\cdot)$ with itself. Being able to construct ML-MCMC algorithms with state-dependent proposals is of great interest from a methodological perspective, as it can overcome some of the drawbacks associated to previous ML-MCMC methodologies. We presented guidelines on how to construct this ML-MCMC sampler using maximal coupling techniques, and, although the focus of this chapter was more on the methodological aspect, we showed that under certain technical conditions there exists a unique invariant joint measure for this type of ML-MCMC algorithms, similarly to the case of the ML-MCMC based on independent proposals. Although at the time of the writing of this thesis, we were unable to analytically verify that the complexity results of [45] could be extended to this method (under reasonable assumptions), numerical simulations suggest that our method presents a clear computational advantage with respect to its single-level counterpart.

7.2 Perspectives

As evidenced by the previous subsection, the proposed methods on this thesis show a lot of promise for their application to large-scale BIP. However, there is, of course, plenty of room for future work both on the theoretical and practical side. The theoretical analysis on the presented methodologies could (and should) be refined.
Our convergence results for the GPT show that the rate of convergence is no worse than that of the slowest-converging chain, however, experimental results suggest that there is a much more dramatic improvement in the convergence of the algorithm.
Concerning the ML-MCMC algorithms, although it is clear that the invariant joint measure induced by the ML-MCMC algorithms depends heavily on the choice of proposal mechanism (contrary to the single-level MCMC case), a more precise description of this dependency is not available at the time of the writing of this work. Further developing and understanding the theoretical aspects behind such methodology would be an interesting continuation of this work.
Additionally, from a methodological perspective, a natural question that arises in the use of ML-MCMC methods is their extension to *multi-fidelity techniques*, where, instead of constructing the hierarchy of forward mapping operators $\{\mathcal{F}_\ell\}_{\ell=0}^{\mathsf{L}}$ based on several levels of discretization accuracy ℓ, one constructs it using a hierarchy of so-called "fidelity models" of $\mathcal{F}$; which could be, e.g., models using increasingly refined physics, Gaussian processes, or low-rank approximations of $\mathcal{F}$. Using multi-fidelity models in the context of statistical inference has been discussed in, e.g., [128, Section 4]. Similarly, one could try to devise a multi-index Markov chain Monte Carlo method based on the ideas presented in [66, 78] and the work presented in this thesis.
From an application perspective, it would be desirable to see the methods discussed in this work applied to other large-scale and potentially more realistic simulations.
In addition to the perspectives discussed in the previous paragraphs, we identify and discuss in slightly more detail the following research directions.

7.2.1 Normalizing flows and ML-MCMC

A drawback associated to our ML-MCMC approach based on IMH is that, in general, it is not easy to find suitable (IMH) proposals. This is particularly true whenever the underlying posterior is high-dimensional and not well-approximated by a Gaussian probability measure. One possible way of alleviating this issue is to construct said proposals using normalizing flows (c.f. Section 2.3.2). In this context, one could visualize a novel ML-MCMC algorithm as follows. Suppose that, at a given level ℓ, we have already collected samples from μ_ℓ^y, on all levels $\ell = 0, 1, 2, \dots \mathsf{L}$, which could have been achieved, e.g., by a previous iteration of C-ML-MCMC algorithm (c.f. Section 5.5). Given a state $\boldsymbol{u}_\ell^n$ we could generate a coupled sample $\boldsymbol{u}_\ell^{n+1}$ with $u_{\ell,\ell-1}^{n+1} \sim \mu_{\ell-1}^y$ and $u_{\ell,\ell}^{n+1} \sim \mu_\ell^y$, using the following procedure:

1. Sample $u_{\ell,\ell-1}^{n+1} \sim \mu_{\ell-1}^y$, using, e.g., sub-sampling approach [45].

2. Obtain $u'_{\ell,\ell} = T_\ell(u^{n+1}_{\ell,\ell-1})$, where for all $\ell \geq 0$, T_ℓ is a *normalizing flow* in the sense of Section 2.3.2 (i.e., a class of bijections from X_ℓ to X_ℓ whose determinant is, in some sense, inexpensive to compute) that maps $\mu^y_{\ell-1}$ into μ^y_ℓ, built in such a way that T_ℓ becomes *easier* to compute as $\ell \to \infty$. As an ansatz one could take, e.g., $T_\ell = I + \delta_\ell$, where I is the identity transformation and $\delta_\ell \to 0$ as $\ell \to \infty$.

3. Set $u^{n+1}_{\ell,\ell} = u'_{\ell,\ell}$ as the new state of the chain with marginal μ^y_ℓ with probability

$$\alpha_\ell(u^n_{\ell,\ell}, u'_{\ell,\ell}) = \min\left\{1, \frac{\mu^y_\ell(u'_{\ell,\ell})}{\mu^y_\ell(u^n_{\ell,\ell})}\frac{\rho_\ell(u^n_{\ell,\ell})}{\rho_\ell(u'_{\ell,\ell})}\right\},$$

where

$$\rho_\ell(x) := \mu^y_{\ell-1}(T^{-1}(x))|\det J_{T_\ell^{-1}}(x)|,$$

otherwise set $u^{n+1}_{\ell,\ell} = u^n_{\ell,\ell}$.

We illustrate the potential use of these techniques in the following (borderline trivial) example. Suppose we are interested in sampling from the family of distributions

$$\mu^y_\ell = \mathcal{N}(2^{-\ell+2}, 1),$$

which approximate $\mu^y = \mathcal{N}(0,1)$ as $\ell \to \infty$. For this particular case, one has that for any $u, v \in \mathsf{X}$, $T_\ell(u) = u - m_{\ell-1} + m_\ell$, where for any $\ell \geq 0$, $m_\ell = 2^{-\ell+2}$. Similarly, $T_\ell^{-1}(v) = v + m_{\ell-1} - m_\ell$ and $|\det J_{T_\ell^{-1}}(u)| = 1$. We implement a ML-MCMC algorithm using this method, the sub-sampling ML-MCMC algorithm of [45], and the maximal coupling algorithm. For all algorithms we take $\mathsf{L} = 2$ and $N_\ell = 5000$, $\ell = 0, 1, 2, 3$. Results are shown in Figures 7.1 and 7.2. As it can be seen in Figure 7.1 where we plot the histograms of the samples obtained with each method for different levels, the proposed approach is able to correctly sample from the right marginals. However, and perhaps more interestingly, is Figure 7.2, where we plot $u_{\ell,\ell-1}$ vs $u_{\ell,\ell}$ for different levels. As it can be seen, the correlation between the samples from the proposed method is stronger than those from the sub-sampling and the maximal-coupling methodologies. Although Figures 7.1 and 7.2 show some promising results, there are still some open questions regarding this approach. We identify the following:

1. There is a large overhead cost for training (deep) neural networks. In many cases such networks are trained using specialized clusters of graphical processing units (GPUs), which are currently more expensive that CPU clusters. As an example, the samples obtained from a normalizing flow depicted in Figure 7.3 were obtained by implementing RealNVP on a single 12GB NVIDIA Tesla K80 GPU (implemented via the Google Colab$^{\text{TM}}$ platform) and required a little over two hours to train. Such transformation was trained using prevously obtained posterior samples .

2. Given that T_ℓ will usually have an extremely complex structure, it might be difficult to guarantee that the induced proposal $\rho = T_{\ell\sharp}\mu^y_{\ell-1}$ satisfies the Assumptions necessary for its chain to be (uniformly) ergodic.

3. Currently, our theoretical results rely upon showing that $\mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1}) \to 0$ as $\ell \to \infty$. This is not the case with this proposed methodology, as one would have that $\mathbb{P}_{\nu_\ell}(u_{\ell,\ell} \neq u_{\ell,\ell-1}) = 1$, however, with $\|u_{\ell,\ell} - u_{\ell,\ell-1}\|_{\mathsf{X}} < \epsilon_\ell$ for some $\epsilon_\ell \to 0$ as $\ell \to \infty$. This implies that the theoretical analysis is slightly more involved that the one presented in Chapter 5.

7.2.2 On the use and analysis of more efficient couplings

The maximal coupling ML-MCMC algorithms discussed so far have been constructed using a maximal coupling of the proposal kernels for each individual chain. We have also limited our case to only using diffusion-based proposals (c.f. Section 3.4), such as pCN, to create the coupled chains. These ideas can be extended based on the recent works [19, 70, 169]. More precisely, [19] introduces a coupling between chains using a mixture between Hamiltonian Monte Carlo and a spherical coupling, such as the one presented in Algorithm 11 (c.f. Section 3.4). Their results seem to suggest that such a mixture of methods is more robust with respect to the dimensionality of the target measure when compared to just using spherical couplings, in the sense that the average meeting time between two chains having the same invariant measure, started at two different points in space seems to increase with the dimension of the space at a significantly slower rate (if at all) than that of Algorithm 11 (see, e.g., [70, Section 5.2]). The work [169] presents a way of generating maximal couplings between Markov transition kernels; as opposed to just coupling the proposals, by modifying the algorithms presented in [76]. Intuitively, this would result in a "stronger" type of coupling (i.e., increasing the probability of the event $u_{\ell,\ell} = u_{\ell,\ell-1}$), thus making this approach interesting to our ML-MCMC setting.

7.2.3 Towards a multi-level generalized parallel tempering

A natural extension to the work presented in this thesis is to combine both our proposed generalized parallel tempering and the discussed ML-MCMC methods; indeed, by introducing and exploiting hierarchies in both temperature and discretization, one could, in theory, propose a multi-level MCMC algorithm that is robust to multi-modality or measure concentration, i.e., a novel MCMC algorithm exploiting the attractive points of both approaches. This is not, however, a trivial extension of these works, as we shall discuss shortly after introducing some notation. For any $\ell = 0, 1, 2, \ldots \mathsf{L}$, let $K = K(\ell)$ denote the (level-dependent) number of temperatures $T_1, \ldots, T_K$, (inducing $K(\ell)$ parallel chains), let $S_{K(\ell)}$ denote the subset of possible permutations at level ℓ of cardinality $|S_{K(\ell)}| \in \mathbb{N}$, and for any $j = 1, 2, \ldots, K$, write

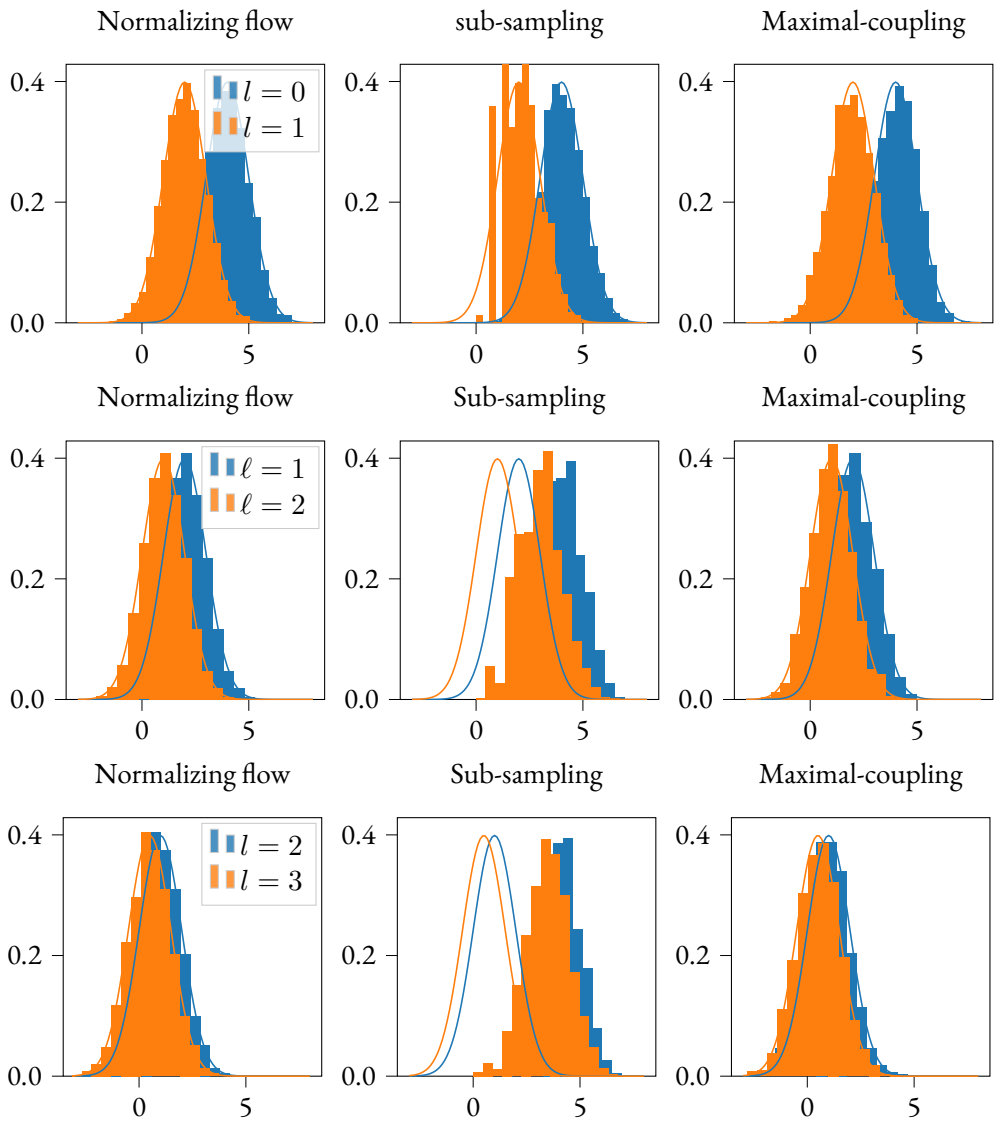

Figure 7.1: Histograms of samples for different ; from top to bottom: $\ell = 0, 1, 2$.

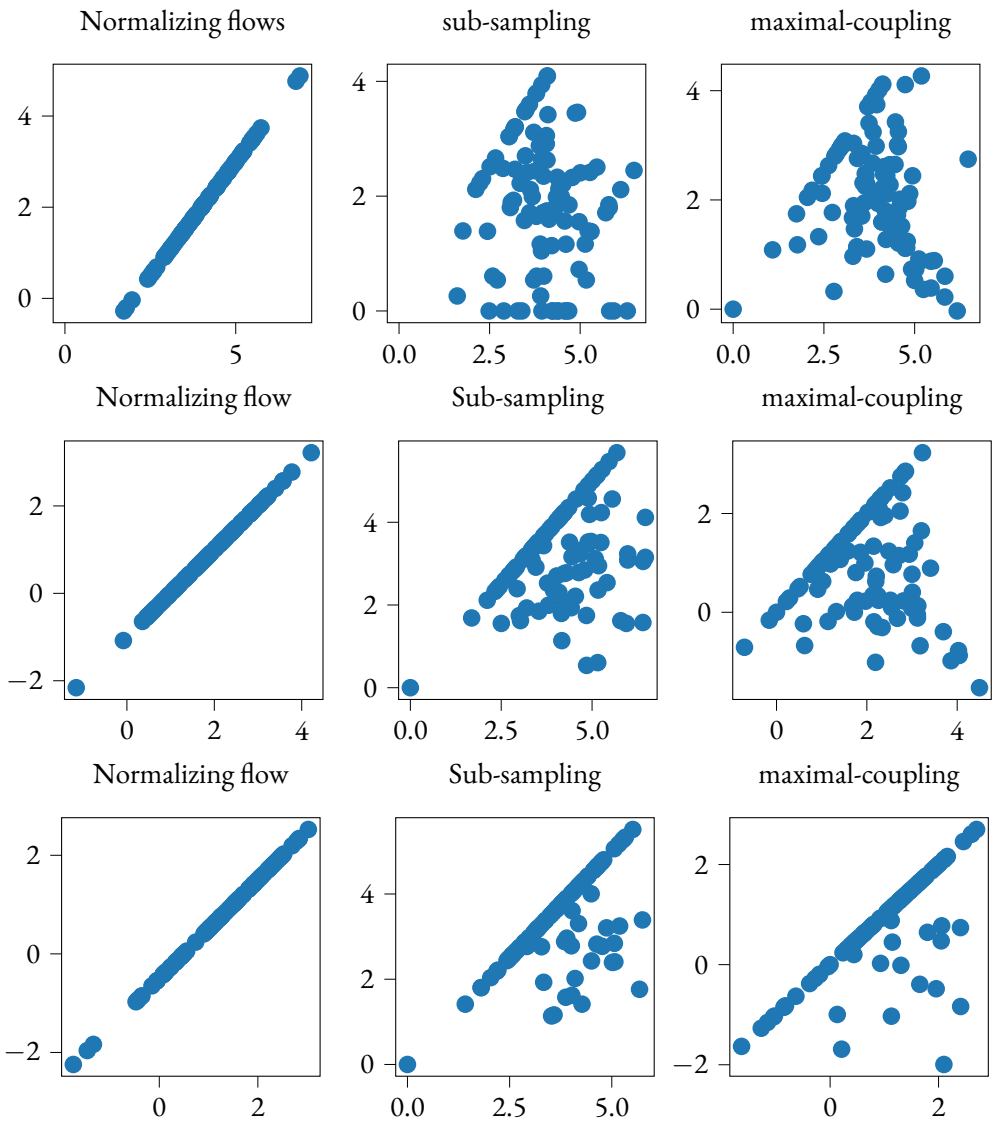

Figure 7.2: diagonal plots of samples $(u_{\ell,\ell-1}, u_{\ell,\ell})$ for different levels level; from top to bottom: $\ell = 0, 1, 2$.

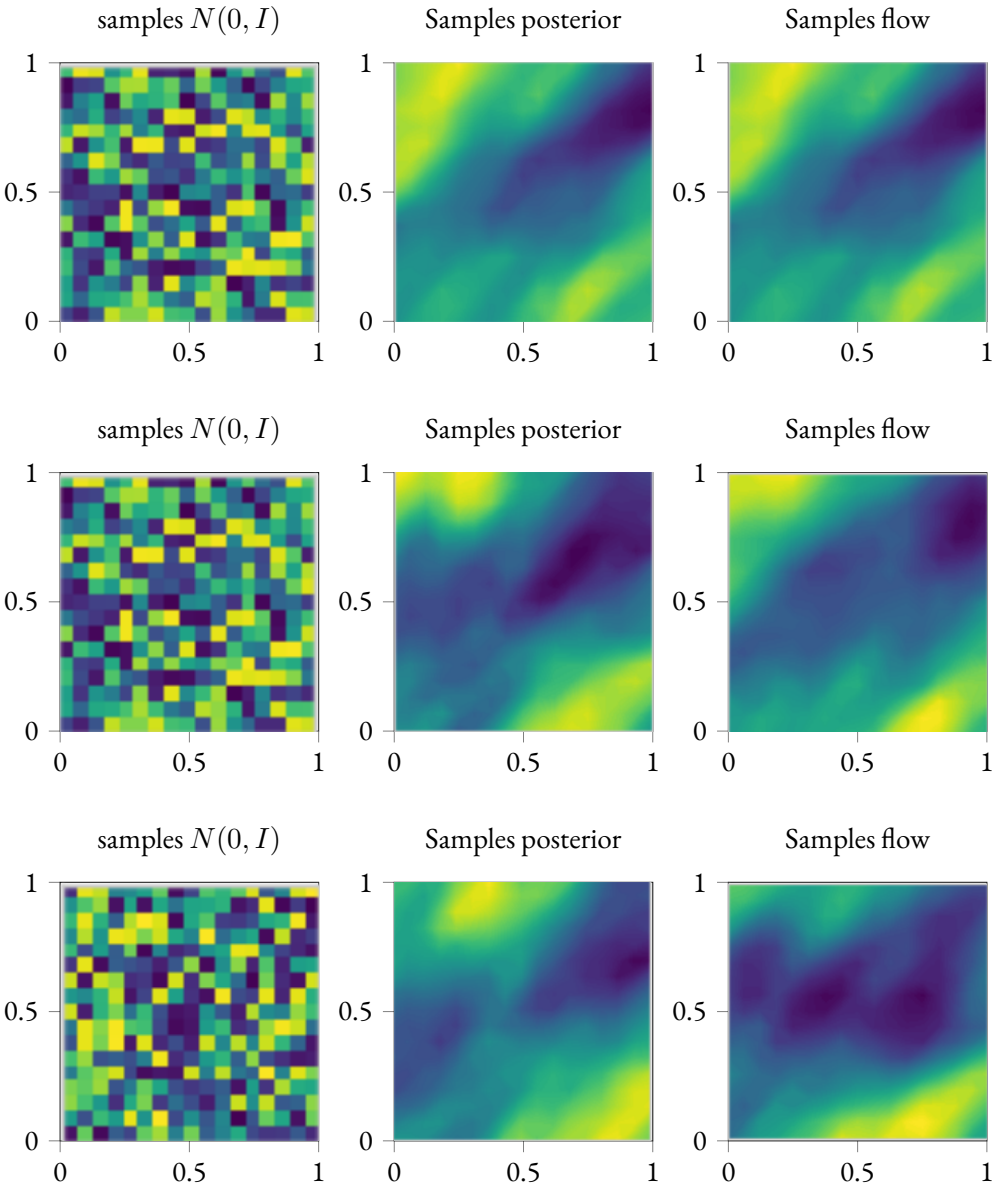

Figure 7.3: (Left). Samples from $\mathcal{N}(0, I_{17\times17})$. (Middle) posteriors samples form a subsurface flow BIP. (Right) Samples obtained with a normalizing flow.

$\mu^y_{\ell,j}(\mathrm{d}u) = \exp(-\Phi_\ell(u;y)T_j^{-1})\mu_{\mathsf{pr}}(\mathrm{d}u)$. Furthermore, let $\boldsymbol{u}_\ell = (u_{\ell,1}, u_{\ell,2}, \dots, u_{\ell,K})$ and for some $\sigma \in S_{K(\ell)}$ let $\boldsymbol{u}_{\ell,\sigma} = (u_{\ell,\sigma(1)}, u_{\ell,\sigma(2)}, \dots, u_{\ell,\sigma(K)})$. Lastly, define

$$\begin{aligned}
\boldsymbol{\mu}^y_\ell &:= \mu^y_{\ell,1} \times \mu^y_{\ell,2} \times \cdots \times \mu^y_{\ell,K}, \\
\boldsymbol{\mu}^y_{\ell,\sigma} &:= \mu^y_{\ell,\sigma(1)} \times \mu^y_{\ell,\sigma(2)} \times \cdots \times \mu^y_{\ell,\sigma(K)}, \\
\boldsymbol{\mu}^y_{\mathrm{W},\ell} &:= \frac{1}{|S_{K(\ell)}|} \sum_{\sigma \in S_{K(\ell)}} \boldsymbol{\mu}^y_{\ell,\sigma}.
\end{aligned}$$

Given an accuracy level L and a μ^y_{L}-integrable quantity of interest QoI, we are interested in computing $\mathbb{E}_{\mu^y_{\mathsf{L}}}[\mathsf{QoI}]$, which can be done via, e.g., the WGPT approach (c.f. Section 4.3.4) using an estimator of the form:

$$\begin{aligned}
\mathbb{E}_{\mu^y_{\mathsf{L}}}[\mathsf{QoI}_{\mathsf{L}}] = [\mathsf{QoI}_{\mathsf{L}}(u_{\mathsf{L}})] &= \mathbb{E}_{\boldsymbol{\mu}^y_{\mathsf{L}}}\left[\mathsf{QoI}_{\mathsf{L}}(u_{\mathsf{L},1})\right] \\
&= \frac{1}{|S_{K(\ell)}|} \sum_{\sigma \in S_{K(\ell)}} \mathbb{E}_{\boldsymbol{\mu}^y_{\mathrm{W},\mathsf{L}}}\left[\mathsf{QoI}(u_{\mathsf{L},\sigma(1)}) \frac{\mathrm{d}\boldsymbol{\mu}^y_{\mathsf{L}}}{\mathrm{d}\boldsymbol{\mu}^y_{\mathrm{W},\mathsf{L}}}(\boldsymbol{u}_{\mathsf{L},\sigma})\right] \\
&= \frac{1}{|S_{K(\mathsf{L})}|} \sum_{\sigma \in S_{K(\ell)}} \mathbb{E}_{\boldsymbol{\mu}^y_{\mathrm{W},\mathsf{L}}}[f_{\mathsf{L}}(\boldsymbol{u}_{\mathsf{L},\sigma})] \quad \boldsymbol{u}_{\mathsf{L},\sigma} \sim \boldsymbol{\mu}^y_{\mathrm{W},\mathsf{L}},
\end{aligned}$$

where we set $f_{\mathsf{L}}(\boldsymbol{u}_{\mathsf{L},\sigma}) = \mathsf{QoI}(u_{\mathsf{L},\sigma(1)}) \frac{\mathrm{d}\boldsymbol{\mu}^y_{\mathsf{L}}}{\mathrm{d}\boldsymbol{\mu}^y_{\mathrm{W},\mathsf{L}}}(\boldsymbol{u}_{\mathsf{L},\sigma})$. Under the convention that $\mathsf{QoI}_{-1} := 0$, this previous expectation can in turn be written in terms of the usual telescoping sum associated to multi-level techniques as:

$$\mathbb{E}_{\mu^y_{\mathsf{L}}}[\mathsf{QoI}_{\mathsf{L}}] = \sum_{\ell=0}^{\mathsf{L}} \frac{1}{|S_{K(\ell)}|} \left(\sum_{\sigma \in S_{K(\ell)}} \mathbb{E}_{\boldsymbol{\mu}^y_{\mathrm{W},\ell}}[f_\ell(\boldsymbol{u}_{\ell,\sigma})] - \mathbb{E}_{\boldsymbol{\mu}^y_{\mathrm{W},\ell-1}}[f_{\ell-1}(\boldsymbol{v}_{\ell-1,\sigma})] \right), \quad (7.1)$$

where $\boldsymbol{u}_{\ell,\sigma} \sim \boldsymbol{\mu}^y_{\mathrm{W},\ell}$ and $\boldsymbol{v}_{\ell-1,\sigma} \sim \boldsymbol{\mu}^y_{\mathrm{W},\ell-1}$. Given that samples from $\boldsymbol{\mu}^y_{\mathrm{W},\ell-1}$ are generated with a kernel of the form

$$\mathbf{p}^{(\mathrm{W}),\ell}(\boldsymbol{u}_\ell, \cdot) := \sum_{\sigma \in S_K} w_{\ell,\sigma}(\boldsymbol{u}_\ell) \mathbf{p}_{\ell,\sigma}(\boldsymbol{u}, \cdot),$$

with

$$w_{\ell,\sigma}(\boldsymbol{u}_\ell) = \frac{\mathrm{d}\boldsymbol{\mu}^y_{\ell,\sigma}}{\mathrm{d}\boldsymbol{\mu}^y_{\mathrm{W},\ell}}(\boldsymbol{u}_{\mathsf{L}}),$$

one then needs to devise a clever way of generating samples from this mixture of kernels, while at the same time keeping the terms $f_\ell, f_{\ell-1}$ in the ergodic estimator of Equation (7.1) highly

correlated. Perhaps a simpler approach is to consider the UGPT algorithm to generate samples, in which case one would obtain the simpler expression

$$\mathbb{E}_{\mu_{\mathsf{L}}^y}[\mathsf{Qol}_{\mathsf{L}}] = \sum_{\ell=0}^{\mathsf{L}} \left(\mathbb{E}_{\mu_{\ell}^y}\left[\mathsf{Qol}_{\ell}\right] - \mathbb{E}_{\mu_{\ell-1}^y}\left[\mathsf{Qol}_{\ell-1}\right] \right),$$

however, in this case one would still need to be careful when constructing the coupling between samplers, since these tempering methods tend to propose rather "large" jumps in the state space, which could rapidly become problematic if, e.g., the chain targeting $\mu_{\ell,1}^y$ makes a large jump and the chain targeting $\mu_{\ell-1,1}^y$ does not (which could happen, e.g., when the swapping kernel of the UGPT algorithm samples two permutations ρ, σ of $\boldsymbol{u}_\ell$ and $\boldsymbol{u}_{\ell-1}$ respectively, with $\rho(1) \neq \sigma(1)$) as it would "inflate" the variance between chains.

Bibliography

[1] Keiiti Aki and Paul G Richards. *Quantitative seismology*. 2002.

[2] R B Ash. *Probability and measure theory*. Harcourt/Academic Press, Burlington, MA, 2000.

[3] Søren Asmussen and Peter W Glynn. *Stochastic simulation: algorithms and analysis*, volume 57. Springer Science & Business Media, 2007.

[4] Marco Ballesio, Joakim Beck, Anamika Pandey, Laura Parisi, Erik von Schwerin, and Raúl Tempone. Multilevel monte carlo acceleration of seismic wave propagation under uncertainty. *GEM-International Journal on Geomathematics*, 10(1):1–43, 2019.

[5] Marco Ballesio, Ajay Jasra, Erik von Schwerin, and Raul Tempone. A Wasserstein coupled particle filter for multilevel estimation. *arXiv preprint arXiv:2004.03981*, 2020.

[6] Johnathan M Bardsley, Antti Solonen, Heikki Haario, and Marko Laine. Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems. *SIAM Journal on Scientific Computing*, 36(4):A1895–A1910, 2014.

[7] E Bax and J Franklin. A finite-difference sieve to compute the permanent. *CalTech-CS-TR-96-04*, 1996.

[8] Joakim Beck, Ben Mansour Dia, Luis FR Espath, Quan Long, and Raul Tempone. Fast Bayesian experimental design: Laplace-based importance sampling for the expected information gain. *Computer Methods in Applied Mechanics and Engineering*, 334:523–553, 2018.

[9] Joakim Beck, Ben Mansour Dia, Luis Espath, and Raúl Tempone. Multilevel double loop Monte Carlo and stochastic collocation methods with importance sampling for Bayesian optimal experimental design. *International Journal for Numerical Methods in Engineering*, 121(15):3482–3503, 2020.

[10] A. Beskos, A. Jasra, N. Kantas, and A. Thiery. On the convergence of adaptive sequential Monte Carlo methods. *Ann. Appl. Probab.*, 26(2):1111–1146, 2016.

[11] A. Beskos, A. Jasra, E. Muzaffer, and A.M. Stuart. Sequential Monte Carlo methods for Bayesian elliptic inverse problems. *Stat. Comp.*, 25:727–737, 2015.

[12] Alexandros Beskos, Mark Girolami, Shiwei Lan, Patrick E Farrell, and Andrew M Stuart. Geometric MCMC for infinite-dimensional inverse problems. *Journal of Computational Physics*, 335:327–351, 2017.

[13] Alexandros Beskos, Ajay Jasra, Kody Law, Raul Tempone, and Yan Zhou. Multilevel sequential Monte Carlo samplers. *Stochastic Processes and their Applications*, 127(5):1417–1440, 2017.

[14] Alexandros Beskos, Ajay Jasra, Ege A Muzaffer, and Andrew M Stuart. Sequential Monte Carlo methods for Bayesian elliptic inverse problems. *Statistics and Computing*, 25(4):727–737, 2015.

[15] Joris Bierkens. Non-reversible Metropolis-Hastings. *Statistics and Computing*, 26(6):1213–1228, 2016.

[16] Ilias Bilionis and Nicholas Zabaras. Solution of inverse problems with limited forward solver evaluations: a Bayesian perspective. *Inverse Problems*, 30(1):015004, 2013.

[17] Patrick Billingsley. *Convergence of probability measures*. Wiley Series in Probability and Statistics: Probability and Statistics. John Wiley & Sons Inc., New York, second edition, 1999. A Wiley-Interscience Publication.

[18] Vladimir Igorevich Bogachev. *Gaussian measures*. Number 62. American Mathematical Soc., 1998.

[19] Nawaf Bou-Rabee, Andreas Eberle, Raphael Zimmer, et al. Coupling and convergence for Hamiltonian Monte Carlo. *Annals of Applied Probability*, 30(3):1209–1250, 2020.

[20] Susanne C Brenner, L Ridgway Scott, and L Ridgway Scott. *The mathematical theory of finite element methods*, volume 3. Springer, 2008.

[21] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov chain Monte Carlo*. CRC press, 2011.

[22] Tan Bui-Thanh and Omar Ghattas. A scalable algorithm for MAP estimators in Bayesian inverse problems with Besov priors. *Inverse Problems & Imaging*, 9(1):27, 2015.

[23] Tan Bui-Thanh, Omar Ghattas, James Martin, and Georg Stadler. A computational framework for infinite-dimensional Bayesian inverse problems Part I: The linearized case, with application to global seismic inversion. *SIAM Journal on Scientific Computing*, 35(6):A2494–A2523, 2013.

[24] Tan Bui-Thanh and Quoc P Nguyen. FEM-based discretization-invariant MCMC methods for PDE-constrained Bayesian inverse problems. *Inverse Problems & Imaging*, 10(4):943, 2016.

[25] Ismaël Castillo and Richard Nickl. On the Bernstein–von Mises phenomenon for nonparametric Bayes procedures. *The Annals of Statistics*, 42(5):1941–1969, 2014.

[26] Guy Chavent. *Nonlinear least squares for inverse problems: theoretical foundations and step-by-step guide for applications*. Springer Science & Business Media, 2010.

[27] Jeff Cheeger. A lower bound for the smallest eigenvalue of the Laplacian. In *Proceedings of the Princeton conference in honor of Professor S. Bochner*, pages 195–199, 1969.

[28] Jeff Cheeger. A lower bound for the smallest eigenvalue of the Laplacian. In *Problems in analysis*, pages 195–200. Princeton University Press, 2015.

[29] Alexey Chernov, Håkon Hoel, Kody JH Law, Fabio Nobile, and Raul Tempone. Multilevel ensemble kalman filtering for spatio-temporal processes. *Numerische Mathematik*, 147(1):71–125, 2021.

[30] K Andrew Cliffe, Mike B Giles, Robert Scheichl, and Aretha L Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3, 2011.

[31] Nathan Collier, Abdul-Lateef Haji-Ali, Fabio Nobile, Erik Von Schwerin, and Raúl Tempone. A continuation multilevel Monte Carlo algorithm. *BIT Numerical Mathematics*, 55(2):399–432, 2015.

[32] Simon L Cotter, Gareth O Roberts, Andrew M Stuart, and David White. MCMC methods for functions: modifying old algorithms to make them faster. *Statistical Science*, pages 424–446, 2013.

[33] Matteo Croci, Michael B Giles, Marie E Rognes, and Patrick E Farrell. Efficient white noise sampling and coupling for multilevel Monte Carlo with nonnested meshes. *SIAM/ASA Journal on Uncertainty Quantification*, 6(4):1630–1655, 2018.

[34] C.Schillings and A.M.Stuart. Analysis of the ensemble Kalman filter for inverse problems. *SINUM*, 55:1264–1290, 2017.

[35] Tiangang Cui, Gianluca Detommaso, and Robert Scheichl. Multilevel dimension-independent likelihood-informed MCMC for large-scale inverse problems. *arXiv preprint arXiv:1910.12431*, 2019.

[36] Tiangang Cui, Kody JH Law, and Youssef M Marzouk. Dimension-independent likelihood-informed MCMC. *Journal of Computational Physics*, 304:109–137, 2016.

[37] Giuseppe Da Prato and Jerzy Zabczyk. *Stochastic equations in infinite dimensions*. Cambridge university press, 2014.

[38] Lisandro Dalcín, Rodrigo Paz, and Mario Storti. MPI for python. *Journal of Parallel and Distributed Computing*, 65(9):1108–1115, 2005.

[39] Masoumeh Dashti, Stephen J Harris, and Andrew Stuart. Besov priors for Bayesian inverse problems. *Inverse Problems and Imaging*, 6(2):183–200, 2012.

[40] Masoumeh Dashti and Andrew M. Stuart. *"The Bayesian Approach to Inverse Problems"*. Springer International Publishing, 2017.

[41] Guillaume Desjardins, Aaron Courville, Yoshua Bengio, Pascal Vincent, and Olivier Delalleau. Tempered Markov chain Monte Carlo for training of restricted Boltzmann machines. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 145–152, 2010.

[42] Ben Mansour Dia. A continuation method in Bayesian inference. *arXiv preprint arXiv:1911.11650*, 2019.

[43] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[44] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[45] Tim J Dodwell, Chris Ketelsen, Robert Scheichl, and Aretha L Teckentrup. A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):1075–1108, 2015.

[46] TJ Dodwell, C Ketelsen, R Scheichl, and AL Teckentrup. Multilevel Markov Chain Monte Carlo. *Siam Review*, 61(3):509–545, 2019.

[47] JD Doll, Nuria Plattner, David L Freeman, Yufei Liu, and Paul Dupuis. Rare-event sampling: Occupation-based performance measures for parallel tempering and infinite swapping Monte Carlo methods. *The Journal of chemical physics*, 137(20):204112, 2012.

[48] Richard M Dudley. *Real analysis and probability*. CRC Press, 2018.

[49] Paul Dupuis, Yufei Liu, Nuria Plattner, and Jimmie D Doll. On the infinite swapping limit for parallel tempering. *Multiscale Modeling & Simulation*, 10(3):986–1022, 2012.

[50] Paul Dupuis and Guo-Jhen Wu. Analysis and optimization of certain parallel Monte Carlo methods in the low temperature limit. *arXiv preprint arXiv:2011.05423*, 2020.

[51] Adam M Dziewonski and Don L Anderson. Preliminary reference earth model. *Physics of the earth and planetary interiors*, 25(4):297–356, 1981.

[52] David J Earl and Michael W Deem. Parallel tempering: Theory, applications, and new perspectives. *Physical Chemistry Chemical Physics*, 7(23):3910–3916, 2005.

[53] Andreas Eberle, Arnaud Guillin, Raphael Zimmer, et al. Couplings and quantitative contraction rates for Langevin dynamics. *The Annals of Probability*, 47(4):1982–2010, 2019.

[54] James M Flegal and Galin L Jones. Implementing MCMC: estimating with confidence. *Handbook of Markov Chain Monte Carlo*, pages 175–197, 2011.

[55] Jerome H Friedman. *The elements of statistical learning: Data mining, inference, and prediction*. springer open, 2017.

[56] Fabrice Gamboa and Elisabeth Gassiat. Bayesian methods and maximum entropy for ill-posed inverse problems. *The Annals of Statistics*, 25(1):328–350, 1997.

[57] Charles J Geyer. Practical Markov chain Monte Carlo. *Statistical science*, pages 473–483, 1992.

[58] Roger Ghanem, David Higdon, and Houman Owhadi. *Handbook of uncertainty quantification*, volume 6. Springer, 2017.

[59] Michael B. Giles. Multilevel Monte Carlo Path Simulation. *Operations Research*, 56(3):607–617, 2008.

[60] Mike Giles, Tigran Nagapetyan, Lukasz Szpruch, Sebastian Vollmer, and Konstantinos Zygalakis. Multilevel Monte Carlo for scalable Bayesian computations. *arXiv preprint arXiv:1609.06144*, 2016.

[61] David G Glynn. The permanent of a square matrix. *European Journal of Combinatorics*, 31(7):1887–1891, 2010.

[62] Peter W Glynn and Chang-han Rhee. Exact estimation for Markov chain equilibrium expectations. *Journal of Applied Probability*, 51(A):377–389, 2014.

[63] Heikki Haario, Marko Laine, Antonietta Mira, and Eero Saksman. DRAM: efficient adaptive MCMC. *Statistics and computing*, 16(4):339–354, 2006.

[64] Heikki Haario, Eero Saksman, and Johanna Tamminen. An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223–242, 2001.

[65] Martin Hairer, Andrew M Stuart, Sebastian J Vollmer, et al. Spectral gaps for a Metropolis–Hastings algorithm in infinite dimensions. *The Annals of Applied Probability*, 24(6):2455–2490, 2014.

[66] Abdul-Lateef Haji-Ali, Fabio Nobile, and Raúl Tempone. Multi-index Monte Carlo: when sparsity meets sampling. *Numerische Mathematik*, 132(4):767–806, 2016.

[67] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[68] W Keith Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Oxford University Press*, 1970.

[69] Stefan Heinrich. Monte Carlo complexity of global solution of integral equations. *Journal of Complexity*, 14(2):151–175, 1998.

[70] Jeremy Heng and Pierre E Jacob. Unbiased Hamiltonian Monte Carlo with couplings. *Biometrika*, 106(2):287–302, 2019.

[71] Viet Ha Hoang, Christoph Schwab, and Andrew M Stuart. Complexity analysis of accelerated MCMC methods for Bayesian inversion. *Inverse Problems*, 29(8):085010, 2013.

[72] Håkon Hoel, Kody JH Law, and Raul Tempone. Multilevel ensemble Kalman filtering. *SIAM Journal on Numerical Analysis*, 54(3):1813–1839, 2016.

[73] Bamdad Hosseini. Well-posed Bayesian inverse problems with infinitely divisible and heavy-tailed prior measures. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):1024–1060, 2017.

[74] Ronald L Iman and Jon C Helton. An investigation of uncertainty and sensitivity analysis techniques for computer models. *Risk analysis*, 8(1):71–90, 1988.

[75] Muhammad Izzatullah, Tristan van Leeuwen, and Daniel Peter. Bayesian seismic inversion: A fast sampling Langevin dynamics Markov chain Monte Carlo method. *Geophysical Journal International*, 2021.

[76] Pierre E Jacob, John O'Leary, and Yves F Atchadé. Unbiased markov chain monte carlo methods with couplings. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(3):543–600, 2020.

[77] Karl Jansen, Eike H Müller, and Robert Scheichl. Multilevel Monte Carlo algorithm for quantum mechanics on a lattice. *Physical Review D*, 102(11):114512, 2020.

[78] Ajay Jasra, Kengo Kamatani, Kody J. H. Law, and Yan Zhou. A multi-index Markov Chain Monte Carlo method. *International Journal for Uncertainty Quantification*, 8(1):61–73, 2018.

[79] Ajay Jasra, Kengo Kamatani, Kody JH Law, and Yan Zhou. Multilevel particle filters. *SIAM Journal on Numerical Analysis*, 55(6):3068–3096, 2017.

[80] Ajay Jasra, Kody Law, and Yaxian Xu. Markov chain Simulation for Multilevel Monte Carlo. *arXiv preprint arXiv:1806.09754*, 2018.

[81] Yefang Jiang and Allan D Woodbury. A full-Bayesian approach to the inverse problem for steady-state groundwater flow and heat transport. *Geophysical Journal International*, 167(3):1501–1512, 2006.

[82] Valen E Johnson. A coupling-regeneration scheme for diagnosing convergence in Markov chain Monte Carlo algorithms. *Journal of the American Statistical Association*, 93(441):238–248, 1998.

[83] Galin L Jones and James P Hobert. Honest exploration of intractable probability distributions via Markov chain Monte Carlo. *Statistical Science*, pages 312–334, 2001.

[84] C. Kahle, K. Lam, J. Latz, and E. Ullmann. Bayesian parameter identification in Cahn–Hilliard models for biological growth. *SIAM/ASA Journal on Uncertainty Quantification*, 7(2):526–552, 2019.

[85] Jari Kaipio and Erkki Somersalo. *Statistical and computational inverse problems*, volume 160. Springer Science & Business Media, 2006.

[86] N. Kantas, A. Beskos, and A. Jasra. Sequential Monte Carlo Methods for High-Dimensional Inverse Problems: A case study for the Navier–Stokes equations. *SIAM/ASA J. Uncertain. Quantif.*, 2(1):464–489, 2014.

[87] Hideki Kobayashi, Paul B Rohrbach, Robert Scheichl, Nigel B Wilding, and Robert L Jack. Critical point for demixing of binary hard spheres. *Physical Review E*, 104(4):044603, 2021.

[88] Dimitri Komatitsch and Jeroen Tromp. Introduction to the spectral element method for three-dimensional seismic wave propagation. *Geophysical journal international*, 139(3):806–822, 1999.

[89] Ioannis Kontoyiannis and Sean P Meyn. Geometric ergodicity and the spectral gap of non-reversible markov chains. *Probability Theory and Related Fields*, 154(1-2):327–339, 2012.

[90] Mateusz Krzysztof Łącki and Błażej Miasojedow. State-dependent swap strategies and automatic reduction of number of temperatures in adaptive parallel tempering algorithm. *Statistics and Computing*, 26(5):951–964, 2016.

[91] Shiwei Lan. Adaptive dimension reduction to accelerate infinite-dimensional geometric Markov chain Monte Carlo. *Journal of Computational Physics*, 392:71–95, 2019.

[92] Krzysztof Łatuszyński, Błażej Miasojedow, Wojciech Niemiro, et al. Nonasymptotic bounds on the estimation error of MCMC algorithms. *Bernoulli*, 19(5A):2033–2066, 2013.

[93] Krzysztof Łatuszyński and Wojciech Niemiro. Rigorous confidence bounds for MCMC under a geometric drift condition. *Journal of Complexity*, 27(1):23–38, 2011.

[94] Jonas Latz. On the well-posedness of Bayesian inverse problems. *SIAM/ASA Journal on Uncertainty Quantification*, 8(1):451–482, 2020.

[95] Jonas Latz, Juan P Madrigal-Cianci, Fabio Nobile, and Raúl Tempone. Generalized parallel tempering on Bayesian inverse problems. *Statistics and Computing*, 31(5):1–26, 2021.

[96] Jonas Latz, Iason Papaioannou, and Elisabeth Ullmann. Multilevel Sequential^2 Monte Carlo for Bayesian inverse problems. *Journal of Computational Physics*, 368:154 – 178, 2018.

[97] Kody JH Law. Proposals which speed up function-space MCMC. *Journal of Computational and Applied Mathematics*, 262:127–138, 2014.

[98] Gregory F Lawler and Alan D Sokal. Bounds on the L^2 spectrum for Markov chains and Markov processes: a generalization of Cheeger's inequality. *Transactions of the American mathematical society*, 309(2):557–580, 1988.

[99] Torgny Lindvall. *Lectures on the coupling method*. Courier Corporation, 2002.

[100] Torgny Lindvall and L. C. G. Rogers. Coupling of Multidimensional Diffusions by Reflection. *The Annals of Probability*, 14(3):860 – 872, 1986.

[101] Anders Logg, Kent-Andre Mardal, and Garth Wells. *Automated solution of differential equations by the finite element method: The FEniCS book*, volume 84. Springer Science & Business Media, 2012.

[102] Quan Long, Marco Scavino, Raúl Tempone, and Suojin Wang. Fast estimation of expected information gains for bayesian experimental designs based on laplace approximations. *Computer Methods in Applied Mechanics and Engineering*, 259:24–39, 2013.

[103] Jianfeng Lu and Eric Vanden-Eijnden. Infinite swapping replica exchange molecular dynamics leads to a simple simulation patch using mixture potentials. *The Journal of chemical physics*, 138(8):084105, 2013.

[104] Yulong Lu. On the bernstein-von mises theorem for high dimensional nonlinear bayesian inverse problems. *arXiv preprint arXiv:1706.00289*, 2017.

[105] Mikkel B Lykkegaard, Grigorios Mingas, Robert Scheichl, Colin Fox, and Tim J Dodwell. Multilevel Delayed Acceptance MCMC with an Adaptive Error Model in PyMC3. *arXiv preprint arXiv:2012.05668*, 2020.

[106] Neal Madras, Dana Randall, et al. Markov chain decomposition for convergence rate analysis. *The Annals of Applied Probability*, 12(2):581–606, 2002.

[107] Juan Pablo Madrigal-Cianci and Fabio Nobile. Multi-Level Markov Chain Monte Carlo algorithms based on maximally-coupled proposals. *Unpublished (in preparation)*, 2021.

[108] Juan Pablo Madrigal-Cianci, Fabio Nobile, and Raul Tempone. Analysis of a class of multi-level markov chain monte carlo algorithms based on independent metropolis-hastings. *arXiv preprint arXiv:2105.02035*, 2021.

[109] E Marinari and G Parisi. Simulated tempering: A new Monte Carlo scheme. *Europhysics Letters (EPL)*, 19(6):451–458, jul 1992.

[110] Youssef Marzouk, Tarek Moselhy, Matthew Parno, and Alessio Spantini. An introduction to sampling via measure transport. *arXiv preprint arXiv:1602.05023*, 2016.

[111] Kerrie L Mengersen, Richard L Tweedie, et al. Rates of convergence of the Hastings and Metropolis algorithms. *The annals of Statistics*, 24(1):101–121, 1996.

[112] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[113] Sean P Meyn and Richard L Tweedie. *Markov chains and stochastic stability*. Springer Science & Business Media, 2012.

[114] Błażej Miasojedow, Eric Moulines, and Matti Vihola. An adaptive parallel tempering algorithm. *Journal of Computational and Graphical Statistics*, 22(3):649–664, 2013.

[115] Mohammad Motamed and Daniel Appelo. Wasserstein metric-driven Bayesian inversion with applications to signal processing. *International Journal for Uncertainty Quantification*, 9(4), 2019.

[116] Radford M Neal. Improving asymptotic variance of mcmc estimators: Non-reversible chains are better. *arXiv preprint math/0407281*, 2004.

[117] Michael A Newton and Adrian E Raftery. Approximate bayesian inference with the weighted likelihood bootstrap. *Journal of the Royal Statistical Society: Series B (Methodological)*, 56(1):3–26, 1994.

[118] Fabio Nobile and Francesco Tesei. A multi level Monte Carlo method with control variate for elliptic PDEs with log-normal coefficients. *Stochastic Partial Differential Equations: Analysis and Computations*, 3(3):398–444, 2015.

[119] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[120] Esa Nummelin. MC's for MCMC'ists. *International Statistical Review*, 70(2):215–240, 2002.

[121] Dean S Oliver, Albert C Reynolds, and Ning Liu. *Inverse theory for petroleum reservoir characterization and history matching*. 2008.

[122] Art B. Owen. *Monte Carlo theory, methods and examples*. 2013.

[123] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *arXiv preprint arXiv:1912.02762*, 2019.

[124] Matthew Parno, Tarek Moselhy, and Youssef Marzouk. A multiscale strategy for Bayesian inference using transport maps. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):1160–1190, 2016.

[125] Matthew D Parno and Youssef M Marzouk. Transport map accelerated markov chain monte carlo. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):645–682, 2018.

[126] Matthew D. Parno and Youssef M. Marzouk. Transport map accelerated markov chain monte carlo. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):645–682, Jan 2018.

[127] Daniel Paulin. Concentration inequalities for Markov chains by Marton couplings and spectral methods. *Electronic Journal of Probability*, 20, 2015.

[128] Benjamin Peherstorfer, Karen Willcox, and Max Gunzburger. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *Siam Review*, 60(3):550–591, 2018.

[129] Francis J Pinski, Gideon Simpson, Andrew M Stuart, and Hendrik Weber. Kullback–Leibler approximation for probability measures on infinite dimensional spaces. *SIAM Journal on Mathematical Analysis*, 47(6):4091–4122, 2015.

[130] Frank J Pinski, Gideon Simpson, Andrew M Stuart, and Hendrik Weber. Algorithms for Kullback–Leibler approximation of probability measures in infinite dimensions. *SIAM Journal on Scientific Computing*, 37(6):A2733–A2757, 2015.

[131] Dan Piponi, Matthew Hoffman, and Pavel Sountsov. Hamiltonian Monte Carlo Swindles. In *International Conference on Artificial Intelligence and Statistics*, pages 3774–3783. PMLR, 2020.

[132] Michele Pisaroni, Fabio Nobile, and Pénélope Leyland. A Continuation Multi Level Monte Carlo (C-MLMC) method for uncertainty quantification in compressible inviscid aerodynamics. *Computer Methods in Applied Mechanics and Engineering*, 326:20–50, 2017.

[133] Nuria Plattner, JD Doll, Paul Dupuis, Hui Wang, Yufei Liu, and JE Gubernatis. An infinite swapping approach to the rare-event sampling problem. *The Journal of chemical physics*, 135(13):134111, 2011.

[134] Alfio Quarteroni and Silvia Quarteroni. *Numerical models for differential problems*, volume 2. Springer, 2009.

[135] Alfio Quarteroni and Alberto Valli. *Numerical approximation of partial differential equations*, volume 23. Springer Science & Business Media, 2008.

[136] Raphael Zimmer. Couplings and Kantorovich contractions with explicit rates for diffusions. 2017.

[137] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[138] Gareth O Roberts and Jeffrey S Rosenthal. Geometric ergodicity and hybrid Markov chains. *Electron. Comm. Probab*, 2(2):13–25, 1997.

[139] Gareth O Roberts, Jeffrey S Rosenthal, et al. General state space Markov chains and MCMC algorithms. *Probability surveys*, 1:20–71, 2004.

[140] Gareth O Roberts and Richard L Tweedie. Bounds on regeneration times and convergence rates for Markov chains. *Stochastic Processes and their applications*, 80(2):211–229, 1999.

[141] Jeffrey S Rosenthal. A review of asymptotic convergence for general state space Markov chains. *Far East J. Theor. Stat*, 5(1):37–50, 2001.

[142] Daniel Rudolf. Explicit error bounds for lazy reversible Markov chain Monte Carlo. *Journal of Complexity*, 25(1):11–24, 2009.

[143] Daniel Rudolf. Explicit error bounds for Markov chain Monte Carlo. *Dissertationes Math.*, 485:1–93, 2012.

[144] Daniel Rudolf and Björn Sprungk. On a generalization of the preconditioned Crank–Nicolson Metropolis algorithm. *Foundations of Computational Mathematics*, 18(2):309–343, 2018.

[145] Håvard Rue, Andrea Riebler, Sigrunn H Sørbye, Janine B Illian, Daniel P Simpson, and Finn K Lindgren. Bayesian computing with INLA: a review. *Annual Review of Statistics and Its Application*, 4:395–421, 2017.

[146] Herbert John Ryser. *Combinatorial mathematics*, volume 14. American Mathematical Soc., 1963.

[147] Jerome Sacks, William J Welch, Toby J Mitchell, and Henry P Wynn. Design and analysis of computer experiments. *Statistical science*, 4(4):409–423, 1989.

[148] Arvind K Saibaba, Jonghyun Lee, and Peter K Kitanidis. Randomized algorithms for generalized hermitian eigenvalue problems with application to computing karhunen–loève expansion. *Numerical Linear Algebra with Applications*, 23(2):314–339, 2016.

[149] Filippo Santambrogio. *Optimal transport for applied mathematicians*. Springer, 1 edition, 2015.

[150] Mark J Schervish. *Theory of statistics*. Springer Science & Business Media, 2012.

[151] Claudia Schillings, Björn Sprungk, and Philipp Wacker. On the convergence of the laplace approximation and noise-level-robustness of laplace-based monte carlo methods for bayesian inverse problems. *Numerische Mathematik*, 145(4):915–971, 2020.

[152] Linus Seelinger, Anne Reinarz, Leonhard Rannabauer, Michael Bader, Peter Bastian, and Robert Scheichl. High performance uncertainty quantification with parallelized multilevel markov chain monte carlo. *arXiv preprint arXiv:2107.14552*, 2021.

[153] Björn Sprungk. Numerical methods for Bayesian inference in Hilbert spaces. 2017.

[154] Björn Sprungk. On the local Lipschitz stability of Bayesian inverse problems. *Inverse Problems*, 36(5):055015, 2020.

[155] Philip B Stark and Luis Tenorio. A primer of frequentist and Bayesian inference in inverse problems. *Large-scale inverse problems and quantification of uncertainty*, pages 9–32, 2010.

[156] Andrew M Stuart. Inverse problems: a Bayesian perspective. *Acta Numerica*, 19:451–559, 2010.

[157] Ne-Zheng Sun. *Inverse problems in groundwater modeling*, volume 6. Springer Science & Business Media, 2013.

[158] Albert Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM, 2005.

[159] Hermann Thorisson. *Coupling, stationarity and regeneration*. Springer, 2000.

[160] Luke Tierney. A note on Metropolis-Hastings kernels for general state spaces. *Annals of applied probability*, pages 1–9, 1998.

[161] Lloyd N Trefethen and David Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.

[162] Jeroen Tromp, Carl Tape, and Qinya Liu. Seismic tomography, adjoint methods, time reversal and banana-doughnut kernels. *Geophysical Journal International*, 160(1):195–216, 2005.

[163] Marc Van Der Sluys, Vivien Raymond, Ilya Mandel, Christian Röver, Nelson Christensen, Vicky Kalogera, Renate Meyer, and Alberto Vecchio. Parameter estimation of spinning binary inspirals using Markov chain Monte Carlo. *Classical and Quantum Gravity*, 25(18):184011, 2008.

[164] Umberto Villa, Noemi Petra, and Omar Ghattas. HIPPYlib: An Extensible Software Framework for Large-Scale Inverse Problems Governed by PDEs: Part I: Deterministic Inversion and Linearized Bayesian Inference. *ACM Trans. Math. Softw.*, 47(2), April 2021.

[165] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

[166] Sebastian J Vollmer. Dimension-independent mcmc sampling for inverse problems with non-gaussian priors. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):535–561, 2015.

[167] Jasper A Vrugt, CJF Ter Braak, CGH Diks, Bruce A Robinson, James M Hyman, and Dave Higdon. Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International Journal of Nonlinear Sciences and Numerical Simulation*, 10(3):273–290, 2009.

[168] Warren E Walker, Poul Harremoës, Jan Rotmans, Jeroen P Van Der Sluijs, Marjolein BA Van Asselt, Peter Janssen, and Martin P Krayer von Krauss. Defining uncertainty: a conceptual basis for uncertainty management in model-based decision support. *Integrated assessment*, 4(1):5–17, 2003.

[169] Guanyang Wang, John O'Leary, and Pierre Jacob. Maximal couplings of the metropolis-hastings algorithm. In *International Conference on Artificial Intelligence and Statistics*, pages 1225–1233. PMLR, 2021.

[170] Philipp A Witte, Mathias Louboutin, Navjot Kukreja, Fabio Luporini, Michael Lange, Gerard J Gorman, and Felix J Herrmann. A large-scale framework for symbolic implementations of seismic inversion algorithms in julia. *Geophysics*, 84(3):F57–F71, 2019.

[171] Dawn B Woodard, Scott C Schmidler, Mark Huber, et al. Conditions for rapid mixing of parallel and simulated tempering on multimodal distributions. *The Annals of Applied Probability*, 19(2):617–640, 2009.

[172] Tang-Qing Yu, Jianfeng Lu, Cameron F. Abrams, and Eric Vanden-Eijnden. Multiscale implementation of infinite-swap replica exchange molecular dynamics. *Proceedings of the National Academy of Sciences*, 113(42):11744–11749, 2016.

Juan Pablo Madrigal Cianci
Curriculum vitae

✍ Avenue de Prefaully 4,
1022–Chavannes-près-Renens (VD),
Switzerland.

☎ +41 (078) 719-4169.

✉ juan.madrigalcianci@epfl.ch

About

I am interested in data science, scientific computation, stochastic simulations, applied probability, and uncertainty quantification. A particular interest of mine is the development and analysis of efficient Monte Carlo and Markov Chain Monte Carlo methods for Bayesian inverse problems. My work and field of expertise lies at the intersection of applied mathematics, probability, and computational science and engineering.

Education

2017 – 2022	École Polytechnique Fédérale de Lausanne (EPFL). Ph.D Mathematics. Advisor: Prof. Fabio Nobile.
2015 – 2017	University of New Mexico (UNM). M.S Mathematics. GPA 4.0/4.0. Graduated with distinction.
2013 – 2015	University of New Mexico (UNM). B.S Mathematics. GPA 3.92/4.0. Summa Cum laude.
2013 – 2015	University of New Mexico (UNM). B.S Statistics. GPA 3.92/4.0. Summa Cum laude.

Communication Skills

Spanish	Native speaker
English	Bilingual, C2.
French	Intermediate, B1.

Skills

Analytical	Monte Carlo, machine learning, deep learning, optimization, regression, time series, linear algebra, finite differences/elements.
Software	Python (including numpy, scipy, scikit-learn, Tensorflow, FEniCS, anaconda, Jupyter), MATLAB, LaTeX, R, shell.

Teaching Experience

1. Analyse Avancée. EPFL. Assistant. 2019,2021.
2. Stochastic Simulations, EPFL. Assistant. 2018-2020.
3. Numerical Approximation of PDE I, EPFL. Assistant. 2018.
4. Analyse III, EPFL. Assistant. 2017.
5. Calculus III, UNM. Lecturer. 2016- 2017.
6. Calculus II, UNM. Lecturer. 2016
7. Calculus I, UNM. Assistant. 2015-2016.

Awards and Honors

2019-2021	Swiss Data Science Center fellowship. Awarded a competitive fellowship for the development of data science as a field, totaling over CHF 100,000.
2019	Special EPFL SMA Teaching prime. Awarded a CHF 1000 prime for my work as a teaching assistant of the Stochastic-simulations course.
2017	Susan Deese-Roberts Teaching Assistant of the Year Award nomination, UNM Nominated for Teaching Assistant of the Year Award, 2017, from the University of New Mexico Center for Teaching Excellence.
2015-2017	Teaching Assistantship, UNM. Assistantship covering tuition fees and a stipend, totaling over USD 40,000.
2015, 2017	Departmental Honors and University, B.S and M.S Degree, UNM. Graduated with honors in both B.S and M.S.

Selected Publications

1. Madrigal-Cianci, J. P., Nobile, F., & Tempone, R. (2021). Analysis of a class of Multi-Level Markov Chain Monte Carlo algorithms based on Independent Metropolis-Hastings. arXiv e-prints, arXiv-2105.
2. Latz, J., Madrigal-Cianci, J. P., Nobile, F., & Tempone, R. (2020). Generalized Parallel Tempering on Bayesian Inverse Problems. Stat Comput 31, 67 (2021). https://doi.org/10.1007/s11222-021-10042-6

Presentations

1. Poster. Markov Chain Monte Carlo methods for seismic source inversion. Presented at the Isaac Newton Institute, Cambridge, UK, in April, 2018.

2. Poster. A multi-level Markov Chain Monte Carlo sampler with applications to a seismic source inversion problem. Presented at the SIAM CSE 19 conference, in Spokane, Washington, USA. February 2019.

3. Talk. The infinite swapping algorithm with applications to seismology. Presented at the SIAM CSE 19 conference, in Spokane, Washington, USA. February 2019.

4. Talk. The infinite swapping algorithm For Bayesian Inverse Problems. Presented at the MATHICSE retreat, Champéry, Switzerland. June 2019.

5. Talk. The infinite swapping algorithm For Bayesian Inverse Problems. Presented at MaxEnt 19 conference, in Munich, Germany. July 2019.

6. Talk. The infinite swapping algorithm For Bayesian Inverse Problems. Presented at the Applied Inverse Problems (AIP) conference, in Grenoble, France. July 2019.

7. Poster. The infinite swapping algorithm For Bayesian Inverse Problems. Presented at the RICAM Workshop on Optimization and Inversion under Uncertainty, Linz, Austria. November 11-15, 2019.

8. Talk. Generalized parallel tempering for Bayesian Inverse problems. Invited talk at the UQ hybrid seminar in RWTH Aachen. Virtually. March 2021. Link https://www.youtube.com/watch?v=kiKgMOC9l6U&t=379s

9. Talk. Analysis of a class of ML-MCMC algorithms. Presented at the SIAM CSE21 conference, virtually. March 2021.

10. Talk. Hierarchical methods for large scale Bayesian inverse problems. Invited talk at the Swiss Data Science Center, virtually. June 2021.

11. Talk. Multi-level Markov chain Monte Carlo using maximally-coupled proposals. Presented at MaxEnt conference, virtually. July 2021.

12. Talk. Multi-level Markov chain Monte Carlo using maximally-coupled proposals. Invited talk at the UQ hybrid seminar in RWTH Aachen virtually. July 2021. Link https://www.youtube.com/watch?v=-DsUjUzOA0Q

13. Talk. Multi-level Markov chain Monte Carlo using maximally-coupled proposals to be presented at MCM21 conference, Mannheim, virtually. August 2021.

14. Talk. Generalized parallel tempering for Bayesian Inverse problems. To be presented in a workshop entitled "Accelerated statistical inference for the sciences", at the University of Bern, Switzerland, September 2021.

Upcoming:

15. Talk. Generalized parallel tempering for Bayesian Inverse problems. To be presented at the CILAMCE-PANACM 2021 conference. Virtually. November 2021.

16. Talk. multilevel MCMC methods for large scale Bayesian inverse problems. To be presented in SIAM UQ22, Atlanta, GA, USA, April 2022.

Events Organized

1. Mini-symposium co-organizer. Accelerating sampling strategies for large-scale Bayesian inverse problems Applied Inverse Problems (AIP) 2019. Co-organizer, together with Prof. Fabio Nobile (EPFL), Prof. Kody Law (University of Manchester), and Dr. Anamika Pandey (RWTH Aachen). July, 2019, in Grenoble, France. The mini-symposium consisted of 7 speakers.

2. Mini-symposium co-organizer. Multilevel and Multifidelity approaches for forward/inverse Uncertainty Quantification and optimization under uncertainty, part 3. SIAM UQ 2020. Co-organizer, together with Dr. Panagiotis Tsilifis (General Electric Research), Dr. Gianluca Geraci, Dr. Michael Eldred, Dr. John Jakeman (Sandia National Laboratories), and Prof. Alex Gorodetski (University of Michigan). Canceled amid increasing Covid-19 concerns.

3. Mini-symposium co-organizer. Recent advances in sampling techniques for large-scale Bayesian inverse problems SIAM CSE 21. Co-organizer, together with Prof. Fabio Nobile (EPFL) and Prof. Kody Law (University of Manchester). March, 2021, virtually. The mini-symposium consisted of 10 invited speakers.

Supervised Master's Students

1. Marc Witkowski. Monte Carlo methods for contact problems with rough surfaces. Master's thesis in mathematics. February 2018.

2. Mathieu Odobez. The zig-zag method. Master's thesis in mathematics. July 2018.

3. Paride Passelli. Deterministic mehtods for seismic source inversion. Master's semester project in computational science and engineering. January 2019.

4. Gavin Lee. Transport maps in Bayesian inverse problems. Master's semester project in computational science and engineering. July 2019.

5. Bruno Rodriguez. Multi-level Monte Carlo methods for coupled dynamics. Co-supervisor together with M.Sc., Sundar Ganesh. Master's semester project in computational science and engineering. June, 2021.